

INF 511 Modern Regression I

Lecture Notes

© J. Barber

August 25, 2022

Contents

1	Introduction	1
1.1	Preface	2
1.2	Before You Start	3
1.3	Initial Data Analysis	3
	Installing & Loading a Library Package	3
	Objects, Generic Functions & Method Functions	4
	Getting Help	8
	Numerical Summaries	8
	Graphical Summaries	11
	Saving/Loading Your Results	19
	Clean Up	19
1.4	When to Use Linear Modeling	20
	Objectives of Regression Analysis	22
1.5	History	23
2	Estimation	25
	Introduction	26
2.1	Linear Model	29
	(Non-)Linear or (Non-)Linear?	30
2.2	Matrix Representation	32
2.3	Estimating β	33
2.4	Least Squares Estimation	35
	Properties of the LS Estimator of Regression Model Parameters	36
	Connections: Method of Maximum Likelihood	38
	Multi-variate Normal Distribution	38
	Special Case: Traditional Normal Linear Model	39
	Normal Likelihood	39
	Method of Maximum Likelihood	40
2.5	Examples of Calculating $\hat{\beta}$	41
2.6	Example: Galapagos Island Biogeography	41
2.7	QR Decomposition	47
2.8	Gauss-Markov Theorem	47
2.9	Goodness of Fit	49

2.10 Identifiability	55
2.11 Orthogonality	56
Example: Chemical Odor	57
3 Inference	61
Introduction	63
3.1 Hypothesis Tests to Compare Models	64
Likelihood Ratio Test	67
General Linear Hypothesis (GLH)	68
Reality, Decisions, Errors & Error Probabilities	72
Careful Wording	74
p-value	76
Reporting Test Results	77
Hypothetical Replications	79
3.2 Testing Examples	81
3.2.1 Test of all the predictors (overall F-test)	81
3.2.2 Testing one predictor	87
3.2.3 Testing a pair of predictors	90
3.2.4 Testing a subspace	91
3.2.5 Non-Zero Hypothesized Value $H_0: \beta_{Elevation} = 0.5$	93
3.2.6 Tests we cannot do (in INF 511)	95
Power	95
Non-Central F Distribution	96
Example: Power for Effect of One Predictor	97
pwr Package	99
Example: A Priori Power Analysis	102
Example: Post Hoc Power Analysis	107
Test Sidedness, Distribution Tails & Power	111
3.3 Permutation Tests	120
3.3.1 Example: Permutation Test of Overall Association	122
3.3.2 Example: Permutation Test of Single Predictor	124
3.3.3 Take-Home Remarks	127
3.4 Sampling	128
3.5 Confidence Intervals for β	128
Correspondence Between Tests & Intervals	133
3.6 Bootstrap Confidence Intervals	135
4 Prediction	139
4.1 Confidence Intervals for Predictions	140
4.2 Example: Predicting Body Fat	143
4.3 Autoregression	148
4.4 What Can Go Wrong with Predictions?	156

5 Explanation	157
Introduction	159
Conditional Mean Model vs. Marginal Mean Model	159
A Justification for Linear Modeling	160
Extrapolation, Meaningful Parameters & Reparameterization	162
Covariates Observed Without Error	164
5.1 Simple Meaning	164
Example: Galapagos Island Biogeography	165
Problems With Our Simple Meaning of Effect	169
5.2 Causality	170
5.3 Designed Experiments	173
Example: Motivation and Creativity	181
t Approximation, Randomization Distribution & Permutation Distribution	184
Formalizing Some Concepts	187
Restricted Randomization	189
Replicate Treatments	191
5.4 Observational Data	193
Voting Example	196
5.5 Matching	202
5.6 Covariate Adjustment	207
5.7 Qualitative Support for Causation	211
Sampling	211
Sampling Distribution: Example	214
Scope of Inference: Summary	223
6 Diagnostics	227
Why check your model?	229
Things to Check	230
6.1 Checking Error Assumptions	231
6.1.1 Constant Variance	232
Prototypical Plots	232
Savings Data Example	233
Typical Fan Pattern	235
Calibrate Your Eye	235
Tests of Non-constant Variance	241
Brown–Forsythe Test	241
Savings Data Example	242
Breusch–Pagan Test	244
Savings Data Example	245
Transformations of Outcomes (Y)	246
A Juggling Act	249
Box–Cox Procedure	249
Galapagos Data Example	250

Remedial Actions for Non-Constant Variance	252
6.1.2 Normality	253
Normal Probability Plot	253
Savings Data Example	255
Calibrate Your Eye	256
Correlation Tests of Normality	258
Test 1	258
Savings Data Example	259
Test 2: Shapiro-Wilk	260
Savings Data Example	261
Remedial Actions for Non-Normality	261
6.1.3 Correlated Errors	261
Global Warming Data Example	262
Durbin-Watson Test for Correlated Errors	264
Remedial Actions for Correlated Errors	265
6.2 Finding Unusual Observations (Outliers)	265
6.2.1 Leverage	266
Outlying X : Hat Matrix Leverage Values	266
Savings Data Example	268
6.2.2 Outliers	272
Outlying Y : Studentized Deleted Residuals	272
Side note: Bonferroni family-wise error rate	281
Remedial Actions for Outliers	284
6.2.3 Influential Observations	285
Influence of Case i on a Single Fitted Value: DFFITS	285
Cook's Distance	286
Influence of Case i on Regression Coefficients: DFBETAS	288
Savings Data Example	289
Default Diagnostic Plots	293
Influence on Inferences of Interest	295
6.3 Checking the Systematic Structure of the Model	295
Added-Variable Plots aka Partial Regression Plots	295
Savings Data Example	296
Partial Residual Plot	298
6.4 Discussion	303
7 Problems with Predictors	305
7.1 Errors in the Predictors	306
7.2 Change of Scale	307
7.3 Collinearity	309
8 Problems with the Error	311
8.3 Testing for Lack of Fit	312

Lecture 0	Page v
14 Categorical Predictors	327
14.1 A Two-Level Factor	329
14.1.1 PTSD Data Example	329
14.2 Factors and Quantitative Predictors	342
14.2.1 PTSD Data Example	342
14.3 Interpretation with Interaction Terms	351
14.3.1 Weekly Natural Gas Consumption Data Example	351
14.4 Factors with More Than Two Levels	358
14.4.1 Cell Reference Coding (Again)	358
14.4.2 Longevity and Sexual Activity Data Example	359
14.4.3 Let's Recall Mixing and Fixing (Nothing to Do with Sex)	374
14.5 Alternative Codings of Qualitative Predictors	377
15 One Factor Models	379
Introduction	383
15.1 The Model and Example	384
Motivating Example	384
Notation & Initial Concepts	386
15.1.1 Cell Means Model	388
15.1.2 Cell Means Model (Example Continued)	392
15.1.3 Effects Model: Before Constraints	397
15.1.4 Effects Model: Treatment Coding/Constraint	401
15.1.5 Effects Model: Treatment Coding/Constraint Example	402
15.1.6 Effects Model: Sum (to Zero) Coding/Constraint	408
15.1.7 Effects Model: Sum to Zero Coding/Constraint Example	409
15.1.8 Regression Approach to ANOVA	414
15.1.9 Model, Parameterization, Reparameterization, Coding, Constraints	415
15.2 An Example (NOT)	416
15.3 Diagnostics	416
15.4 Pairwise Comparisons	421
Simultaneous Inferences & Multiple Comparison Procedures	422
Confidence Intervals	423
Tests	424
Hypothetical Replications	424
Example: Confidence Intervals	425
Family-wise Confidence Level	426
Family-wise Error Rate	427
Bonferroni Inequality & Multiple Comparison Procedure	427
Tukey–Kramer Pairwise Comparison Procedure	430
Tukey Pairwise Comparison Example	431
Scheffé's Procedure for Contrasts	435
Pre-planned Comparisons and Data Snooping	435
Remarks on Multiple Comparison Procedures	439

15.5 False Discovery Rate	440
16 Models with Several Factors	447
16.1 Initial Concepts and Notation	449
16.2 Example	451
16.3 Cell Means Model of $E(Y_{ijk} \mathbf{x})$	456
16.4 Effects Model: Before Constraints	458
16.5 Effects Model: Sum-to-Zero Constraints/Coding	460
16.5.1 Effects Model: STZ Coding: Parameter Interpretation	462
16.5.2 (Non-)Additive Model & Saturation	464
16.5.3 ANOVA Model Components: Means and Effects	465
16.5.4 Effects Model: STZ: Example: Initial Analysis	472
16.5.5 Effects Model: STZ: Example: Diagnostics	476
16.5.6 Effects Model: STZ: Example: ANOVA For Common $C\beta$	486
16.5.7 Pit Stop: What is ANOVA?	487
16.5.8 Effects Model: STZ: Example: F v R & $C\beta$ Approach	490
16.5.9 Effects Model: STZ: Example: Summary So Far	492
16.6 Effects Model: Treatment Constraints/Coding	493
16.6.1 Effects Model: Trmt Coding: Parameter Interpretation	494
16.6.2 (Non-)Additive Model & Saturation	496
16.6.3 Effects Model: Trmt: Example: Initial Analysis	498
16.6.4 Effects Model: Trmt: Example: ANOVA For Common $C\beta$	502
16.6.5 Effects Model: Trmt: Example: F v R & $C\beta$ Approach	503
16.6.6 Effects Model: Trmt: Example: Summary	505
16.7 SS Type, Balance & the Marginality Principle	505
16.7.1 Sequential (Type I) SS ANOVA	505
16.7.2 Partial (Type III) SS ANOVA	510
16.7.3 Marginality Principle (aka Hierarchy Principle)	512
16.7.4 Summary & Remarks	516
16.8 Additive Model: Tests for Overall Main Effects	516
16.8.1 F v R Approach	517
16.8.2 $C\beta$ Approach	519
16.9 Additive Model: More Detailed Inference of Main Effects	521
16.10 Final Remarks	531
A Basic Results in Probability and Statistics	535
A.1 Review	537
A.2 Summation Operator	538
A.2.1 Properties of Summation Operator	539
A.3 Double Summation Operator	540
A.4 Product Operator	540
A.5 R Example	540
A.6 Logarithms and Exponentiation	542

Lecture 0	Page vii
A.7 Random Variables	545
A.7.1 Modeling Reality	547
A.7.2 Discrete Random Variables	550
A.7.3 Continuous Random Variables	558
A.7.4 Some Remarks	565
A.8 Characteristics of Random Variables	566
A.8.1 Expected (Mean) Value	566
A.8.2 Variance Operator	570
A.9 Random Vectors	573
A.9.1 Covariance Operator & Its Properties	573
A.9.2 Independence	576
A.10 Linear Combinations of RVs	578
A.11 Central Limit Theorem	579
A.12 pdqr Functions in R	580
A.12.1 Example	582
B Matrices & Vectors	587
B.1 Notation, Dimension, Rows, Columns, Elements	589
B.2 Matrix Arithmetic	593
B.2.1 Addition/Subtraction	593
B.2.2 Multiply a Matrix by a Scalar	595
B.2.3 Matrix Multiplication	596
B.2.4 Matrix Transpose	598
B.2.5 Special Matrices	601
B.2.6 Linear Dependence & Rank	606
B.3 Combining Things: Random Vectors and Matrices	618
B.3.1 Expectation of a Random Vector/Matrix	619
B.3.2 Variance(–Covariance) Matrix	620
B.3.3 Linearity of Expectation Operator (just as in scalar case)	620
B.3.4 Variance(–Covariance) of $a + BY$	621
B.3.5 Distribution of Linear Function of Normal RV	621
B.4 Normal and χ^2 Results	625
B.5 t and F Distribution Results	631
B.6 Joint, Marginal & Conditional Distributions	634
B.7 Conditional Distribution Model Specification	644
C Bayesian Linear Model	647
C.1 Introduction	650
C.1.1 Data Distribution	650
C.1.2 Bayes Theorem: Data, Prior, Joint, Posterior $[\theta y]$	651
C.1.3 Prior Predictive $[y]$	654
C.1.4 Posterior Predictive $[y^* y]$	655
C.1.5 Summary	658

C.1.6	Remarks	659
C.2	Linear Model	660
C.2.1	Overview	661
C.2.2	Conditional Normal Prior & Posterior for $\beta \sigma^2$	662
C.3	Conjugate Prior	665
C.3.1	Posterior	667
C.3.2	Marginal Posterior for β is a t	670
C.3.3	Posterior Predictive is a t	672
C.3.4	Remarks	674
C.4	A Common Improper Prior	675
C.4.1	Posterior	676
C.4.2	Marginal Posterior for β is a Familiar t	676
C.4.3	Posterior Predictive is a Familiar t	677
C.5	STAT 101 Redux a la Bayes	678
C.5.1	t -based Intervals for β_j	679
C.5.2	t -based Test for β_j	681
C.5.3	t -based Intervals for $E(Y x) = x^t \beta$	683
C.5.4	t -based Prediction Intervals for $Y x$	684
C.6	Prostate Data Example with Improper Prior	685
C.6.1	Frequentist R Summary	687
C.6.2	Bayesian Summary	687
C.7	A Common Independence Prior	692
C.7.1	Full Conditional Posterior Distributions	693
C.8	2-Stage Gibbs Sampling	695
C.9	3-Stage Gibbs Sampling: Prostate Data Example with "Combo" Prior	696
C.9.1	Eliciting a Prior	697
C.9.2	Full Conditionals	698
C.9.3	Gibbs Sampling Algorithm	701
C.9.4	Gibbs Sampling Code	702
C.9.5	Posterior Convergence Diagnostics with coda	705
C.9.6	Posterior Summaries with coda	711
C.9.7	Regression Function and Posterior Predictive	714
C.10	HMC in Stan: Prostate Data Example with "Combo" Prior	716
C.10.1	Functions Block	717
C.10.2	Data Block	717
C.10.3	Transformed Data Block	718
C.10.4	Parameters Block	718
C.10.5	Transformed Parameters Block	718
C.10.6	Model Block	718
C.10.7	Generated Quantities Block	719
C.10.8	Altogether for Stan	719
C.10.9	Translate Stan to C++ with stanc	721

Lecture 0	Page ix
C.10.10 Make an Executable Stan Model with <code>stan_model</code>	721
C.10.11 Data List for Stan	722
C.10.12 List of Initial Value Lists for Stan	722
C.10.13 Executing a Stan Model with <code>sampling</code>	723
C.10.14 Posterior Convergence Diagnostics with <code>coda</code>	723
C.10.15 Posterior Summaries with <code>coda</code>	732
C.11 Prostate Data Example Summary	733
C.12 Other Priors	737

List of Tables

3.1 Reality, decisions, errors and error probabilities.	72
---	----

List of Figures

2.1 Bivariate regression model	31
2.2 Least squares geometry	34
5.1 Scope of inference	223
A.1 A view of the process of doing statistics.	548
A.2 A useful model of reality?	549
A.3 Relationship between pmf and cdf.	551
A.4 Table of cumulative binomial probabilities (Source: forgotten!).	556
A.5 Relationship between pdf and cdf.	558
A.6 Continuous cdf.	560
A.7 Z-table (source: [KNNL05, Table B.1]).	563
A.8 Discrete uniform pmf supported on 1, 2, 3.	568
A.9 pdf is balanced on mean (μ).	569
A.10 Sketch to illustrate covariance.	574
A.11 Sketch of marginal and joint distributions.	577

Lecture 1

Introduction

Contents

1.1	Preface	2
1.2	Before You Start	3
1.3	Initial Data Analysis	3
	Installing & Loading a Library Package	3
	Objects, Generic Functions & Method Functions	4
	Getting Help	8
	Numerical Summaries	8
	Graphical Summaries	11
	Saving/Loading Your Results	19
	Clean Up	19
1.4	When to Use Linear Modeling	20
	Objectives of Regression Analysis	22
1.5	History	23

Reading:

- [Far14, Chap. 1]
- These notes!

R

1.1 Preface

- **Quickly.** We will move quickly through this first chapter of notes as much of it is relatively easily accessible in [Far14, Chap. 1 & App. A] and in countless other resources on the Internet and in many books.
- **Download & Install R.** I assume that you are able to download and install R from <https://www.r-project.org>. Isn't it fair to assume that a person enrolled in a PhD course in Informatics can do this?
- **Textbook Site.** See <http://www.maths.bath.ac.uk/~jjf23/LMR/> for R scripts used in your textbook, Linear Models with R ([Far14]). I use them in our notes, sometimes edited.
- **By the Book's Numbers.** We (our notes) will follow the section numbers of your textbook, but we may add **unnumbered** sections not directly related to your textbook and may add numbered subsections to numbered sections in your textbook. (Our note appendices do not correspond to [Far14].)
- **I WILL AUGMENT THESE NOTES WITH HANDWRITTEN ANNOTATION DURING LECTURE, ESPECIALLY IN THE R CODE HEREIN!!!**

1.2 Before You Start

- Please read the preliminary section [Far14, §1.1] on points relating to
 - **formulating the problem** and
 - **understanding the how the data are collected.**
- We will have much more to say about the second item, later!

1.3 Initial Data Analysis

- **READ** [Far14, §1.2]

Installing & Loading a Library Package

- Is the **faraway** package loaded? No, not by default. (Our first R code chunk, below.)

```
> search()  
  
[1] ".GlobalEnv"      "package:knitr"    "ESSR"  
[4] "package:stats"   "package:graphics" "package:grDevices"  
[7] "package:utils"    "package:datasets" "package:methods"  
[10] "Autoloads"       "package:base"
```

- Let's get the package, if necessary, then load the package, and check to see that it's loaded.

- **Side Note: Python Modules & R Library Packages.** R libraries/packages are similar to Python modules that you can ‘add-on’ to the base installations.

```
> ## install.packages("faraway", depend=TRUE) ## <- (not run)
> library(faraway) ## `load' or make package objects available
> search()

[1] ".GlobalEnv"           "package:faraway"    "package:knitr"
[4] "ESSR"                  "package:stats"     "package:graphics"
[7] "package:grDevices"    "package:utils"      "package:datasets"
[10] "package:methods"      "Autoloads"         "package:base"
```

Objects, Generic Functions & Method Functions

- We'll use the **pima** data set on female Pima Indians living near Phoenix, AZ, to illustrate R fundamentals, here, and some numerical and graphical data summaries, shortly.

```
> ls()
character(0)

> data(pima, package="faraway")
> ls()

[1] "pima"

> find("pima")

[1] ".GlobalEnv"           "package:faraway"

> head(pima)
```

	pregnant	glucose	diastolic	triceps	insulin	bmi	diabetes	age	test
1	6	148	72	35	0	33.6	0.627	50	1
2	1	85	66	29	0	26.6	0.351	31	0
3	8	183	64	0	0	23.3	0.672	32	1
4	1	89	66	23	94	28.1	0.167	21	0
5	0	137	40	35	168	43.1	2.288	33	1
6	5	116	74	0	0	25.6	0.201	30	0

```
> summary(pima)
```

pregnant	glucose	diastolic	triceps
Min. : 0.00	Min. : 0	Min. : 0.0	Min. : 0.0
1st Qu.: 1.00	1st Qu.: 99	1st Qu.: 62.0	1st Qu.: 0.0
Median : 3.00	Median :117	Median : 72.0	Median :23.0
Mean : 3.85	Mean :121	Mean : 69.1	Mean :20.5
3rd Qu.: 6.00	3rd Qu.:140	3rd Qu.: 80.0	3rd Qu.:32.0
Max. :17.00	Max. :199	Max. :122.0	Max. :99.0
insulin	bmi	diabetes	age
Min. : 0.0	Min. : 0.0	Min. :0.078	Min. :21.0
1st Qu.: 0.0	1st Qu.:27.3	1st Qu.:0.244	1st Qu.:24.0
Median : 30.5	Median :32.0	Median :0.372	Median :29.0
Mean : 79.8	Mean :32.0	Mean :0.472	Mean :33.2
3rd Qu.:127.2	3rd Qu.:36.6	3rd Qu.:0.626	3rd Qu.:41.0
Max. :846.0	Max. :67.1	Max. :2.420	Max. :81.0
test			
Min. :0.000			
1st Qu.:0.000			
Median :0.000			
Mean :0.349			
3rd Qu.:1.000			
Max. :1.000			

- Fundamental object type (i.e., **mode**), object **classes** and **method** functions give us some idea about how R works.

```
> mode(pima)
[1] "list"

> class(pima)
[1] "data.frame"

> sapply(pima, data.class)

pregnant    glucose diastolic    triceps    insulin        bmi    diabetes
"numeric" "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"
      age      test
"numeric" "numeric"

> methods(class="data.frame")

[1] [          [[          [[<-
[5] $<-          aggregate   anyDuplicated anyNA
[9] as.data.frame as.list      as.matrix     as.vector
[13] by           cbind       coerce        dim
[17] dimnames     dimnames<-  droplevels   duplicated
[21] edit          format      formula      head
[25] initialize   is.na       Math         merge
[29] na.exclude   na.omit     Ops          plot
[33] print         prompt      rbind       row.names
[37] row.names<-  rowsum      show        slotsFromS3
[41] split         split<-    stack        str
[45] subset        summary     Summary     t
[49] tail          transform  type.convert unique
[53] unstack      within      xtfrm

see '?methods' for accessing help and source code
```

- A **generic function** can dispatch a **method function** depending on the class of object.

```
> methods(generic.function="summary")  
[1] summary,ANY-method  
[3] summary,sparseMatrix-method  
[5] summary.aov  
[7] summary.aspell*  
[9] summary.connection  
[11] summary.corARMA*  
[13] summary.corCompSymm*  
[15] summary.corGaus*  
[17] summary.corLin*  
[19] summary.corRatio*  
[21] summary.corStruct*  
[23] summary.data.frame  
[25] summary.default  
[27] summary.factor  
[29] summary.gls*  
[31] summary.lm  
[33] summary.lmList*  
[35] summary.loess*  
[37] summary.manova  
[39] summary.merMod*  
[41] summary.modelStruct*  
[43] summary.nls*  
[45] summary.packageStatus*  
[47] summary.pdCompSymm*  
[49] summary.pdIdent*  
[51] summary.pdMat*  
[53] summary.pdSymm*  
[55] summary.POSIXct  
[57] summary.ppr*  
[59] summary.prcomplist*  
[61] summary.proc_time  
[63] summary.rlm*  
[65] summary.srcfile  
[67] summary.stepfun  
[69] summary.summary.merMod*  
[71] summary.trellis*  
[73] summary.varComb*  
[75] summary.varConstProp*  
[77] summary.varFixed*  
[79] summary.varIdent*  
[81] summary.warnings  
[summary,diagonalMatrix-method]  
[summary.allFit*]  
[summary.aovlist*]  
[summary.check_packages_in_dir*]  
[summary.corAR1*]  
[summary.corCAR1*]  
[summary.corExp*]  
[summary.corIdent*]  
[summary.corNatural*]  
[summary.corSpher*]  
[summary.corSymm*]  
[summary.Date]  
[summary.ecdf*]  
[summary.glm]  
[summary.infl*]  
[summary.lme*]  
[summary.lmList4*]  
[summary.loglm*]  
[summary.matrix]  
[summary.mlm*]  
[summary.negbin*]  
[summary.nlsList*]  
[summary.pdBlocked*]  
[summary.pdDiag*]  
[summary.pdLogChol*]  
[summary.pdNatural*]  
[summary.polr*]  
[summary.POSIXlt]  
[summary.prcomp*]  
[summary.princomp*]  
[summary.reStruct*]  
[summary.shingle*]  
[summary.srcref]  
[summary.stl*]  
[summary.table]  
[summary.tukeysmooth*]  
[summary.varConstPower*]  
[summary.varExp*]  
[summary.varFunc*]  
[summary.varPower*]
```

see '?methods' for accessing help and source code

Getting Help

- Typical ways to get **help** in R (output omitted).

```
> ?summary # --> or help(summary)
> ??summary # --> or help.search("summary")
> help.start() # opens browser...
> ## and the Web, of course
```

```
> ## Use of package namespace operator when getting help  
> ?faraway::pima  
> ## ?":;" ## (not run)
```

Numerical Summaries

- A previous summary suggests something unusual about the data in `pima`; zero has been used to indicate missing values.
 - Let's use R's **missing value** (`NA` “Not Available”) symbol, instead.
 - Warning: The `length` of the R vectors includes missing values!

```

> ## Logical indexing and replacement
> pima$diastolic[pima$diastolic == 0] <- NA
> pima$glucose[pima$glucose == 0] <- NA
> pima$triceps[pima$triceps == 0] <- NA
> pima$insulin[pima$insulin == 0] <- NA
> pima$bmi[pima$bmi == 0] <- NA
>
> ## Again, length includes NAs!
> length(pima$diastolic)

[1] 768

> sum(is.na(pima$diastolic))

[1] 35

> sum(!is.na(pima$diastolic))

[1] 733

```

- Better:

```

> summary(pima)

  pregnant      glucose      diastolic      triceps
Min.    : 0.00  Min.    : 44  Min.    : 24.0  Min.    : 7.0
1st Qu.: 1.00  1st Qu.: 99  1st Qu.: 64.0  1st Qu.:22.0
Median  : 3.00  Median  :117  Median  : 72.0  Median  :29.0
Mean    : 3.85  Mean    :122  Mean    : 72.4  Mean    :29.2
3rd Qu.: 6.00  3rd Qu.:141  3rd Qu.: 80.0  3rd Qu.:36.0
Max.    :17.00  Max.    :199  Max.    :122.0  Max.    :99.0
                  NA's    :5   NA's    :35     NA's    :227
  insulin       bmi          diabetes      age
Min.    : 14.0  Min.    :18.2  Min.    :0.078  Min.    :21.0
1st Qu.: 76.2  1st Qu.:27.5  1st Qu.:0.244  1st Qu.:24.0
Median  :125.0  Median  :32.3  Median  :0.372  Median  :29.0
Mean    :155.6  Mean    :32.5  Mean    :0.472  Mean    :33.2
3rd Qu.:190.0  3rd Qu.:36.6  3rd Qu.:0.626  3rd Qu.:41.0

```

```
Max.    :846.0   Max.    :67.1   Max.    :2.420   Max.    :81.0
NA's    :374     NA's    :11
test
Min.    :0.000
1st Qu.:0.000
Median  :0.000
Mean    :0.349
3rd Qu.:1.000
Max.    :1.000
```

- **Categorical variables** are often called **factors** (**classification**) variables or **qualitative** variables). See Def. A.13.
- Let's make sure R 'sees' a factor as it should and give the factor more meaningful values; the values of a factor variable are often called **levels**.

```
> ## Change test data.class from numeric to factor
> data.class(pima$test)
[1] "numeric"
> pima$test<- factor(pima$test)
> data.class(pima$test)
[1] "factor"
> summary(pima$test)
 0    1
500 268
> levels(pima$test)
[1] "0" "1"
> levels(pima$test) <- c("neg", "pos") ## <-- replacement function
> summary(pima$test)
neg pos
500 268
> levels(pima$test)
[1] "neg" "pos"
```

- Typical numerical summaries

```
> mean(pima$diastolic)
[1] NA

> mean(pima$diastolic, na.rm=TRUE)
[1] 72.405

> var(pima$diastolic, na.rm=TRUE)
[1] 153.32

> sd(pima$diastolic, na.rm=TRUE)
[1] 12.382

> median(pima$diastolic, na.rm=TRUE)
[1] 72

> quantile(pima$diastolic, na.rm=TRUE,
+            probs=c(0.025,0.25,.5,0.75,0.975))
2.5%    25%    50%    75%   97.5%
50.0   64.0   72.0   80.0   97.4
```

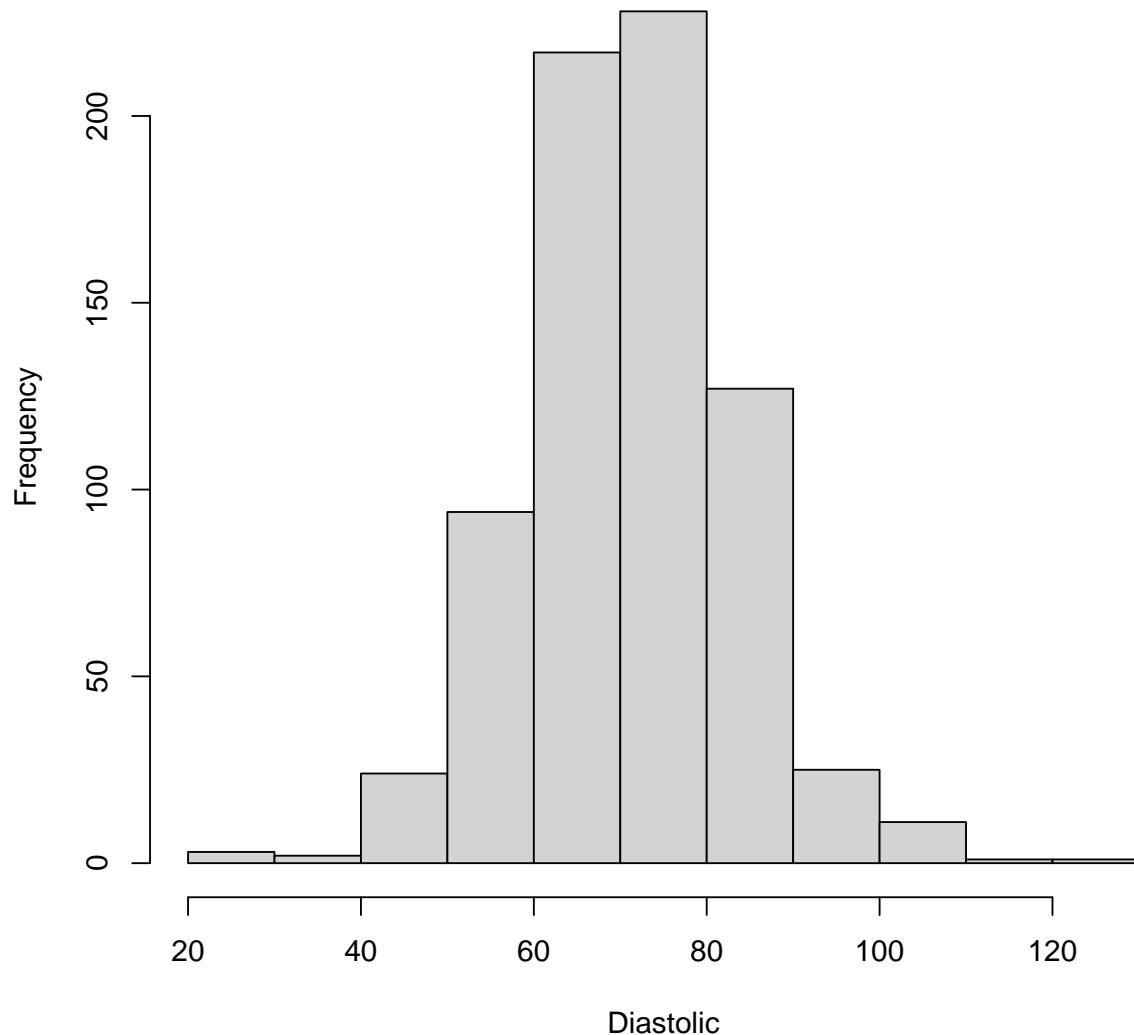
- See [Far14, §1.2] for more numerical summaries.

Graphical Summaries

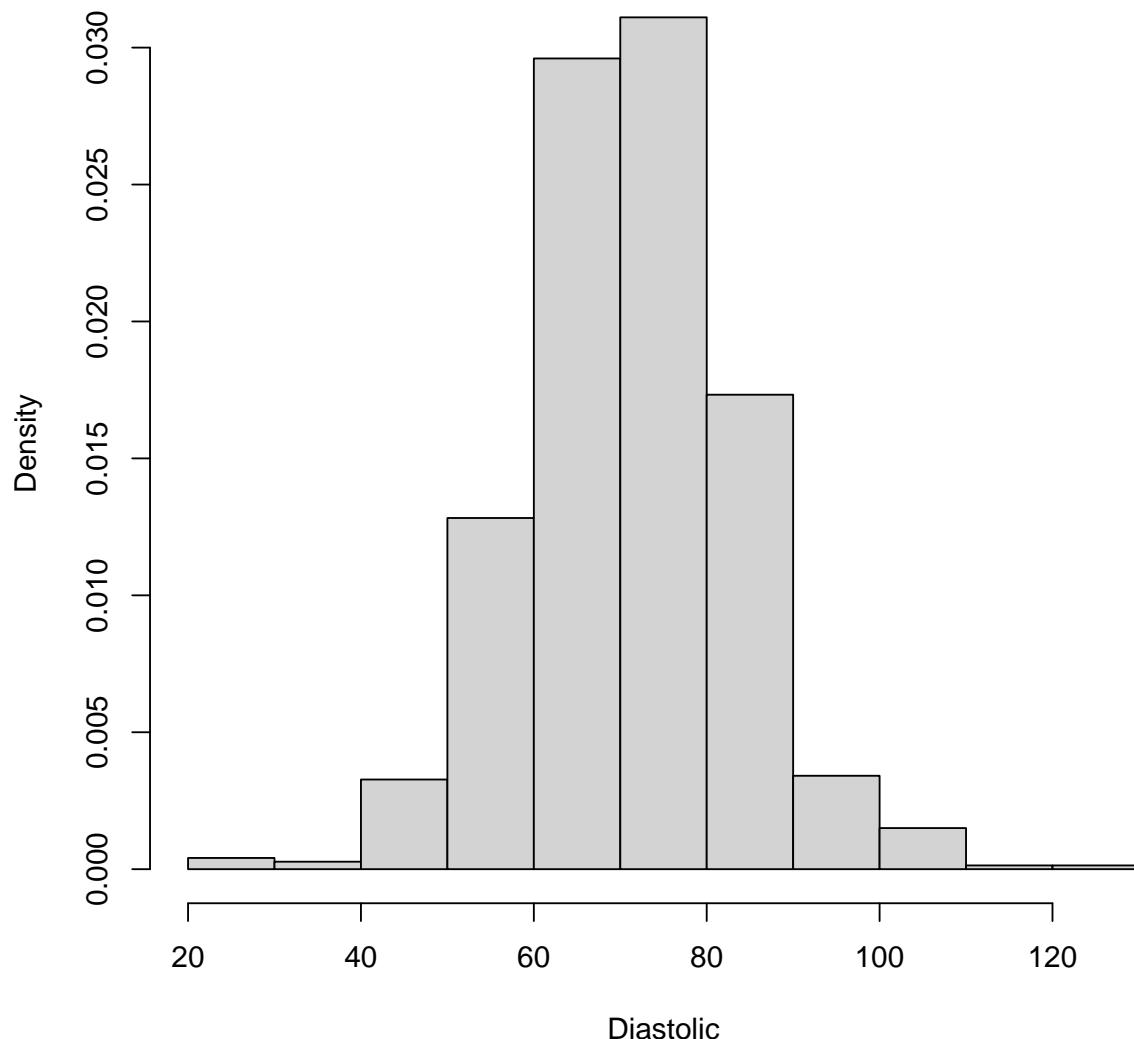
- **R Graphics 3rd Edition.** See the book by Paul Murrell [Mur21] (<https://www.stat.auckland.ac.nz/~paul/RG3e/>). Paul's book discusses the `graphics` package, `grid` package, `lattice` package (Deepayan Sarkar's R port of the S language Trellis graphics by William Cleveland), the `ggplot2` package (Hadley Wickham's implementation of the book, Grammar of Graphics, by the late Leland Wilkinson) and more. Both `lattice` and `ggplot2` are based on `grid` (in different ways), which is powerful but relatively low level, and I suspect few of us will ever likely use `grid` directly.
- **We Only Scratch the Surface.**

- Histogram. A summary of the distribution of a (usually single 'x' or 'y') quantity, here, diastolic blood pressure.

```
> hist(pima$diastolic, xlab="Diastolic", main="")
```

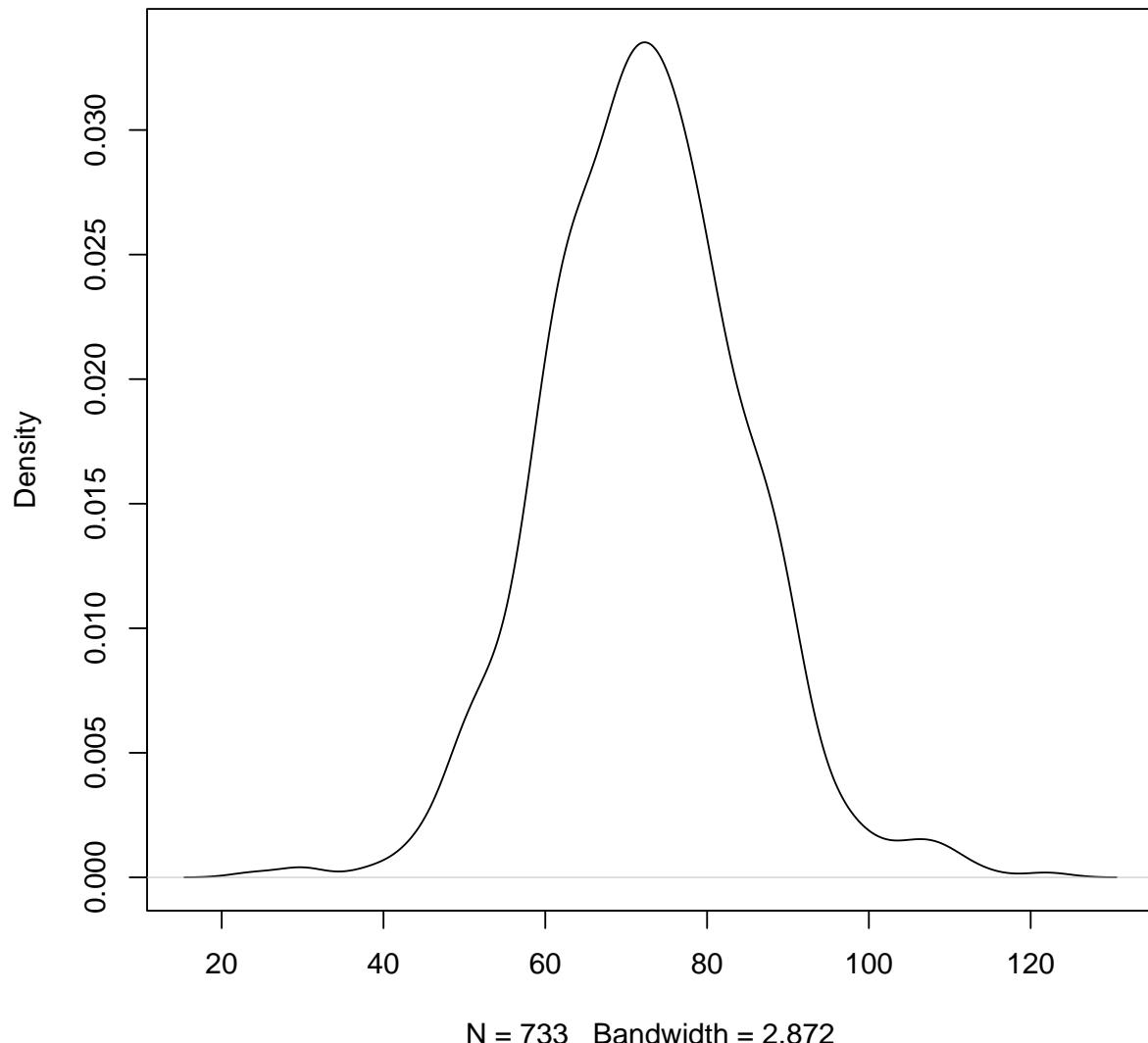


```
> hist(pima$diastolic, xlab="Diastolic", main="", prob=TRUE)
```



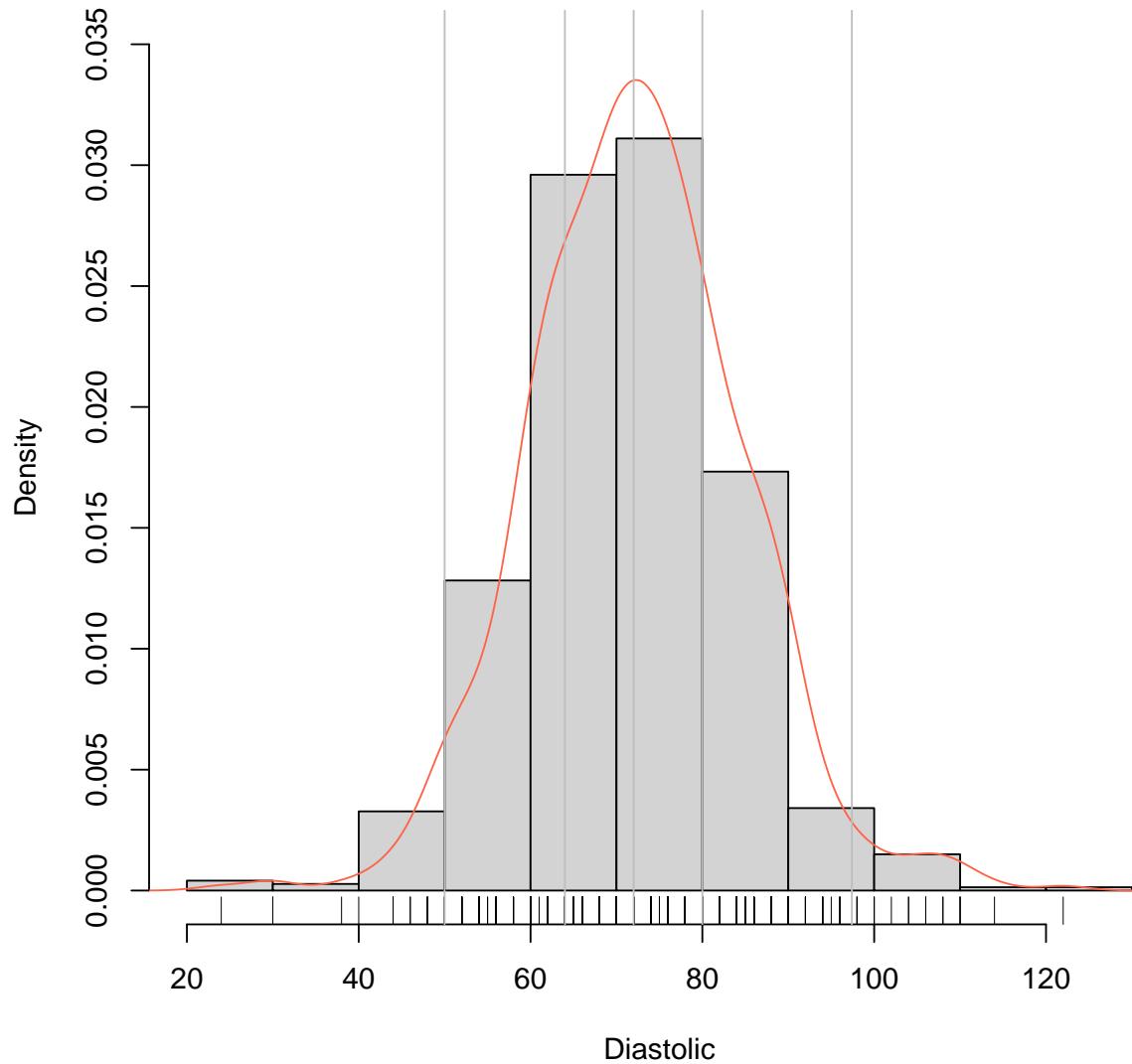
- Density plot. A smooth sort of histogram.

```
> plot(density(pima$diastolic,na.rm=TRUE),main="")
```



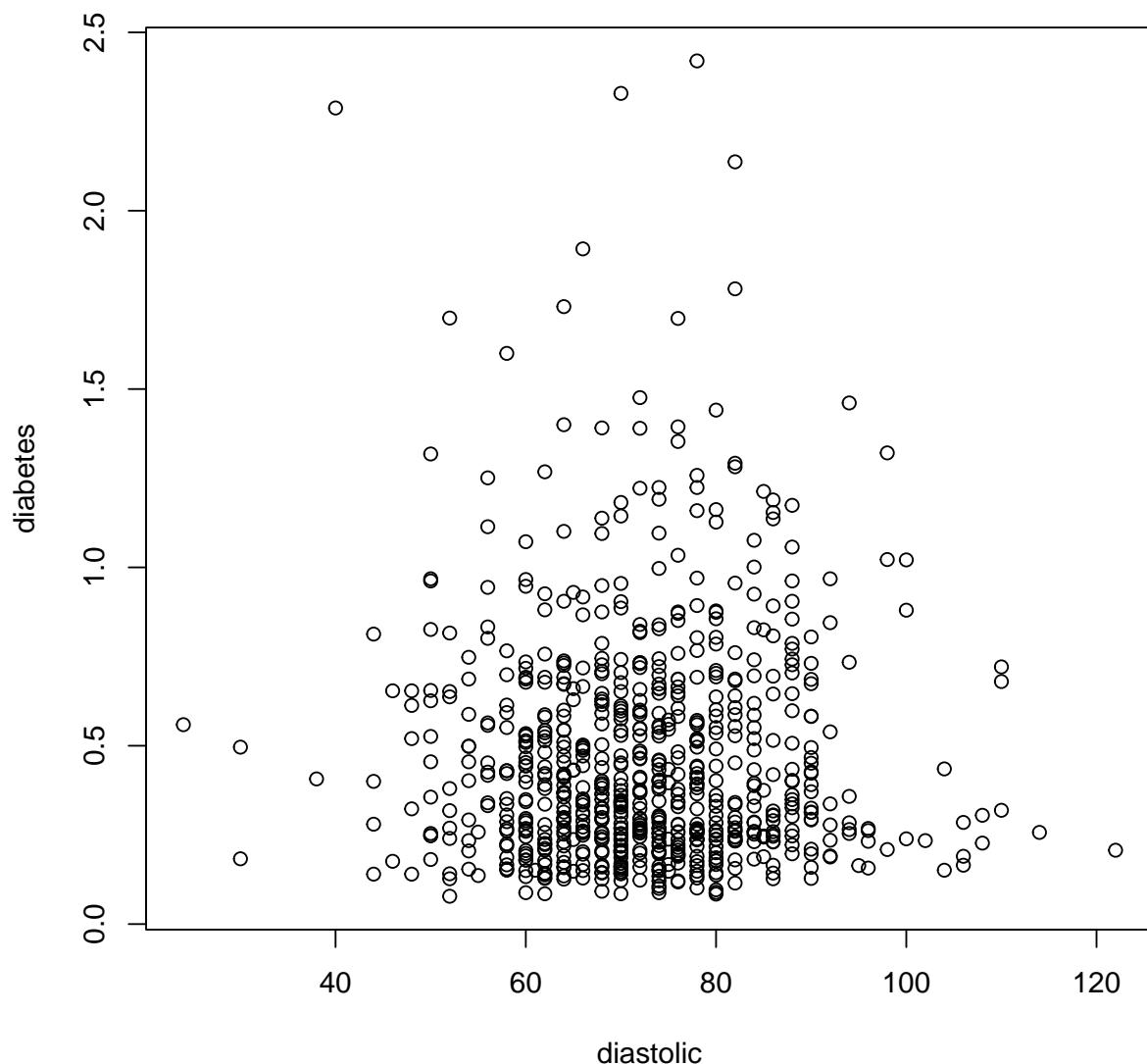
- Overlay histogram with density plot, add a data 'rug' and indicate quantiles (as computed in a previous chunk)

```
> hist(pima$diastolic, xlab="Diastolic", main="", prob=TRUE,
+       ylim=c(0,0.035))
> lines(density(pima$diastolic,na.rm=TRUE),main="", col="tomato")
> rug(pima$diastolic)
> qtiles<- quantile(pima$diastolic, na.rm=TRUE,
+                      probs=c(0.025,0.25,.5,0.75,0.975))
> abline(v=qtiles, col="grey")
```



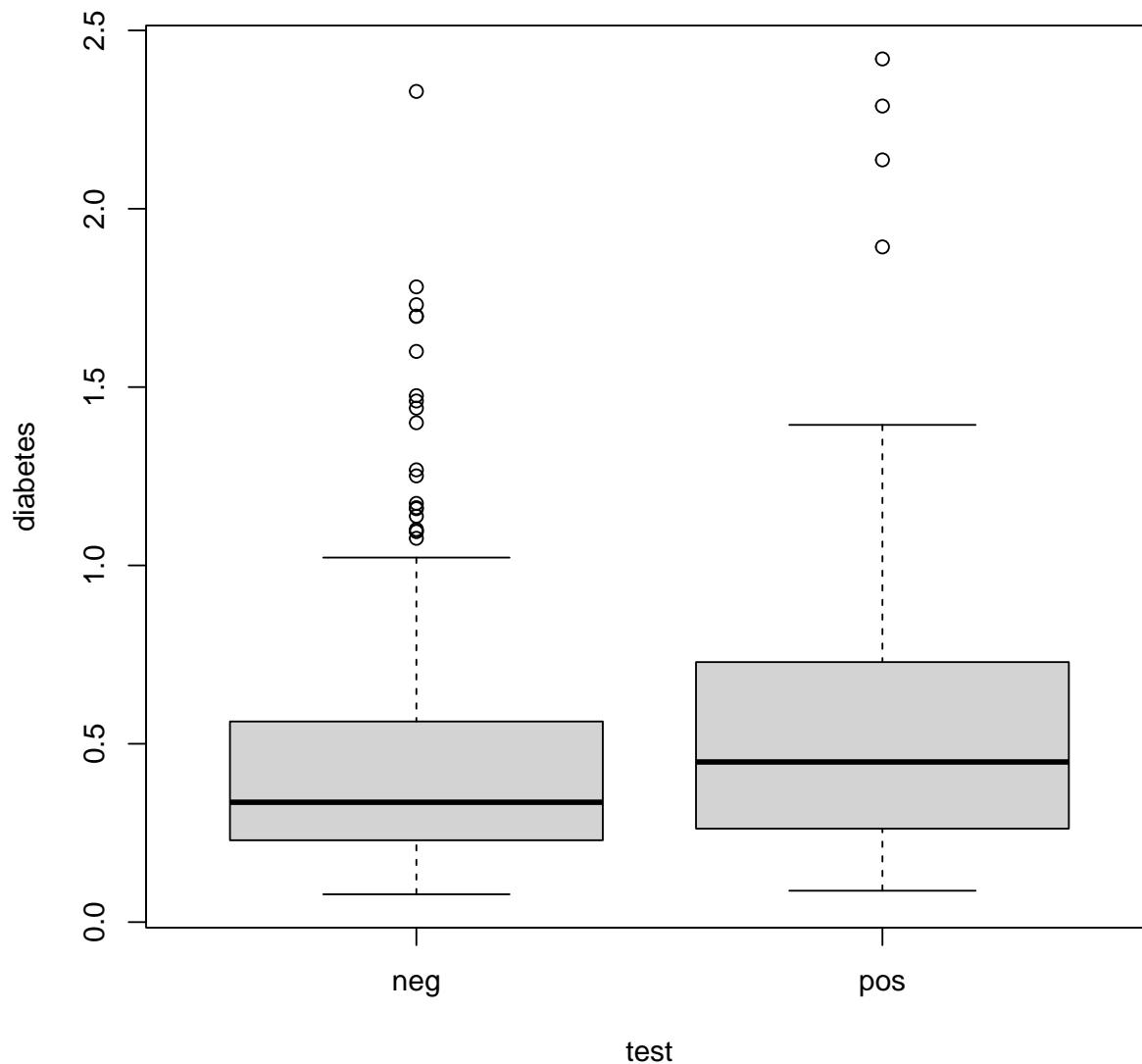
- Scatterplot of diabetes vs. diastolic, i.e., 'y vs x' so-to-speak'

```
> plot(diabetes ~ diastolic, data = pima)
```



- Boxplots. 'y vs x' when 'x' is discrete (categorical or factor)

```
> plot(diabetes ~ test, data = pima)
```



- See [Far14, §1.2] for more graphical summaries.

- TAKE HOME MESSAGE: ALWAYS CHECK YOUR DATA (graphically and/or numerically)!

Saving/Loading Your Results

- Do you want to **save** what we've created?

```
> ls()  
[1] "pima"    "qtiles"  
  
> ## See help(save) and help(load) (or ?save and ?load).  
> save.image() ## <-- saves all objects in .GlobalEnv to file .RData by default  
> save(list=c("pima"), file="yourspecialfile.RData")  
> rm(list=ls()) ## erase everything in .GlobalEnv! careful with this!  
> load(file="yourspecialfile.RData") ## good thing you saved!
```

Clean Up

- Do you want to ‘clean up?’ (You could merely quit and restart without removing objects or without detaching components from search path.)

```
> ls()  
  
[1] "pima"    "qtiles"  
  
> rm("pima")  
> ## rm(list=ls()) ## <-- careful!!  
> ls()  
  
[1] "qtiles"  
  
> search()  
  
[1] ".GlobalEnv"      "package:faraway"   "package:knitr"  
[4] "ESSR"            "package:stats"    "package:graphics"  
[7] "package:grDevices" "package:utils"    "package:datasets"  
[10] "package:methods"  "Autoloads"       "package:base"  
  
> detach(package:faraway)  
> search()  
  
[1] ".GlobalEnv"      "package:knitr"     "ESSR"  
[4] "package:stats"    "package:graphics" "package:grDevices"  
[7] "package:utils"    "package:datasets" "package:methods"  
[10] "Autoloads"       "package:base"
```

1.4 When to Use Linear Modeling

- READ [Far14, §1.3]

- We have a number of observations of each of several random variables somehow distinguished and denoted as (Y, \mathbf{X}) , where $\mathbf{x}^T = (x_1, \dots, x_k)$. (See the discussion of random variables and random vectors in §A.7 & B.3.)

- Y : response, output (machine learning), outcome, dependent variable. Often continuous (Def. A.15). Always numeric, not categorical (Def. A.13) in this class, INF 511. See INF 504 and 512 for categorical ‘ Y ’.
 - x_j : predictor, input (machine learning), explanatory variable, independent variable, regressor, covariate, factor (categorical or qualitative variable), exposure, co-morbidity, feature,.... (In note chapter 5, we will eventually discuss why we typically view predictors as known, hence denoting predictors in lowercase.)
- **NOTE:** Your textbook’s author (sometimes) uses p to denote the number of “ p ” redictor variables. Here, we use k instead to denote the number of predictors (“ k ” ovariates?). (We will see that our use of p will denote the number of “ p ” arameters in our linear regression model; $p = k + 1$ if we include an intercept (usually), else $p = k$). More later.
 - Simple linear regression, $k = 1$ (one predictor/input)
 - Multiple linear regression, $k > 1$ (multiple predictors/inputs)
 - Multivariate linear regression, more than one response/output, Y (numeric vector response, \mathbf{Y}), simple ($k = 1$ predictor) or multiple ($k > 1$ predictors)
 - Analysis of Variance (ANOVA; all predictors/inputs are factors)
 - Analysis of Covariance (ANCOVA mix of numeric and factor predictors/inputs)
 - Classification in INF 504 and INF 512: like here, in INF 511, we will model mean (‘regression’) functions of predictors/inputs, but for numerically encoded categorical variables.
 - Regression and classification are known as supervised learning methods in machine learning. Some people just call these methods ‘regression’ methods.

- Refer back to **pima** data set for concrete examples. (in class)

Objectives of Regression Analysis

While we focus on linear modeling, much of this section applies to general regression modeling, **linear or not**. Most of us will see non-linear models in INF 512, with some in INF 504, too.

1. **Prediction** of future or unseen response/output, Y , given specified values of predictors/inputs, x . We will spend a **relatively small amount of time** on prediction in INF 511. Much of machine learning, on the other hand, is pre-occupied with prediction (or estimation of the mean of such responses).
 2. **Explanation:** We seek to somehow assess the effect of or relationship between, “explanatory” (predictor) variables, x , and the response, Y . Can we infer causality? We will spend a **lot of our time** here. (Supervised) Machine learning typically is relatively unconcerned with what’s inside the black box (i.e., model/algorithm of the relationship) used for prediction.
- **Attribution.** In the spirit of [Efr20], we may include the goal of attribution, too, which refers to determining which inputs contribute to the output, i.e., which inputs are practically and statistically significantly related to the output. ([Efr20] gives a taxonomy of prediction (of outputs), estimation (of means) and attribution (to inputs).)

- The relationship we seek to use for prediction or explanation may come from
 - **theory**, or from a

- **recurring phenomenon** seen in similar data, or from a model that we create
 - **empirically**, from the data at hand.
- [HTF01, §2.4] provide a bit of theoretical motivation for developing models of a functional relationship between inputs and output. What are we aiming at? Why? We give a bit more in the (unnumbered) Introduction to note chapter 2.

1.5 History

See [Far14, §1.4] for the historical development of the term **regression**, which is often a misnomer nowadays, depending on the application.

Lecture 2

Estimation

Contents

Introduction	26
2.1 Linear Model	29
(Non-)Linear or (Non-)Linear?	30
2.2 Matrix Representation	32
2.3 Estimating β	33
2.4 Least Squares Estimation	35
Properties of the LS Estimator of Regression Model Parameters	36
Connections: Method of Maximum Likelihood	38
Multi-variate Normal Distribution	38
Special Case: Traditional Normal Linear Model	39
Normal Likelihood	39
Method of Maximum Likelihood	40
2.5 Examples of Calculating $\hat{\beta}$	41
2.6 Example: Galapagos Island Biogeography	41
2.7 QR Decomposition	47
2.8 Gauss-Markov Theorem	47
2.9 Goodness of Fit	49
2.10 Identifiability	55
2.11 Orthogonality	56
Example: Chemical Odor	57

Introduction

(Notice, this section is not numbered. So, as mentioned, it has no counterpart in [Far14, Chap. 2]. Value added! Cha-ching!)

- **Goals.** As we mentioned briefly in §1.4:

(1) **Predict.** We want to use a (vector of) predictor(s)/input(s), \mathbf{x} , to predict y (e.g., use image characteristics to predict (classify) car, motorcycle, pedestrian, other, etc.; INF 504) or

(2) **Explanation.** And, we want to assess (explain) the relationship between y and \mathbf{x} .

(We also mentioned **attribution**, but we skip it. More in INF 504 with variable selection/importance.)

- **Envision a Function.** Thus, we might consider looking for a function, f , of inputs, such that, loosely speaking, for the moment,

$$y \approx f(\mathbf{x})$$

(ideally but practically unobtainably equal due to ‘noise’ or ‘error’ in observations), with the goal of using such f to help us with (1) or (2).

- **Envision a Population (Distribution).** To help us make more precise what we mean by such a function, f , we consider briefly a relatively theoretical probabilistic perspective. In short, the random variable (vector),

$$(Y, \mathbf{X}^t),$$

or its (joint) distribution,

$$[y, \mathbf{x}],$$

represents the ‘(super-)population’ of all possible values of response values and covariates, $(y, \mathbf{x}^t)^t$, about which we may be interested for purposes (1) or (2). See Appendix B for more on the joint distribution of multiple random variables.

- **A Function Somehow “Good for the Population”.** We might gain insight into f by first asking ourselves,

“How would we determine f in the ideal situation in which we know the entire population of values for (Y, \mathbf{X}) ? ”

(Granted, (1) seems obviated in this hypothetical case (why?), and, in practice, we will only see a relatively small number of observations, $(y_i, \mathbf{x}_i), i = 1, \dots, n$ from a population.)

- **Choose f to Minimize Average Loss.** One popular notion of ‘goodness’ is to choose f to minimize

$$L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$$

(**squared error loss**), on average, over all values of y and \mathbf{x} in a population, i.e., find f to minimize expected loss, $E(L(Y, f(\mathbf{X})))$.

- **Conditional Mean Minimizes Squared Error Loss.** With this (squared-error loss) notion of optimality,

$$f(\mathbf{x}) = E(Y | \mathbf{x})$$

[HTF01, §2.4], where $E(Y | \mathbf{x})$ is the mean of the conditional distribution of $Y | \mathbf{x}$, and is called the **regression function** our **target**, at least in the theoretical world where we know the entire population. Details omitted. (Other theoretical targets arise from other goodness criteria; perhaps a bit more of this in INF 504.)

- **Don’t Worry.** Don’t let the short-hand notation scare you. It just means that we’re talking about a random variable (response/output) whose (generally, distribution, or, specifically, mean) value depends on covariates/inputs, \mathbf{x} , and f is the mean (i.e., ‘expected value’) of this random variable in this idealistic case of averaging over the whole population of values of Y associated with a value of \mathbf{x} ; after averaging over the values of Y , we get some function of inputs, \mathbf{x} . See §A.8 and B.3 for more on about the mean (i.e., expectation) of (a function of) a random

variable/vector, and see §B.6 and B.7 for more on conditional (and joint (multivariate) and marginal) distributions and conditional distribution means.

- **Goodness in Practice: A Sample, Additive Error Model & Least Squares.** Of course, in practice, we do not know the entire population distribution or the conditional distribution(s), $Y | \mathbf{x}$, nor their mean(s), their regression function(s). Thus, with only a sample of values,

$$(y_i, \mathbf{x}_i), i = 1, \dots, n,$$

how might we choose f ? Motivated by the above theoretical consideration of finding f via minimization of expected squared error loss over the entire population, we define ‘errors’ $\epsilon_i = Y_i - f(\mathbf{x}_i)$, $i = 1, \dots, n$, to get the very popular, additive error regression model

$$Y_i = f(\mathbf{x}_i) + \epsilon_i,$$

and we seek to choose f that minimizes the sum (or average if you want) of squared errors (losses)

$$\sum_{i=1}^n (y_i - f(\mathbf{x}_i))^2,$$

now practically over the n sample observations, not theoretically over the entire population.

- **Assumptions or Structure.** Generally, the problem of finding f , in this case, is **ill-posed** without additional assumptions or model structure. In particular, we can envision any number of such functions, $f(\mathbf{x})$, such that $y_i = f(\mathbf{x}_i)$ exactly, $i = 1, \dots, n$, without error. (We can draw an uncountable number of curves through the data points (assuming a unique output per input).)
- **Linear Model Structure.** In this class (INF 511), we will assume that f (i.e., that the regression function, $E(Y | \mathbf{x})$, under the above, theoretical notion of optimality (minimize expected squared error (loss))) has the

form of a linear model. That is, we will specify a linear model of the regression function, and will use data to estimate the model (its parameters) or to predict unobserved response values given observed covariates. (We will perform model **diagnostics** to help justify model choice.)

- **INF 504/512.** Much of INF 504 involves linear models. In INF 512 (and a bit of INF 504), we will consider non-linear models.

2.1 Linear Model

Traditional Linear Model. The traditional linear model of n observation pairs, (y_i, \mathbf{x}_i) , $i = 1, \dots, n$, is a model for the conditional distribution, $[y_i | \mathbf{x}_i]$, arising from

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \epsilon_i,$$

([Far14, §2.2]), i.e.,

$$Y_i = \mathbf{x}_i^t \boldsymbol{\beta} + \epsilon_i,$$

where

- **Response.** Y_i is the response (**output**) random variable associated with observation y_i ,
- **Predictors.** $\mathbf{x}_i = (1, x_{i1}, \dots, x_{ik})^t$ is a $p = (k + 1)$ -vector with 1 and k observed predictors (**covariates, inputs**).
- **Parameters.** $\boldsymbol{\beta} = (\beta_0, \dots, \beta_k)^t$ is a $p = (k + 1)$ -vector of unknown parameters ('**weights**' in machine learning), with β_0 often called the **intercept** ('**bias**' in machine learning, not to be confused with statistical bias)

- **Random Errors.** The ϵ_i are random errors with expectation (mean) $E(\epsilon_i) = 0$, often with further specification, which we give later [Far14, Chap. 2].

- **Model of the Target Regression Function (Regression Model).** Thus, under the assumption of a linear model for the conditional mean, we have

$$f(\mathbf{x}, \boldsymbol{\beta}) = \mathbf{x}^t \boldsymbol{\beta},$$

i.e., we have assumed a linear model form for the regression function $f(\mathbf{x}) = E(Y | \mathbf{x})$, now indicating dependence of the model on unknown regression model parameters in $\boldsymbol{\beta}$.

- See §B.2.4 for the transpose operator, t , and, more generally, see Appendix B for more on vectors (and matrices)).

- Figure 2.1 illustrates a linear model of the regression function $E(Y | \mathbf{x}) = 10 + 2x_1 + 5x_2$ (Source: [KNNL05]).

(Non-)Linear or (Non-)Linear?

- **Linear in the Parameters.** Note that **linear** in **linear model** technically refers to how the parameters enter the model of the regression function (conditional mean) of Y_i : the model is a **linear combination** of the parameters, β_j , with “coefficients” x_{ij} , $j = 0, \dots, k$ ($x_{i0} = 1$ for intercept), i.e.,

$$\mathbf{x}_i^t \boldsymbol{\beta}.$$

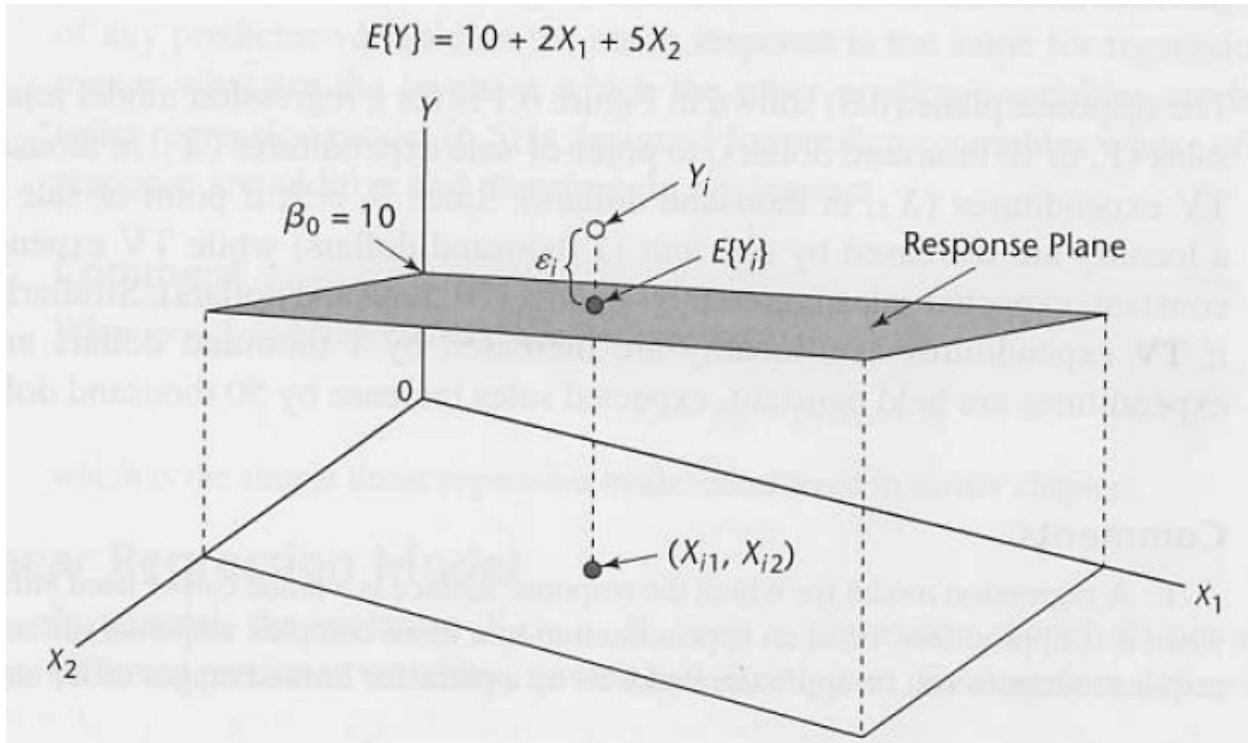


Figure 2.1: Bivariate regression model (Source: [KNNL05]).

(See §A.10 for more on linear combinations; β_j here plays the role of Y_i there, and the x_{ij} here play the role of the c_i there.) We will make much of inference for linear combinations of β , later.

- **Typically Non-Linear Relationships.** This does **not** mean that linear models cannot capture **non-linear relationships between inputs and the output**—far from the truth, as we will appreciate more, later.
- **Example.** For example, the first model below is a linear model, in β , and a non-linear model, in x , while the latter two models are non-linear models, in both β and x .

$$\begin{aligned}
 Y_i &= \beta_0 + \beta_1 \log(x_{i1}) + \epsilon_i, \\
 Y_i &= \beta_0 + \beta_1 x_{i1} + \beta_2 \log(x_{i2})^{\beta_3} + \epsilon_i, \\
 Y_i &= \frac{\beta_1 x_{i1}}{\beta_2 + x_{i2}} + \epsilon_i \quad (\text{Michaelis-Menton mean model})
 \end{aligned}$$

- **INF 504/512.** We'll see non-linear models (in the parameters β (and in the covariates)) in INF 512 and INF 504.

2.2 Matrix Representation

- The linear model can be written in **matrix form** as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

i.e.,

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

- Again, see Appendix B for a discussion of matrices and vectors, especially if these concepts are new to you.
- \mathbf{X} now represents a matrix of known predictors/inputs, not a random vector.

2.3 Estimating β

- **Back to the Function.** Now, how do we find/estimate/learn $f(\mathbf{x}) = E(Y | \mathbf{x})$? Now, with our assumption of the linear model form, i.e., with this imposition of linear model structure, $f(\mathbf{x}, \boldsymbol{\beta}) = \mathbf{x}^t \boldsymbol{\beta}$, the search for $f(\mathbf{x})$ is (usually) no longer ill-posed, and the above question translates to how do we estimate $\boldsymbol{\beta}$?
- **Least Squares Geometry.** There are several different estimation methods, least squares already being suggested, above, with more detail below. The figure on [Far14, p. 15] is another way to view our linear regression model (in addition to Figure 2.1, shown previously) that suggests conceptually one way to find $\boldsymbol{\beta}$ via least squares. See our Figure 2.2, nearby.
- **LS Estimator/Estimate of $\boldsymbol{\beta}$ & Fitted Value.** In short, we seek the value $\hat{\boldsymbol{\beta}}$ of parameter $\boldsymbol{\beta}$ that makes the fitted value vector (or **predicted value** vector)

$$\hat{\mathbf{y}} \equiv \mathbf{X}\hat{\boldsymbol{\beta}} = \sum_j \hat{\beta}_j \mathbf{x}_j$$

as close as possible to the observed output vector \mathbf{y} .

- **Residual.** Thus, the shortest vector between the observed and fitted vectors is

$$\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}},$$

and is called the residual vector.

- **Orthogonal Decomposition.** And, $\hat{\mathbf{y}}^t \hat{\boldsymbol{\epsilon}} = 0$; i.e., the fitted (orthogonally projected) vector is perpendicular to the residual vector, and the fitted and residual vector sum to \mathbf{y} , i.e.,

$$\mathbf{y} = \hat{\mathbf{y}} + \hat{\boldsymbol{\epsilon}},$$

thus giving an orthogonal decomposition of \mathbf{y} (like Pythagorus); see again the figure of LS geometry on [Far14, p. 15]

- **Lower Dimensional Representation (Plus Residual).** Thus, we approximate an n -dimensional observation vector, \mathbf{y} , as the sum of a vector $\hat{\mathbf{y}}$, in p dimensions (why p dimensional?), plus a vector $\hat{\boldsymbol{\epsilon}}$ in the remaining $(n - p)$ dimensions (some detail omitted, of course; more discussion in class); see top of [Far14, p. 16].

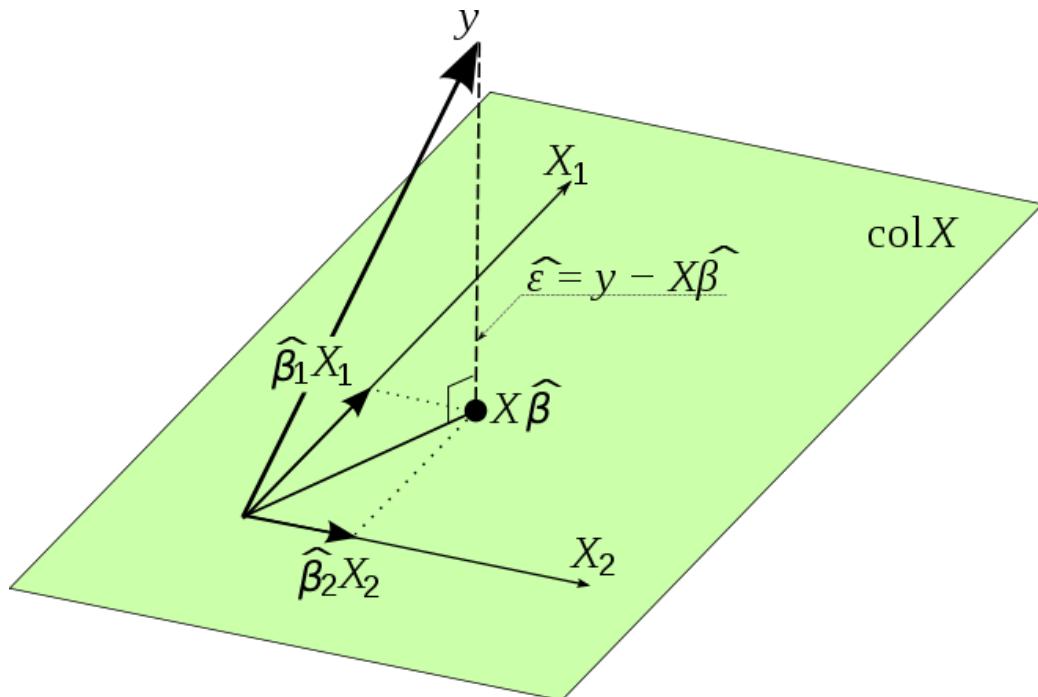


Figure 2.2: Least squares geometry (Source: forgotten).

- **A Third Figure.** We've pictured the linear model 2 ways. If, in Figure

2.1, x_1 is x and x_2 is x^2 , then we see a non-linear, parabolic relationship between the (mean of) y and x . Must I illustrate this?

2.4 Least Squares Estimation

- **LS Objective.** Thus, according to our discussion, **we seek the value of β to minimize the squared length of the error vector**

$$\sum_i \epsilon_i^2 = \boldsymbol{\epsilon}^t \boldsymbol{\epsilon} = (\mathbf{y} - \mathbf{f}(\mathbf{x}; \boldsymbol{\beta}))^t (\mathbf{y} - \mathbf{f}(\mathbf{x}; \boldsymbol{\beta})) = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^t (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}).$$

- **Gradient.** Minimizing the objective (in $\boldsymbol{\beta}$) is a calculus problem. Differentiating (computing the **gradient** of f) with respect to $\boldsymbol{\beta}$, we get

$$\mathbf{X}^t (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

(which, incidentally, is an example of an **(linear in the data) estimating function for β** . Estimating functions/equations are widely used in statistics. More in INF 512.).

- **Normal Equations.** Setting the above function of $\boldsymbol{\beta}$ equal to zero leads to the normal equations

$$(\mathbf{X}^t \mathbf{X})\boldsymbol{\beta} = \mathbf{X}^t \mathbf{y}$$

which can be solved for $\boldsymbol{\beta}$ as long as \mathbf{X} is **full rank** so that $(\mathbf{X}^t \mathbf{X})$ is invertible (non-singular). See Appendix B for a discussion of linear (in)dependence of a set of vectors and rank of a matrix.

- **LS Estimator/Estimate of β , Hat Matrix & Fitted Vector (Algebraically Now).** Thus, we get

$$\begin{aligned}\hat{\boldsymbol{\beta}} &= (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y} \\ \mathbf{X}\hat{\boldsymbol{\beta}} &= \mathbf{X}(\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y} \\ \hat{\mathbf{y}} &= \mathbf{H}\mathbf{y},\end{aligned}$$

where

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t$$

is called the **hat matrix**, which projects \mathbf{y} orthogonally onto the space spanned by the columns of \mathbf{X} to get $\hat{\mathbf{y}}$; informally, it puts a hat on y . (Again: see the figure [Far14, p. 15]).

- **Residual Vector (again).** Thus, we can write the residual vector as

$$\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{H}\mathbf{y} = (\mathbf{I} - \mathbf{H})\mathbf{y}.$$

- **Residual Sum-of-Squares (RSS).** And, we define the residual sum-of-squares (RSS) as

$$\sum_i (y_i - \hat{y}_i)^2 = \sum_i \hat{\epsilon}_i^2 = \hat{\boldsymbol{\epsilon}}^t \hat{\boldsymbol{\epsilon}} = \mathbf{y}^t (\mathbf{I} - \mathbf{H})^t (\mathbf{I} - \mathbf{H}) \mathbf{y} = \mathbf{y}^t (\mathbf{I} - \mathbf{H}) \mathbf{y},$$

(the last equality following from \mathbf{H} being symmetric and **idempotent** ($\mathbf{H}^2 = \mathbf{H}$), i.e., it's a (perpendicular) **projection matrix**; see, e.g., [Chr02, Theorem B.33] (now in its 4th edition) (($\mathbf{I} - \mathbf{H}$) projects $\boldsymbol{\epsilon}$ onto the orthogonal complement of the space spanned by the columns of \mathbf{X})).

Properties of the LS Estimator of Regression Model Parameters

- We can use results in Appendix B to get some properties of the least squares estimator, $\hat{\boldsymbol{\beta}}$. (While least squares may be used by itself, we assume underlying random variables / distributions. The quantities here are ‘theoretical’ ((super-)population) quantities to be estimated via data.)
- **Mean (Unbiased).** Because we assume zero mean error, $E(\boldsymbol{\epsilon}) = \mathbf{0}$, we have mean

$$E(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta} \quad (\text{unbiased}).$$

- **Variance.** Assuming an **error variance**,

$$\text{Var}(\boldsymbol{\epsilon}) = \boldsymbol{\Sigma},$$

we have variance

$$\begin{aligned}\text{Var}(\hat{\beta}) &= \text{Var}((\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y}) = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \text{Var}(\mathbf{y}) \mathbf{X} (\mathbf{X}^t \mathbf{X})^{-1} \\ &= \text{Var}((\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y}) = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \Sigma \mathbf{X} (\mathbf{X}^t \mathbf{X})^{-1}\end{aligned}$$

(by results in Appendix B).

- **Assuming Constant & Uncorrelated Error Variance.** If we assume, in particular,

$$\Sigma = \sigma^2 \mathbf{I},$$

then **Variance** is

$$\text{Var}(\hat{\beta}) = \sigma^2 (\mathbf{X}^t \mathbf{X})^{-1}.$$

In particular,

$$\text{Var}(\hat{\beta}_j) = \sigma^2 (\mathbf{X}^t \mathbf{X})_{jj}^{-1},$$

where we use

$$(\mathbf{X}^t \mathbf{X})_{jj}^{-1}$$

to denote the j th diagonal element of $(\mathbf{X}^t \mathbf{X})^{-1}$ (perhaps with some notational fix-up for starting at $j = 0$ in β_0).

- **Standard Error.** Thus, the standard error of $\hat{\beta}_j$ is

$$\text{se}(\hat{\beta}_j) = \sigma \sqrt{(\mathbf{X}^t \mathbf{X})_{jj}^{-1}}.$$

Note, in Appendix B, we defined standard deviation as the square root of variance. Standard error is a special term used to denote the standard deviation of a random quantity used to estimate something, e.g., a mean or parameters of a mean model.

- **Error Variance Estimator.** Standard results (details omitted) give the **expected RSS** (residual sum-of-squares),

$$E(\hat{\epsilon}^t \hat{\epsilon}) = \sigma^2(n - p).$$

This suggests the **method of moments (MOM) estimator** of σ^2 ,

$$\hat{\sigma}^2 = \frac{\hat{\epsilon}^t \hat{\epsilon}}{(n - p)} = \frac{\sum_i (y_i - \hat{y}_i)^2}{(n - p)} = \frac{RSS}{(n - p)},$$

which is often called the (estimated) **mean squared error (MSE)**.

- **df.** We will call $(n - p)$ **residual degrees of freedom** and p the **model degrees of freedom**.
- **(Estimated) Standard Error.** This leads to the (estimated) standard error

$$\widehat{se}(\widehat{\beta}_j) = \widehat{\sigma} \sqrt{(\mathbf{X}^t \mathbf{X})_{jj}^{-1}}.$$

- **Potato, Potato.** R calls $\widehat{\sigma}$ then **residual standard error**, which is not uncommon. (We might call it an estimated standard deviation of the errors...not common at all.)
- **Normality.** Later, we will make a normality assumption for the errors, ϵ_i , which will lead to $\widehat{\beta}$ being normal with above mean and variance. This will lead to typical inference procedures for β —tests, p-values, confidence intervals.

Connections: Method of Maximum Likelihood

- The least squares (LS) estimator, $\widehat{\beta}$, of the linear regression (mean) model parameter, β , is the same as the normal maximum likelihood (ML, not 'm'achine 'l'earning) estimator (MLE).
- Our MSE estimator, $\widehat{\sigma}^2$, of the error variance, σ^2 is *not* the same as the MLE, though differences diminish as sample size, n , increases.

Multi-variate Normal Distribution

For any n -vector of real values, μ , and $n \times n$ symmetric positive definite (what's that?) matrix of real values, Σ , the n -vector \mathbf{Y} has a(n) (n -variate or multi-variate) normal distribution (of dimension n) with mean μ

and variance(co-variance matrix) Σ , denoted

$$\mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma),$$

if the pdf (see Appendices A & B) of \mathbf{Y} is

$$[\mathbf{y} | \boldsymbol{\mu}, \Sigma] = (2\pi)^{-\frac{n}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{y} - \boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{y} - \boldsymbol{\mu})\right).$$

Special Case: Traditional Normal Linear Model

In the special case of a linear regression (mean) function and uncorrelated errors with constant variance, $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$, $\Sigma = \sigma^2 \mathbf{I}$, as in INF 511, the pdf may be written as,

$$[\mathbf{y} | \mathbf{X}, \boldsymbol{\beta}, \sigma^2] = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mathbf{x}_i^t \boldsymbol{\beta})^2\right),$$

or, observation-wise (marginally, for each y_i) in this special case,

$$[y_i | \mathbf{x}_i, \boldsymbol{\beta}, \sigma^2] = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2\sigma^2} (y_i - \mathbf{x}_i^t \boldsymbol{\beta})^2\right).$$

Normal Likelihood

A normal likelihood function is a normal pdf viewed as a function of the parameters (usually) given the data values $\mathbf{Y} = \mathbf{y}$ (and the covariates). In our special case,

$$L(\boldsymbol{\beta}, \sigma^2 | \mathbf{y}, \mathbf{X}) = (2\pi\sigma^2)^{-\frac{n}{2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \mathbf{x}_i^t \boldsymbol{\beta})^2\right).$$

Method of Maximum Likelihood

- **Often a Calculus Problem.** Like the method of least squares (LS), the method of maximum likelihood (ML, not ‘m’achine ‘l’earning) is a very popular way to obtain parameter estimators: choose the value of the parameter that maximizes the value of the likelihood function, usually a calculus problem of computing the **gradient** of (often the natural log of) the likelihood, wrt the parameter (vector), setting the result equal to zero and solving for the parameter.
- **ML same as LS in Some Cases.** In the case of a normal likelihood (linear or non-linear mean model), the MLE of β is obtained by minimizing the error sum-of-squares. That is, in the case of a normal likelihood the methods of LS and ML lead to the same estimator of the regression model parameters; for the special case of a normal linear model with uncorrelated errors and constant variance, we have

$$\hat{\beta} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{Y}$$

(or estimate $\hat{\beta} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{y}$).

- **Score Function.** ($S(\beta) = \mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)$) is a particular instance of a score function (of β), which is more generally obtained as the gradient of the (natural) log of a likelihood. More on score functions in INF 512.)
- **MSE/MOM Estimator Instead of MLE of Error Variance.** Typically, we do not use the MLE of σ^2 ($1/n \sum_i (y_i - \mathbf{x}_i^t \hat{\beta})^2$) but use MSE (' $1/(n-p)$ version'), instead,

$$\begin{aligned}\hat{\sigma}^2 &= \frac{1}{(n-p)} \sum_i (y_i - \mathbf{x}_i^t \hat{\beta})^2 \\ &= \frac{RSS}{n-p}\end{aligned}$$

where p is the dimension of the regression model parameter, β . (Again, this correspondence between LS and ML occurs with linear or non-linear

regression models of the mean of a normal distribution with constant variance; we'll see it again in INF 512.)

2.5 Examples of Calculating $\hat{\beta}$

Read [Far14, §2.5]. We go directly to the next section.

2.6 Example: Galapagos Island Biogeography

We follow the Galapagos island biogeography example of [Far14, §2.6]. Read it. The presentation here is comparatively terse, leaving a lot for us to discuss in class.

```
> library(faraway)
> data(gala, package="faraway")
> head(gala[,-2])

      Species Area Elevation Nearest Scruz Adjacent
Baltra      58  25.09       346     0.6    0.6    1.84
Bartolome   31   1.24       109     0.6   26.3  572.33
Caldwell     3   0.21       114     2.8   58.7    0.78
Champion    25   0.10        46     1.9   47.4    0.18
Coamano      2   0.05       77     1.9    1.9  903.82
Daphne.Major 18   0.34       119     8.0    8.0    1.84

> ## Relatively automated LS estimation of beta and then some:
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+             data=gala)
> (lmodsum<- summary(lmod))
```

Call:

```
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
  data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-----	----	--------	----	-----

```

-111.68 -34.90 -7.86 33.46 182.58

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 7.06822   19.15420   0.37    0.7154    
Area        -0.02394   0.02242  -1.07    0.2963    
Elevation    0.31946   0.05366   5.95 0.0000038 ***  
Nearest      0.00914   1.05414   0.01    0.9932    
Scruz        -0.24052   0.21540  -1.12    0.2752    
Adjacent     -0.07480   0.01770  -4.23    0.0003 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 61 on 24 degrees of freedom
Multiple R-squared:  0.766, Adjusted R-squared:  0.717 
F-statistic: 15.7 on 5 and 24 DF,  p-value: 6.84e-07

```

```

> ## Using our matrix knowhow (Appendix B) to compute the LS betahat
> ## (somewhat conceptual if not computationally efficient; see QR
> ## decomposition in textbook section 2.7)
> x <- model.matrix(~ Area + Elevation + Nearest + Scruz + Adjacent,gala)
> y <- gala$Species
> xtxi <- solve(t(x) %*% x)
> as.vector(betahat<- xtxi %*% t(x) %*% y)

[1] 7.068221 -0.023938 0.319465 0.009144 -0.240524 -0.074805

> as.vector(betahat<- solve(crossprod(x,x), crossprod(x,y)))

[1] 7.068221 -0.023938 0.319465 0.009144 -0.240524 -0.074805

> ## Our results are the same as computed by the lm function (of course):
> coef(lmod)

(Intercept)          Area          Elevation        Nearest         Scruz        
7.068221     -0.023938     0.319465     0.009144     -0.240524  
Adjacent
-0.074805

```

```

> ## Lots of results here.
> ## We prefer to use existing functions to get at results
> ## (in following chunks).
> names(lmod)

[1] "coefficients"   "residuals"      "effects"       "rank"
[5] "fitted.values"  "assign"        "qr"            "df.residual"
[9] "xlevels"         "call"          "terms"         "model"

> is.list(lmod)

[1] TRUE

> names(lmodsum)

[1] "call"           "terms"          "residuals"     "coefficients"
[5] "aliased"        "sigma"          "df"            "r.squared"
[9] "adj.r.squared"  "fstatistic"    "cov.unscaled"

> is.list(lmodsum)

[1] TRUE

```

```

> ## fitted values vector yhat
> as.vector(yhat<- x%*%betahat) ## <-- `by hand'

[1] 116.72595 -7.27315 29.33066 10.36427 -36.38392 43.08771
[7] 33.91967 -9.01899 28.31420 30.78594 47.65649 96.98960
[13] -4.03328 64.63380 -0.49718 386.40356 88.69454 4.03723
[19] 215.67949 150.47538 35.07581 75.55312 206.95188 277.67632
[25] 261.41641 85.37649 195.61663 49.80509 52.93573 26.70057

> fitted(lmod) ## <-- more automatically

      Baltra    Bartolome    Caldwell    Champion    Coamano
      116.72595     -7.27315     29.33066     10.36427    -36.38392
Daphne.Major Daphne.Minor    Darwin      Eden    Enderby
      43.08771     33.91967    -9.01899     28.31420     30.78594
      Espanola    Fernandina    Gardner1   Gardner2    Genovesa
      47.65649     96.98960    -4.03328     64.63380    -0.49718
      Isabela     Marchena    Onslow      Pinta     Pinzon
      386.40356     88.69454     4.03723    215.67949    150.47538
      Las.Plazas    Rabida SanCristobal SanSalvador SantaCruz

```

35.07581	75.55312	206.95188	277.67632	261.41641
SantaFe	SantaMaria	Seymour	Tortuga	Wolf
85.37649	195.61663	49.80509	52.93573	26.70057

```

> ## residual vector epsilonhat
> as.vector(epshat<- y - yhat) ## `by hand'

```

[1]	-58.7259	38.2732	-26.3307	14.6357	38.3839	-25.0877
[7]	-9.9197	19.0190	-20.3142	-28.7859	49.3435	-3.9896
[13]	62.0333	-59.6338	40.4972	-39.4036	-37.6945	-2.0372
[19]	-111.6795	-42.4754	-23.0758	-5.5531	73.0481	-40.6763
[25]	182.5836	-23.3765	89.3834	-5.8051	-36.9357	-5.7006

```
> residuals(lmod) ## <-- more automatically
```

Baltra	Bartolome	Caldwell	Champion	Coamano
-58.7259	38.2732	-26.3307	14.6357	38.3839
Daphne.Major	Daphne.Minor	Darwin	Eden	Enderby
-25.0877	-9.9197	19.0190	-20.3142	-28.7859
Espanola	Fernandina	Gardner1	Gardner2	Genovesa
49.3435	-3.9896	62.0333	-59.6338	40.4972
Isabela	Marchena	Onslow	Pinta	Pinzon
-39.4036	-37.6945	-2.0372	-111.6795	-42.4754
Las.Plazas	Rabida	SanCristobal	SanSalvador	SantaCruz
-23.0758	-5.5531	73.0481	-40.6763	182.5836
SantaFe	SantaMaria	Seymour	Tortuga	Wolf
-23.3765	89.3834	-5.8051	-36.9357	-5.7006

```
> ## Are fitted and residual orthogonal?
```

```
> sum(yhat * epshat)
```

[1] 1.8283e-10

> ## Do they sum to the observed vector (of course)?

```
> as.vector(y - (yhat + epshat))
```

```
> ## residual sum of squares RSS
```

```
> sum(epshat^2) ## `by hand'
```

```
[1] 89231

> deviance(lmod) ## <-- more automatically

[1] 89231

> ## residual df
> (n<- length(na.omit(y))) ## num. of obs.

[1] 30

> (p<- dim(x)[2]) ## <-- num. of reg. model parameters

[1] 6

> (n-p) ## <-- residual df

[1] 24

> df.residual(lmod) ## <-- more automatically

[1] 24

> ## MSE (MOM estimator of sigma2) and sqrt thereof
> ## ((estimated) residual standard error)
> (sigmahat<- sum(epshat^2)/(n-p)) ## <-- `by hand'

[1] 3718

> (sigmahat<- deviance(lmod)/df.residual(lmod)) ## <-- more automatically

[1] 3718

> sqrt(sigmahat)

[1] 60.975

> ## estimated variance (matrix)
> (varbetahat<- sigmahat * xtxi) ## <-- `by hand'

(Intercept)      Area   Elevation   Nearest
```

```
(Intercept) 366.883294 0.14047404 -0.58073853 -0.8696442
Area          0.140474  0.00050276 -0.00096430  0.0048111
Elevation     -0.580739 -0.00096430  0.00287970 -0.0131964
Nearest        -0.869644  0.00481107 -0.01319645  1.1112026
Scruz         -1.398067 -0.00018267  0.00114544 -0.1420666
Adjacent      0.085879  0.00017178 -0.00060984  0.0052971
                  Cruz   Adjacent
(Intercept) -1.39806717 0.08587895
Area          -0.00018267 0.00017178
Elevation     0.00114544 -0.00060984
Nearest        -0.14206665 0.00529710
Scruz         0.04639813 -0.00072811
Adjacent      -0.00072811 0.00031330
```

```
> (varbetahat<- vcov(lmod)) ## <-- more automatically
```

	(Intercept)	Area	Elevation	Nearest
(Intercept)	366.883294	0.14047404	-0.58073853	-0.8696442
Area	0.140474	0.00050276	-0.00096430	0.0048111
Elevation	-0.580739	-0.00096430	0.00287970	-0.0131964
Nearest	-0.869644	0.00481107	-0.01319645	1.1112026
Scruz	-1.398067	-0.00018267	0.00114544	-0.1420666
Adjacent	0.085879	0.00017178	-0.00060984	0.0052971
	Cruz	Adjacent		
(Intercept)	-1.39806717	0.08587895		
Area	-0.00018267	0.00017178		
Elevation	0.00114544	-0.00060984		
Nearest	-0.14206665	0.00529710		
Scruz	0.04639813	-0.00072811		
Adjacent	-0.00072811	0.00031330		

```
> ## (estimated) standard errors
```

```
> sqrt(diag(varbetahat))
```

	Intercept	Area	Elevation	Nearest	Scruz
Intercept	19.154198	0.022422	0.053663	1.054136	0.215402
Adjacent					
	0.017700				

2.7 QR Decomposition

The computations performed by `lm()` and related functions are not only **convenient**, they're relatively **computationally efficient and numerically stable**, too, compared to using our conceptually convenient formulas to compute in R, 'by hand,' above. See [Far14, §2.7] for the so-called QR matrix decomposition of \mathbf{X} and its use in least squares computation. We skip this section.

2.8 Gauss-Markov Theorem

- Why use the LS estimator of β ?
- In other words, how might we justify using it to our colleagues?
- Perhaps the geometry makes sense ([Far14, §2.3]).
- Perhaps we feel good about vanquishing (or, at least, minimizing) errors.
- The Gauss-Markov theorem gives more justification for the LS estimator of β .

- According to our linear model (without the normality assumption), we let
$$\mathbf{Y} = \mathbf{X}\beta + \epsilon \quad \epsilon \sim ?(\mathbf{0}, \sigma^2 \mathbf{I}).$$
- So, in particular, we assume **constant error variance** and **uncorrelated errors**.
- Further, we assume our **regression model is correct**, i.e., we assume $E(\mathbf{Y} | \mathbf{X}) = \mathbf{X}\beta$ is the "true unknown regression function," f (vector thereof), that we seek.

- With these assumptions, consider the LS estimator

$$\hat{\beta} = (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^t \mathbf{Y}.$$

- Then, $\hat{\beta}$ is uniquely **BLUE**: the Best Linear Unbiased Estimator of β .
- Best** in the following sense. Let

$$\beta = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_k \end{bmatrix}, \quad \hat{\beta} = \begin{bmatrix} \hat{\beta}_0 \\ \vdots \\ \hat{\beta}_k \end{bmatrix},$$

and let

$$\mathbf{c}^t = [c_0, \dots, c_k]$$

be **any** single row matrix of $p = k + 1$ constants. Then,

$$\text{Var}(\mathbf{c}^t \hat{\beta}) \leq \text{Var}(\mathbf{c}^t \hat{\beta}^*),$$

where $\hat{\beta}^*$ is any estimator you can create of the form $\hat{\beta}^* = \mathbf{a} + \mathbf{A}\mathbf{Y}$ with $E(\hat{\beta}^*) = \beta$. That is, if you consider any arbitrary linear (in \mathbf{Y}) unbiased estimator of β , then the variances of linear combinations of $\hat{\beta}$ are least as small as the variances of the same linear combinations of your estimator, $\hat{\beta}^*$.

- Linear** means the estimator is a linear function of the data, i.e., is of the form $\mathbf{a} + \mathbf{B}\mathbf{Y}$ (see §B.3.3 & B.3.4).
- Unbiased** means, as we have seen, $E(\hat{\beta}) = \beta$.
- Estimator** means it is a function of our (random) data vector \mathbf{Y} and, in particular, does not depend on β . (It would be unfortunate if it depended on the unknown quantity being estimated!)
- If, in addition, errors are normal, then $\hat{\beta}$ is the **maximum likelihood estimator** and is **BUE: Best Unbiased Estimator**, linear or otherwise.
- If errors are correlated or have unequal variance, then we may appeal to a generalized GM theorem, which leads us to **(estimated) generalized least squares ((E)GLS)** [Far14, §8.1].

- If errors are not normal, BLUE may be much worse than BUE. With long-tailed error distributions (relative to the normal distribution), we may consider robust estimators that are not linear in \mathbf{Y} [Far14, §8.4].
- If two or more observed predictor vectors (i.e., columns of \mathbf{X}) are highly correlated (**collinear**), then $\hat{\beta}$ will be highly variable, in which case we may consider slightly biased estimators, as in ridge regression ([Far14, Chap. 11]).

2.9 Goodness of Fit

- **Model RSS.** Consider the residual sum of squares for our regression model

$$RSS(\beta) = (\mathbf{Y} - \mathbf{x}^t \hat{\beta})^t (\mathbf{Y} - \mathbf{x}^t \hat{\beta}).$$

(Note, this is often called **sum of squared errors (SSE)**, and is the numerator of MSE , discussed earlier.)

- **Null Model RSS, aka TSS.** Now, consider the residual sum of squares of the **null model**, i.e., the regression (mean) model with only a single parameter, β_0 , and no predictors (other than $x_{i0} = 1$),

$$\begin{aligned} TSS &= RSS(\beta_0) = \sum_i (y_i - \hat{\beta}_0)^2 \\ &= \sum_i (y_i - \bar{y})^2 \\ &= (\mathbf{y} - \bar{y}\mathbf{1})^t (\mathbf{y} - \bar{y}\mathbf{1}), \end{aligned}$$

where TSS stands for **total sum of squares** (of the y_i , 'corrected for their (estimated) mean').

- (The LS estimate of β_0 in different regression models is generally different, of course!).

- **Decomposing Variability.** It can be shown that we can **decompose the total sums of squares** as

$$\sum_i (y_i - \bar{y})^2 = \sum_i (\hat{y}_i - \bar{y})^2 + \sum_i (y_i - \hat{y}_i)^2$$

$$TSS = (TSS - RSS) + RSS$$

total variability = var. due to regression + var. due to error

(Note, $(TSS - RSS)$ is often referred to as sum of squares due to regression (SSR), RSS as SSE , and TSS as SST , which gives $SST = SSR + SSE$.)

- **Graphical Illustration of Decomposition (in class).**

Definition 2.1 (Coefficient of Determination (R^2)).

$$\begin{aligned} R^2 &= \frac{\text{var. due to regression}}{\text{total variability}} \\ &= \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} \\ &= \frac{TSS - RSS}{TSS} \\ &= 1 - \frac{RSS}{TSS} \\ &= 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2} \end{aligned}$$

- Typically called “R-squared.”
- **Interpretation.** The proportion/fraction of the total variability (as measured by TSS) in the response accounted for by the **linear** association

of the inputs x_1, \dots, x_k with the output, y . (Note, at this point, we are often warned *not* to imply that R^2 is the proportion of variability in the outputs *caused* by or, less strongly, *explained* by, the inputs, unless data are collected from a randomized experiment, hence the careful use of *accounted* and *association*.)

- **For SLR.** For simple linear regression, with one predictor, $R^2 = r^2$, where

$$r = \frac{\sum_i (y_i - \bar{y})(x_i - \bar{x})}{(n - 1)s_y s_x}$$

is the (empirical) correlation between observed output y and input vector $\mathbf{x} = (x_1, \dots, x_n)^t$, with s_y and s_x being empirical standard deviations of the outputs and inputs, respectively.

- **Somewhat Analogously for MLR.**

$$R^2 = \left(\frac{\sum_i (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{(n - 1)s_y s_{\hat{y}}} \right)^2 = r_{y\hat{y}}^2 \quad \text{I like this one,}$$

the squared empirical correlation between the observed values and the fitted values (based on inputs; some detail omitted). This is suggestive of popular diagnostic plots of **observed values**, y_i , **verses predicted values**, \hat{y}_i , which we will see, later. (Note, $\bar{\hat{y}} = \bar{y}$ if you have an intercept in your model.)

- **WARNING.** R somehow anticipates what you intend for a **null model**, with no predictors, in the following sense. If you include an intercept, β_0 , in your model, then R considers the null model to consist only of β_0 , as we have discussed, and will estimate it as $\hat{\beta}_0 = \bar{y}$ and will compute TSS as in the definition of R^2 (“corrected for the mean”). However, if you omit β_0 (unusual, but not unheard of), then R “thinks” your null model is $\beta_0 = 0$ and uses this value instead of $\hat{\beta}_0 = \bar{y}$ to compute TSS (in this case, it’s “not corrected for the mean”). This will often give a misleadingly high R^2 value (unless \bar{y} happens to be close to 0). See `r.squared` in `help(summary.lm)`.

The version $R^2 = r_{yy}^2$ seems to me to implement what is intended for a measure of goodness of fit, for a linear model, intercept or not, or non-linear model, including random forests, neural nets, etc.—any procedure that gives fitted values. (As mentioned, associated plot of “observed vs. fitted” is often produced along with this R^2 .)

```
> ## With an intercept, beta0, results agree with our R^2 definition
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+             data=gala)
> lmodsum<- summary(lmod)
> lmodsum$r.squared
[1] 0.76585

> ## r^2_{y,yhat} always works
> cor(gala$Species, fitted(lmod))^2
[1] 0.76585

> mean(gala$Species); mean(fitted(lmod)) ## <-- same when beta0 in model
[1] 85.233
[1] 85.233
```

```
> ## Without intercept, R^2 reported by R may be misleading
> lmodnob0 <- update(lmod, . ~ . -1) ## TBD: CHANGE NAME e.g. b0eq0?
> lmodnob0sum<- summary(lmodnob0)
> lmodnob0sum$r.squared

[1] 0.85019

> ## r^2_{y,yhat} always works (only slightly lower than original
> ## model with intercept because estimated intercept is not inconsistent
> ## with zero)
> cor(gala$Species, fitted(lmodnob0))^2

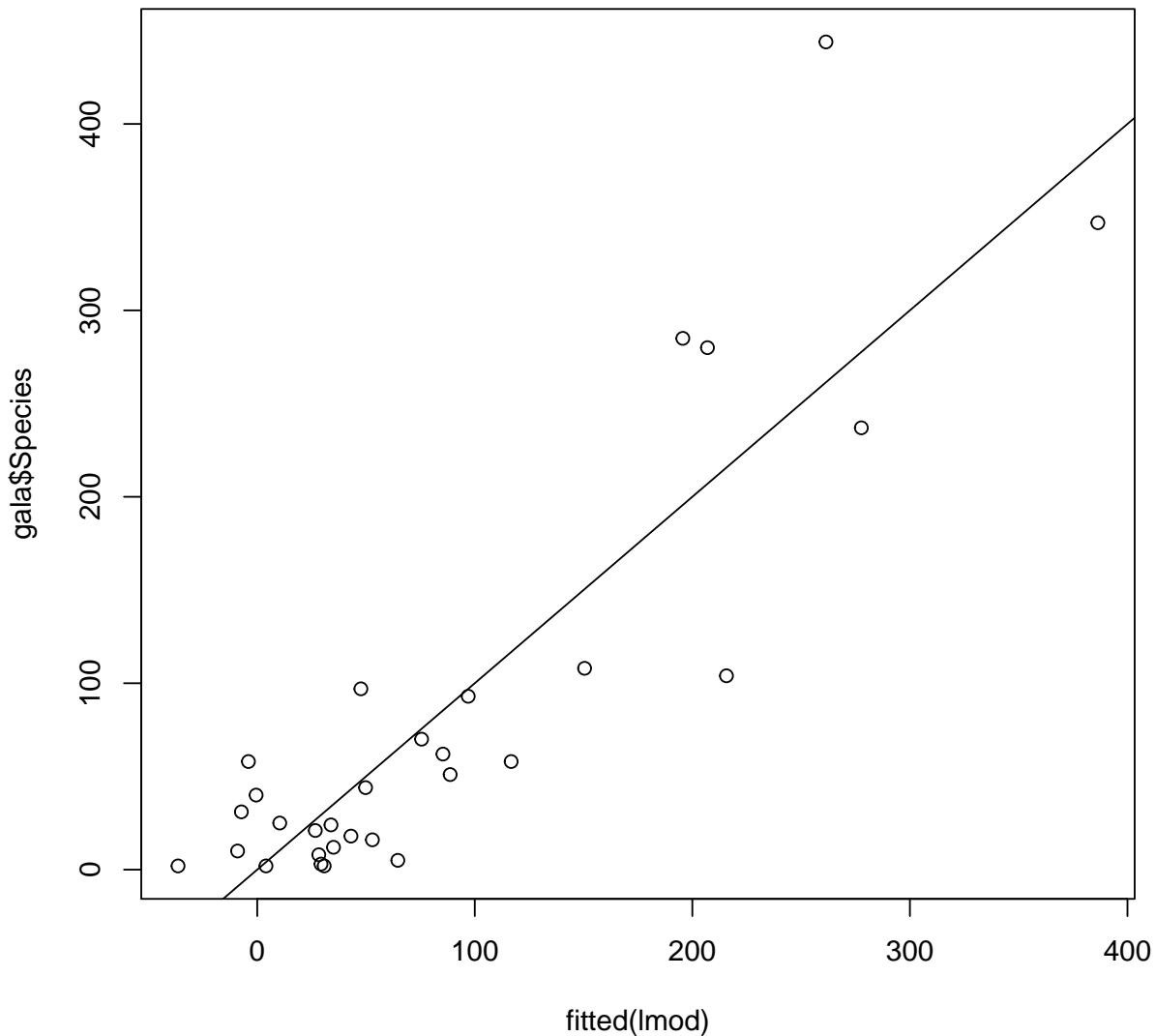
[1] 0.76527

> mean(gala$Species); mean(fitted(lmodnob0)) ## <-- not same means w/o beta0
[1] 85.233
[1] 82.846
```

```
> ## R^2 = r^2_{y,yhat} version always has nice graphical interpretation
> ## (using fit with b0 to illustrate):
>
> cor(gala$Species, fitted(lmod))^2

[1] 0.76585

> plot(gala$Species ~ fitted(lmod))
> abline(c(0,1))
```



- $0 \leq R^2 \leq 1$ (using Def. 2.1 for linear models with intercept or using $R^2 = r_{yy}^2$ as $-1 \leq r \leq 1$)
- What is a “good” R^2 ? It depends on the field of study and on your model.

- $R^2 = 1$ means fitted values equal observed values, a “perfect” fit, which is highly unlikely in practice; it’s more likely that you have a ridiculous model.
- $R^2 = 0$ is unlikely in practice, too, unless your model is the null model (again, somewhat ridiculous)!
- A high R^2 by itself does not by itself mean that your model is adequate. You should perform obligatory diagnostics, too.
- A low R^2 does not necessarily suggest there is no relationship between Y and \mathbf{x} ; there may be quite a distinct **non-linear** relationship between Y and \mathbf{x} .
- If you have a practical understanding of the error standard deviation, σ , then you might consider the **residual standard error** $\hat{\sigma}$ as an additional indication of the goodness of fit (along with diagnostics).

2.10 Identifiability

Sometimes, we cannot solve for unique values of all parameters in β because the \mathbf{X} matrix is not full rank, hence $(\mathbf{X}^t \mathbf{X})$ is not invertible ([Far14, §2.10], Appendix B).

1. Dumb situations: scaled (linear function of) predictor is included with the predictor itself. Non-linear transformations of predictors are okay.
2. Big p ($p > n$): throw out predictors or use different methods (e.g., forward step-wise predictor selection, shrinkage, principal components,)
3. Factor (categorical) predictors involve standard methods to deal with non-identifiability (non-estimable parameters) [Far14, Chap. 14-17]

2.11 Orthogonality

- Orthogonality is important to **parameter interpretation** [Far14, Chap. 5] and **stability of parameter estimates** [Far14, §7.3].
- For two vectors, orthogonality means they are geometrically **perpendicular**, i.e., their **dot product is zero** (aka, more generally, inner product or scalar product).
- That is, if $\mathbf{x}_j = (x_{1j}, \dots, x_{nj})^T$ and $\mathbf{x}_{j'} = (x_{1j'}, \dots, x_{nj'})^T$ are two n -vectors of observed predictor variables, $j \neq j'$, i.e., two columns of \mathbf{X} , then their dot product is $\mathbf{x}_j^T \mathbf{x}_{j'} = 0$ if they are orthogonal.
- We can generalize this to column partitions of the \mathbf{X} matrix.
- For example, suppose we have the partition $\mathbf{X} = [\mathbf{X}_1 | \mathbf{X}_2]$, with corresponding partitions of the parameter vector $\boldsymbol{\beta}^t = (\boldsymbol{\beta}_1^t, \boldsymbol{\beta}_2^t)$.
- Then can write our model as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} = \mathbf{X}_1\boldsymbol{\beta}_1 + \mathbf{X}_2\boldsymbol{\beta}_2 + \boldsymbol{\epsilon}.$$

- If each column of \mathbf{X}_1 is orthogonal to each column of \mathbf{X}_2 , then $\mathbf{X}_1^T \mathbf{X}_2 = \mathbf{0}$ (matrix), and

$$\mathbf{X}^T \mathbf{X} = \begin{pmatrix} \mathbf{X}_1^T \mathbf{X}_1 & \mathbf{0} \\ \mathbf{0}^T & \mathbf{X}_2^T \mathbf{X}_2 \end{pmatrix},$$

and

$$\begin{aligned} \hat{\boldsymbol{\beta}}_1 &= (\mathbf{X}_1^T \mathbf{X}_1)^{-1} \mathbf{X}_1 \mathbf{Y} \\ \hat{\boldsymbol{\beta}}_2 &= (\mathbf{X}_2^T \mathbf{X}_2)^{-1} \mathbf{X}_2 \mathbf{Y}, \end{aligned}$$

so that each sub-parameter LS estimate does not depend on the other.

- Again, orthogonal predictors will aid interpretation of the linear model [Far14, Chap. 5]; the value of and hence interpretation of $\hat{\boldsymbol{\beta}}_1$ does not depend on $\hat{\boldsymbol{\beta}}_2$ or \mathbf{X}_2 and vice-versa.

- Also, again, orthogonality subdues unstable (high variance) parameter estimates stemming from colinear predictors (nearly linearly dependent (§B.2.6) column vectors of \mathbf{X}) [Far14, §7.3].
- Orthogonality is often a part of designed experiments but almost never happens in observational studies (except between the 1 input column and the other input columns when centered so that $\hat{\beta}_0$ is uncorrelated with remaining regression model parameter estimators (next bullet), and β_0 is then interpretable as the response for the ‘typical’ case)).
- Orthogonal vectors have zero empirical covariance and correlation.
- Partially conversely, if centered by subtracting the average value, uncorrelated predictor column vectors are orthogonal.

Example: Chemical Odor

As mentioned, orthogonality is often part of an experimental design. Here, we look at data from an experiment to determine the effects of (factor predictors) column temperature, gas/liquid ratio and packing height in reducing unpleasant odor (response) of chemical product that was being sold for household use. We will use factor predictors, later. For now, notice how the vectors of numerically coded values of factor levels are orthogonal among the three factors.

```
> ## Designed experiment
> data(odor, package="faraway")
> odor

  odor temp  gas pack
1   66   -1   -1    0
2   39    1   -1    0
3   43   -1    1    0
4   49    1    1    0
5   58   -1    0   -1
6   17    1    0   -1
7   -5   -1    0    1
8  -40    1    0    1
9   65    0   -1   -1
```

10	7	0	1	-1
11	43	0	-1	1
12	-22	0	1	1
13	-31	0	0	0
14	-35	0	0	0
15	-26	0	0	0

```
> ## Zero predictor empirical covariance / correlation
> cov(odor[,-1])
```

```
          temp      gas      pack
temp  0.57143 0.00000 0.00000
gas   0.00000 0.57143 0.00000
pack  0.00000 0.00000 0.57143
```

```
> cor(odor[,-1])
```

```
      temp  gas  pack
temp    1   0   0
gas     0   1   0
pack    0   0   1
```

```
> ## Orthogonality translates to zero covariance/correlation of betahat
> ## (up to numerical precision)
> lmod <- lm(odor ~ temp + gas + pack, odor)
> summary(lmod, cor=T)
```

Call:

```
lm(formula = odor ~ temp + gas + pack, data = odor)
```

Residuals:

Min	1Q	Median	3Q	Max
-50.20	-17.14	1.18	20.30	62.93

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	15.2	9.3	1.63	0.13
temp	-12.1	12.7	-0.95	0.36
gas	-17.0	12.7	-1.34	0.21
pack	-21.4	12.7	-1.68	0.12

```
Residual standard error: 36 on 11 degrees of freedom
Multiple R-squared:  0.334, Adjusted R-squared:  0.152
F-statistic: 1.84 on 3 and 11 DF,  p-value: 0.199
```

Correlation of Coefficients:

	(Intercept)	temp	gas
temp	0.00		
gas	0.00	0.00	
pack	0.00	0.00	0.00

```
> vcov(lmod)
```

	(Intercept)	temp	gas	pack
(Intercept)	86.455	0.0000e+00	0.0000e+00	0.0
temp	0.000	1.6210e+02	1.9089e-14	0.0
gas	0.000	1.9089e-14	1.6210e+02	0.0
pack	0.000	0.0000e+00	0.0000e+00	162.1

```
> ## Fit of one beta and the zero cov/cor not affected by other beta.
> ## Estimated standard errors change, of course (but not wildly as may
> ## happen with collinearity).
> lmod <- update(lmod,. ~ . - temp) ## e.g.
> summary(lmod)
```

Call:

```
lm(formula = odor ~ gas + pack, data = odor)
```

Residuals:

Min	1Q	Median	3Q	Max
-50.20	-26.70	1.17	26.80	50.80

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	15.20	9.26	1.64	0.13
gas	-17.00	12.68	-1.34	0.20
pack	-21.37	12.68	-1.69	0.12

```
Residual standard error: 35.9 on 12 degrees of freedom
Multiple R-squared:  0.279, Adjusted R-squared:  0.159
F-statistic: 2.32 on 2 and 12 DF,  p-value: 0.141
```


Lecture 3

Inference

Contents

Introduction	63
3.1 Hypothesis Tests to Compare Models	64
Likelihood Ratio Test	67
General Linear Hypothesis (GLH)	68
Reality, Decisions, Errors & Error Probabilities	72
Careful Wording	74
p-value	76
Reporting Test Results	77
Hypothetical Replications	79
3.2 Testing Examples	81
3.2.1 Test of all the predictors (overall F-test)	81
3.2.2 Testing one predictor	87
3.2.3 Testing a pair of predictors	90
3.2.4 Testing a subspace	91
3.2.5 Non-Zero Hypothesized Value $H_0: \beta_{Elevation} = 0.5$	93
3.2.6 Tests we cannot do (in INF 511)	95
Power	95
Non-Central F Distribution	96
Example: Power for Effect of One Predictor	97
pwr Package	99
Example: A Priori Power Analysis	102
Example: Post Hoc Power Analysis	107
Test Sidedness, Distribution Tails & Power	111

3.3	Permutation Tests	120
3.3.1	Example: Permutation Test of Overall Association	122
3.3.2	Example: Permutation Test of Single Predictor	124
3.3.3	Take-Home Remarks	127
3.4	Sampling	128
3.5	Confidence Intervals for β	128
	Correspondence Between Tests & Intervals	133
3.6	Bootstrap Confidence Intervals	135

Introduction

- **LS Estimator, Properties, Etc.** The LS estimation method of the previous chapter produced estimators ($\hat{\beta}$ and $\hat{\sigma}^2$ (MSE)). An assumption of random errors (with zero mean) gave expected values,

$$\begin{aligned} E(\hat{\beta}) &= \beta \quad \text{LS estimator is unbiased} \\ E(\hat{\sigma}^2) &= \sigma^2, \end{aligned}$$

and variance,

$$\text{Var}(\hat{\beta}) = \sigma^2(\mathbf{X}^t \mathbf{X})^{-1},$$

not to mention

- **fitted values**,
- **residuals**,

etc. Though we used the notion of random variables and expectation, and despite examples in Appendix B often indicating normality, **none of these properties require normality**, including the goodness of the Gauss-Markov theorem.

- **Inference.** However, to obtain **probabilistic** statements for further inference, we will use a full probability distribution, going beyond just (point estimates of) expectations and variances, i.e., going beyond just the point estimates $\hat{\beta}$, $\hat{\sigma}^2$ and $\hat{\sigma}^2(\mathbf{X}^t \mathbf{X})^{-1}$.
- **INF 511 Assumes Normal Errors.** For further inference in INF 511, we assume normality of errors, which leads to

$$\hat{\beta} \sim N(\beta, \sigma^2(\mathbf{X}^t \mathbf{X})^{-1}),$$

which, in turn, leads to typical t and F ‘sampling distribution’ procedures—likely the focus of your previous, introductory statistics courses—including tests at pre-specified Type I / Type II error rates (e.g., Type I error probability $\alpha = 0.05$), p-values of test statistics ([Far14, §3.1, 3.2]), and confidence interval estimates of parameters with nominal confidence levels (e.g., $(1 - \alpha) = 0.95$) ([Far14, §3.5]).

- **Check Assumptions!** We will check our regression model assumptions when discussing model diagnostics ([Far14, Chap. 6]) to help justify the inference procedures that follow from model assumptions or to motivate remodeling (not much of the latter in INF 511).
- **Abnormal?** When normality is not a reasonable assumption, we (INF 511) will consider permutation/randomization tests ([Far14, §3.3, 5.3]) and bootstrap confidence intervals ([Far14, §3.6]). INF 512 focuses on other methods for non-normal regression models.
- **Large n .** When n is “large,” however, we may still get valid, approximate inference for functions of the regression model parameters, β , using z , χ^2 , t or F distributions, despite errors (and responses) not being normal, by appealing to some version of the CLT (§A.11). In short, if our regression (mean) model is correct, and our variance model (uncorrelated, constant variance) is correct, then we get asymptotically correct inference from our procedures, meaning correct in the limit as $n \rightarrow \infty$, with the approximation getting better with larger n . More in INF 512.
- **Appendix B.** Our methods here rest mainly on details given in B.5, to which we may visit directly more or less depending on time.

3.1 Hypothesis Tests to Compare Models

NOTE, your author now seems to switch to p to denote the number of parameters in the mean (regression) model. This is now consistent with our notes. He also uses q to denote a number of parameters; but we use a subscript on p , instead, as we will see, and we briefly use q to denote number of predictors (perhaps we should have used k_ω as you will see). There should be no confusion.

- **Full (Ω or F) vs. Reduced (ω or R) Regression Models.** Assume that we have a current model under consideration. For reasons to become clear momentarily, we now call this model our full model (denoted Ω (book notation) or F). Consider a second model, which is a simplification of the full model in the sense that this reduced model is obtained by (linearly) restricting the regression model parameters of the full model. (More on such linear restrictions, shortly.) We call this simplified model a reduced model (denoted ω (book notation) or R), and we say that the reduced model is **nested** in the full model because the restriction makes the set of possible parameter values of the reduced model a subset of the set of possible parameter values of the full model.
- **E.g., Typical Full and Reduced Models.** For example, we often test a (full) model to omit predictors to obtain a more **parsimonious** (reduced) model. In this case, this is equivalent to restricting to zero the parameters of the full model corresponding to the omitted predictors.
- **Full and Reduced Model Residual Sums of Squares (RSS).** Consider the residual sums of squares (RSS) for the full (Ω) and the reduced (ω) models,

$$\begin{aligned} RSS_{\Omega} &= (\mathbf{Y} - \mathbf{x}_{\Omega}^T \hat{\boldsymbol{\beta}}_{\Omega})^T (\mathbf{Y} - \mathbf{x}_{\Omega}^T \hat{\boldsymbol{\beta}}_{\Omega}) \\ &= \hat{\boldsymbol{\epsilon}}_{\Omega}^T \hat{\boldsymbol{\epsilon}}_{\Omega} \\ &= (n - p_{\Omega}) \hat{\sigma}_{\Omega}^2 \end{aligned}$$

and

$$\begin{aligned} RSS_{\omega} &= (\mathbf{Y} - \mathbf{x}_{\omega}^T \hat{\boldsymbol{\beta}}_{\omega})^T (\mathbf{Y} - \mathbf{x}_{\omega}^T \hat{\boldsymbol{\beta}}_{\omega}) \\ &= \hat{\boldsymbol{\epsilon}}_{\omega}^T \hat{\boldsymbol{\epsilon}}_{\omega} \\ &= (n - p_{\omega}) \hat{\sigma}_{\omega}^2. \end{aligned}$$

- **Compare RSS.** If the full model **residual sum-of-squares**, RSS_{Ω} , is much less than reduced model's, RSS_{ω} , i.e., if the difference is sums of squares,

$$RSS_{\omega} - RSS_{\Omega},$$

is somehow “large,” then we may tend to think that the reduced model does not “explain” (beware cause-effect interpretation) as much of the residual variability as does the full model, and we may tend to reject the reduced model in favor of the full model. Otherwise, if the sums of squares are somehow not different, then we may not reject the reduced model in favor of parsimony.

- [Far14, Fig. 3.1, p. 34] gives a geometric interpretation of comparing full (big) and reduced (small) models. The squared length of the figure’s model residual vectors are the models’ residual sums of squares. The squared length of the difference of residual vectors between the models is the difference of the residual sums of squares (squared lengths). If the difference is somehow large compared to the squared length of the full model’s residual vector, then we tend to reject the reduced model in favor of the full model.
- **F Statistic.** This intuition is good, but we standardize (or denominate) the difference of RSS values by the difference in the number of parameters, $r = p_\Omega - p_\omega$, and compare the difference relative to that of the full model. (Note that the difference in RSS values is “reduced minus full,” but the difference in model sizes (number of parameters) is “full minus reduced,” so that we always have a non-negative result.) That is, we will consider, instead, the “largeness” of

$$F = \frac{(RSS_\omega - RSS_\Omega)/(p_\Omega - p_\omega)}{RSS_\Omega/(n - p_\Omega)}, \quad (3.1)$$

- **F Distribution Assuming Reduced Model True.** As the notation suggests, the above F statistic follows an F distribution, under the **(null) hypothesis**, H_0 : reduced model is true. That is,

$$F \sim F(df_1 = p_\Omega - p_\omega, df_2 = n - p_\Omega),$$

where, often, df_1 is called the **numerator degrees of freedom** and df_2 is called the **denominator degrees of freedom**. (Google “F distribution.”)

- **How Large Must F Be Before We Reject the Reduced Model?**

Again, if the F statistic is large, this suggest that the reduced model is not true and that the richness of the larger model is required to be more consistent with the data. We typically determine the **cut-off** F value, or **critical value**, at which non-rejection/rejection hinges, by somewhat large quantiles of the F distribution (assuming the reduced model is true), i.e., reject when

$$F \geq F_{crit} = F(1 - \alpha, p_\Omega - p_\omega, n - p_\Omega),$$

where, α typically takes on values 0.01, 0.05, or 0.10 (similar to rejecting for large t values where the critical value is determined by quantiles of the t , as you likely did in previous courses). More on α , critical values and **rejection regions**, shortly.

Likelihood Ratio Test

- **LRT .** Another criterion for comparing two models is to compare their (here, normal) likelihoods; see the unnumbered section beginning on page 38. In particular, we can compute the **likelihood ratio test (LRT) statistic**

$$LRT = \frac{\max_{(\beta, \sigma^2) \in \Omega} L(\beta, \sigma^2)}{\max_{(\beta, \sigma^2) \in \omega} L(\beta, \sigma^2)}.$$

- **Reject Reduced for Large LRT .** Intuitively, when the reduced model (ω) is simple compared to the full model (Ω), it will have a smaller likelihood when evaluated at its MLE; again, see the unnumbered section beginning on page 38. As with the F statistic, above, if this ratio is “large,” then we may reject the reduced model in favor of the full model, which makes the data more likely in this case, otherwise, we may not reject the reduced model in favor of parsimony.

- **A Figure To Be Drawn in Class.** A figure akin to [Wak13, Figure 2.4] (a text used in INF 512).
- **F is Proportional to LRT .** We will discuss the LRT statistic more in INF 512. Here, we merely mention that our F statistic is proportional to the LRT,

$$\begin{aligned} F &= \frac{(RSS_{R\omega} - RSS_{\Omega})/(p_{\Omega} - p_{\omega})}{RSS_{\Omega}/(n - p_{\Omega})} \\ &= \frac{n - p_{\Omega}}{p_{\Omega} - p_{\omega}} \frac{(RSS_{R\omega} - RSS_{\Omega})}{RSS_{\Omega}} \\ &= \frac{n - p_{\Omega}}{p_{\Omega} - p_{\omega}} LRT. \end{aligned}$$

- **F and LRT Result in Same Test (INF 511).** Thus, rejecting the null hypothesis for a large LRT statistic corresponds to rejecting for large a F statistic. (Typically, when performing an LRT, the reference distribution is a χ^2 , which an appropriately rescaled F approaches with large sample size, n . More in INF 512.)

General Linear Hypothesis (GLH)

Before moving on to testing full and reduced models, we present testing a bit more formally and introduce a different, but equivalent, approach to testing, which directly relates to the F quadratic form in Result B.9, in Appendix B, which you should read.

- Consider the **General Linear Hypothesis (GLH)**

$$H_0 : C\beta = d \quad \text{null hypothesis}$$

$$H_1 : C\beta \neq d \quad \text{alternative hypothesis}$$

for some $r \times p$ matrix of linear combination coefficients \mathbf{C} . We will not consider hypotheses/estimation involving non-linear functions of β [Far14, top p. 40]. Each row of the matrix, \mathbf{C} , when multiplied by β , gives a linear combination of the parameters in β , e.g., from the first row of \mathbf{C} , $\sum_{j=1,\dots,p} c_{1j}\beta_{j-1}$. (Having read Appendix B will shed light on the matrix multiplication going on here.)

- **Current Model: Full Model (F or Ω)**. The idea behind the GLH is that we have a current model under consideration, which, for reasons that will become clear momentarily, we call the **full model**, denoted by Ω (or F). Of course, this will correspond to the full model discussed previously.
- **Full Model Number of Parameters p_Ω (or p_F)**. The full model has p_Ω unrestricted (free) **parameters** in β . (or p_F)
- **Null, Restrictions & Reduce Model (R or ω)**. The specified coefficients in the $r > 0$ (linearly independent) rows of \mathbf{C} in the **null hypothesis** act to place r (linear) restrictions/constraints on the parameters of the model under consideration to obtain a **reduced or nested model**, denoted ω (or R), with $p_\omega = p_\Omega - r$ free **parameters**. (or $p_R = p_F - r$)
- **Different Complexity**. That is, $r = p_\Omega - p_\omega$ is the **difference in the number of (free) parameters in the full and reduced models**, a difference in model sizes, a difference in model complexity.
- **Null/Reduced Nested in Alternative/Full**. Thus, as discussed in the *RSS* approach to comparing full and reduced models, previously, the null hypothesis/reduced model correspond to a special case of the alternative hypothesis/full model, and we say that the null hypothesis/reduced model is **nested** in the alternative hypothesis/full model. Hence the terminology “full” and “reduced” (again).
- **F Statistic**. Again, as above, we may test the hypothesis via an F

statistic (Result B.9), assuming, further, the null hypothesis to be true,

$$F = (\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{d})^T(\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1}(\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{d}) / \text{rank}(\mathbf{C}) \quad (3.2)$$

$$\sim F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p_\Omega). \quad (3.3)$$

- (Thus, we assume a value of $\mathbf{C}\boldsymbol{\beta} = \mathbf{d}$ to get around not knowing it in the testing situation.)
- Without showing details, it turns out that

$$\begin{aligned} F &= (\mathbf{C}\hat{\boldsymbol{\beta}}_\Omega - \mathbf{d})^T(\hat{\sigma}_\Omega^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1}(\mathbf{C}\hat{\boldsymbol{\beta}}_\Omega - \mathbf{d}) / r \\ &= \frac{(RSS_\omega - RSS_\Omega)/(p_\Omega - p_\omega)}{RSS_\Omega/(n - p_\Omega)}, \\ &= \frac{(RSS_\omega - RSS_\Omega)/(\text{rank}(\mathbf{C}))}{RSS_\Omega/(n - p_\Omega)} \\ &\sim F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p_\Omega), \end{aligned}$$

where $\text{rank}(\mathbf{C}) = r = p_\Omega - p_\omega$ for some reduced model corresponding to \mathbf{C} .

Common Hypothesis to Omit Predictors

- To help clarify the above general linear hypothesis, consider the particular, very common case where we have k predictors in a **full model**, and we consider omitting $0 < r \leq k$ of them to arrive at a **reduced model** with $0 \leq q = k - r < k$ predictors.
- Thus, considering an intercept in each model, we have a $p_\Omega = k + 1$ dimensional parameter, $\boldsymbol{\beta}$, in the full model, which we can partition into two sub-vectors

$$\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2],$$

where $\boldsymbol{\beta}_1 = (\beta_0, \dots, \beta_q)^t$ and $\boldsymbol{\beta}_2 = (\beta_{q+1}, \dots, \beta_k)^t$, of size $p_\omega = q + 1$ and $r = k - q$, respectively.

- Thus, particular case of omitting $r = k - q$ covariates corresponds to restricting $r = p_\Omega - p_\omega$ parameters in the full model to equal zero, i.e.,
- In this case, we may write

$$\begin{aligned} H_0 &: \beta_1 \text{ unrestricted}, \beta_2 = \beta_{20} \\ H_1 &: \beta = [\beta_1, \beta_2] \neq [\beta_1, \beta_{20}], \end{aligned}$$

where, β_{20} the r -vector of specified “null” values, all zeros in this particular case. Read subscript: ‘two naught’ not ‘twenty.’

- Clearly, the reduced model/null hypothesis is nested in the full model/alternative hypothesis.
- **In the GLH form, what would C look like in this special case of omitting r covariates? What about d ?** We'll give examples, shortly.

Other Remarks

- **Zero Isn't Everything.** Of course, we are not bound to setting parameters to zero, necessarily. We will illustrate.
- **Beware of Inferring the Intercept.** Note that we typically want to give special consideration before setting the intercept to zero (which the setup of the above particular case seems to discourage).
- **Whichever is Easier.** Unlike the common case of omitting predictors, sometimes, it may not be easy to see the correspondence between C and a reduced model. That is, starting with an interesting linear combination, $C\beta$, we may not easily see a reduced model, and, conversely, with a reduced model, it may be relatively difficult to find C . Luckily, we may use the F v R (Ω v ω) RSS approach discussed initially, or we may use the $C\beta$ /GLH approach. We demonstrate both, shortly.

Reality, Decisions, Errors & Error Probabilities

- **Error Matrix.** Table 3.1 provides more insight into our conventional hypothesis testing framework. To be discussed in class.

Error Matrix		Decision	
		Do not reject null	Reject null
Reality	Null is true	No error $\Pr(\text{no reject} \mid \text{null true}) = 1 - \alpha$	Type I error $\Pr(\text{reject} \mid \text{null true}) = \alpha$
	Null is false	Type II error $\Pr(\text{no reject} \mid \text{null false}) = \beta$	No error $\Pr(\text{reject} \mid \text{null false}) = 1 - \beta$ (power)

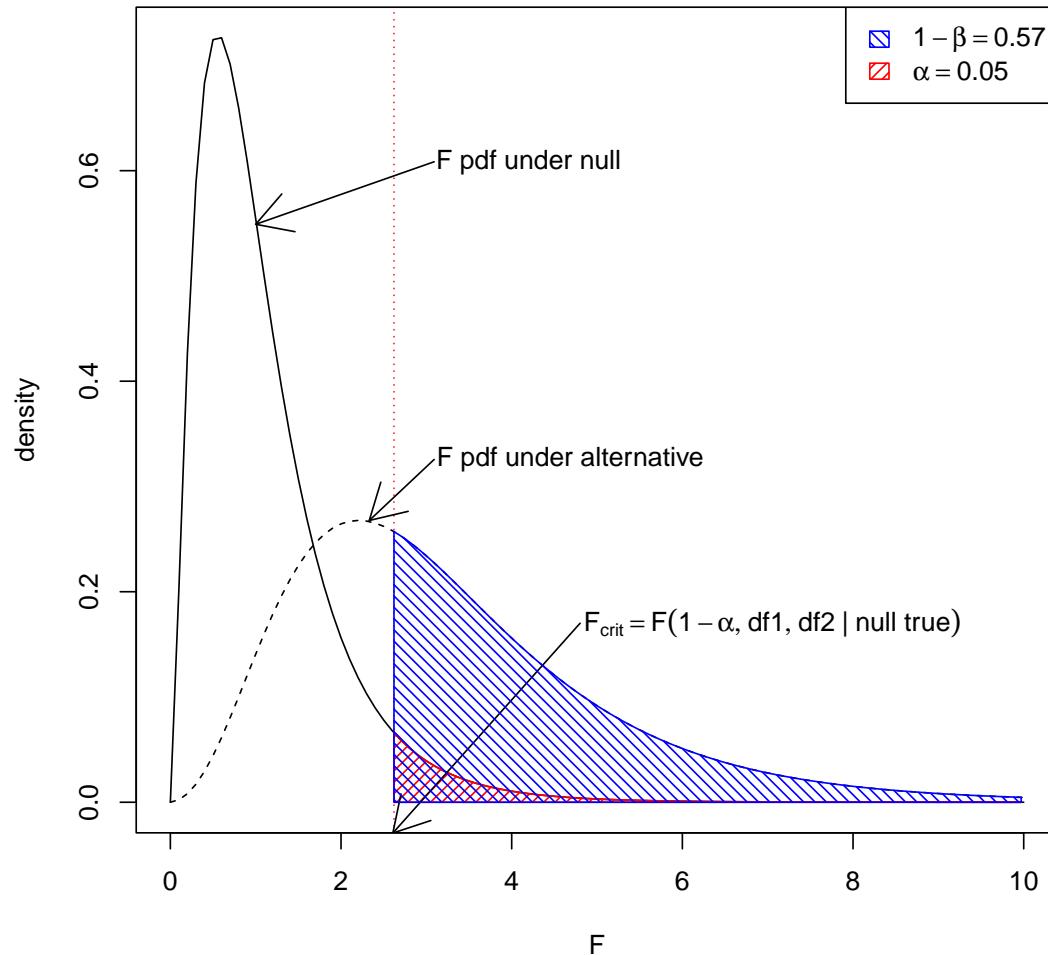
Table 3.1: Reality, decisions, errors and error probabilities.

- **Make Error Probabilities Small.** Ideally and intuitively, we would like to make each of the probabilities of a **type I error** and **type II error** small. (Right?!) But, to make both small typically requires relatively large sample sizes or requires “effects” to be too small to be of practical importance. We’ll discuss effects more as the course progresses. For now, think of effects as being differences between mean/regression models. Intuitively, we need relatively few data to detect large effects, but we need more data to detect small effects (And, we don’t want to waste time, effort, resources to collect data to detect an effect that’s considered to be too small in a practical sense, e.g., what would it mean, practically speaking, to detect a difference in blood pressure of 0.1 mm Hg when blood pressure magnitudes are on the order of 100 and can range over 10’s of mm Hg?)
- **Set Type I Error Probability α .** In practice, we make $\alpha = \Pr(\text{reject } H_0 \mid H_0 \text{ true})$ small, **A PRIORI, BEFORE LOOKING AT THE DATA** (unless we are using post-hoc methods that accommodate

data snooping). Conventionally, as mentioned above, $\alpha = 0.05$, more or less (usually 0.10 or 0.01), depending on the field of practice. α is also called the **(statistical) significance (level)** (of the test; or just **level** (of the test)) or the **error rate** (of the test). Of course, many of our methods will use F distributions to compute such probabilities. In this case, as already mentioned, rejecting the null/reduced model for

$$F \geq F_{crit} = F(1 - \alpha, p_\Omega - p_\omega, n - p_\Omega),$$

gives a level α test of the null hypothesis/reduced model, and we see that small significance levels correspond to a relatively large F values (**assuming the null hypothesis is true**), which we called the **critical value**, denoted F_{crit} . See nearby figure, below.



Careful Wording

- We often hear careful wording in describing the outcome of a test.

- If the null hypothesis is rejected, we say
“we **reject** the null hypothesis (at level α)”.

(More informally, but deviating from our point here, we may say that the data are not consistent with the null hypothesis (at level α)).

- If the null is not rejected, we say
“we **do not reject** the null hypothesis (at level α)”;

We should be careful **not** to say “we **accept** the null hypothesis.”
The nearby plot and the discussion below helps to explain why.

- **Null F Distribution.** When the null hypothesis, $H_0: C\beta = d$, is true, our F statistic follows the aforementioned F distribution, which we now distinguish as a **central** F distribution for reasons we will not discuss thoroughly.
- **Alternative F Distributions.** When the alternative hypothesis, $H_1: C\beta \neq d$, is true, then $C\beta = d_1 \neq d$, and our F statistic follows a **non-central** F distribution with an extra, non-centrality parameter depending on the “effect size,” i.e., the size of $d_1 - d$; different non-central F distributions result for different effect sizes. (The degrees of freedom are the same as the central F .)
- **Plot of $1 - \beta$ and α .** The plot shows a **central** F and a **non-central** F along with probabilities of rejection, α (red) and power (blue), respectively; these correspond to a particular null and a particular alternative for some testing setup that we will discuss in more detail, later. For now, we ask, **how does the figure suggest care with words describing test results?** In short, $\beta = \Pr(\text{not reject} \mid \text{null false})$ may be relatively high; the figure suggests that not rejecting the null is a very probable event ($\beta = 0.43$) when the null is false, i.e., when the observed F came from an alternative model, different from the null. Saying “**not reject**” is a way of acknowledging the possibility of low power (high β)—that the null may still not be true—whereas “**accept**” does not. We discuss power a bit more in a subsequent section.

p-value

Definition 3.1 (p-value). *The probability of a test statistic value from the null distribution being as extreme as or more extreme than the actual value observed.*

In INF 511, our model assumptions (to be checked!) lead to F or t sampling distributions used to compute p-values.

$$\text{p-value} = \Pr(F \geq F_{stat} \mid \text{null true})$$

or, in terms of one-sided t tests,

$$\text{p-value} = \Pr(t \geq t_{stat} \mid \text{null true})$$

or

$$\text{p-value} = \Pr(t \leq t_{stat} \mid \text{null true}),$$

as the case may be.

Remark 3.1 (Interpreting the p-value).

- *Probability of observing data (or a statistic) at least as extreme as those (the one) actually observed, given that the null hypothesis is true. (definition)*
- *A small p-value suggests either that we have observed a rare (but possible) result under the null, or that the null is not true.*
- *A measure of how plausible our data (or statistic) is to have arisen from a particular distribution (the distribution under the null hypothesis). That is, a measure of how (in)consistent our data/statistic is with the null hypothesis/model.*

- *The proportion of times (relative frequency or rate) that we would observe data (or a statistic) at least as extreme as those (the one) we actually did observe, in a large number of **hypothetical replications** of our study and test procedure. See Hypothetical Replications section, below.*
- A small (large) *p*-value does not prove the alternative (null).
- Sometimes referred to as the **observed significance level**, but use of this term may be discouraged as it may lead to confusion with the **significance level**, α , which it is not.
- Generally NOT the probability that the null hypothesis is true. There exist particular instances of a particular statistical framework (Bayesian statistics) wherein *p*-values do coincide numerically exactly with probabilities that (particular) null hypotheses are true, but, despite numerical equivalence with probability, interpretation is much different depending on the perspective. 1-(*p*-value) is not the probability that the alternative is true.

Reporting Test Results

- **Test Reporting.** In a classic hypothesis testing framework, using a previous example to illustrate, we might report our results something like,

“the data are consistent with the null hypothesis, we do not reject the null hypothesis at level $\alpha = 0.05$, and we conclude no linear association of species counts with island elevation in the presence of remaining predictors”

or

“the data are not consistent with the null hypothesis, we reject the null hypothesis at level $\alpha = 0.05$, and we conclude a linear association of

species counts with island elevation in the presence of remaining predictors,"

for non-rejection and rejection, respectively.

- **Or, Report p-value.** However, when we do not have to make a decision regarding a null hypothesis, we would typically report the p-value associated with our test and let the reader decide the statistical significance, rather than simply reporting "reject" or "not reject" (at level α). We might say something like,

"the data indicate little support for a linear association of species counts with with island elevation in the presence of remaining predictors (p-value = 0.27),"

or

"the data support a linear association of species counts with with island elevation in the presence of remaining predictors (p-value = 0.001),"

- **p-value Reporting More Common.** The reporting of the p-value has been much more common than the more classic reject/non-reject approach in recent decades. The p-value allows readers to decide significance for themselves and indicates a degree of support for the data under the null hypothesis that a simple binary decision does not convey. (Perhaps we should say that a small p-value shows non-support for the data under the null (in support of some alternative), but we should be careful when discussing a large p-values' support for data under nulls unless we can ensure high power, which then suggests data are unlikely to have come from an alternative that is importantly different from the null, hence making us more comfortable letting large p-values suggest support for the data under the null.)
- **Statistically Significant vs Practically Significant.** (Or, statistical significance vs practical importance.)

- Who cares about a highly statistically significant result (small p-value) if the estimated effect size is of no practical importance? Perhaps we wasted resources to obtain a large sample for little practical gain.
 - Or, perhaps an estimate appears large and of practical importance, but is not statistically significant. In this case, perhaps we may be motivated to collect more data.
- **Confidence Interval, Too.** These notions of practical vs statistical significance has led to the reporting of interval estimates, too, in recent years, i.e., confidence intervals, along with testing results, to give readers and idea of the estimated **magnitude** of a parameter (or general estimand or “effect,” $C\beta$), which may help gauge the practical importance of a result. Again, if ‘most’ of an interval contains values of practical significance, we may be motivated to collect more data, as we have said.
 - **Power & Sample Size.** Relatedly, we may want to do a(n) (ideally, a priori,) power analysis to determine a sample size to detect a practically significant effect with sufficiently high power towards ensuring against either of the above cases.

Hypothetical Replications

Now is a fair time to discuss the **frequentist** statistician’s interpretation of probability. We use type I error probability to illustrate.

Definition 3.2 (Error Rate). *The error rate is the rate at which a testing procedure for performing a single test would falsely reject the null hypothesis of the test in a large number of hypothetical replications.*

$$\alpha = \lim_{\# \rightarrow \infty} \frac{\text{number of tests falsely rejecting the null}}{\# \text{ of tests}}$$

- **Hypothetical Replications.** At the heart of the frequentist statistician's interpretation of probability is the notion of hypothetical repeated experiments or repeated sampling studies; for each hypothetical and identical study—identical except for a different, randomly selected data set of the same size—we perform a test (or construct a confidence interval (later) or, more generally, perform some inference procedure).
- **Long-Run Rates or Proportions.** We perform the test (procedure) repeatedly (hypothetically)—many, many times—and, in the long run, the proportion of falsely rejected null hypotheses approaches α (or the coverage rate or the proportion of confidence intervals that include the true $C\beta$ approaches $1 - \alpha$; again, intervals later.)
- **Long-Run Procedure Performance Measure.** Note that we may view the type I error rate to indicate the performance of a testing procedure, i.e., how well a procedure works (in terms of a type I error) if we use it (properly!) many times.
- **A Priori!** Error rates (and confidence levels) are only valid when specified **before looking at the data**. In short, if you let your data suggest to you your inference, then probabilities computed under the assumption of the null distribution are often no longer valid. (Some procedures allow us to look at the data beforehand.) More later. (That is, be weary of data snooping, data dredging, data torture, p-hacking, trying and trying again, ...)
- **Often Hidden.** When using a mathematical model (e.g., normal or t or F), we often forget this fundamental frequentist notion, but will see it directly when discussing **randomization distributions** and **sampling distributions**, later.
- **Sometimes Not Applicable.** E.g., global average temperature tomorrow. Uncertain? Yes. Can we reasonably characterize our uncertainty

with a long-run relative frequency, i.e., frequentist, interpretation of probability? In the beginning... (Still, we may view the error rate as an assessment of the testing procedure, but we may be getting too philosophical...)

3.2 Testing Examples

3.2.1 Test of all the predictors (overall F-test)

Overall F-test Hypotheses

$$\begin{aligned} H_0 : \quad & \beta_1 = \beta_2 = \cdots = \beta_{k=p-1} = 0 \\ H_a : \quad & \text{not all } \beta_j, j = 1, \dots, k \text{ are zero} \end{aligned}$$

Notice β_0 is not part of this test.

We'll compute the test using both approaches discussed above.

- **$C\beta$ Approach.**

$$F = (\mathbf{C}\hat{\boldsymbol{\beta}}_\Omega - \mathbf{d})^T (\hat{\sigma}_\Omega^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1} (\mathbf{C}\hat{\boldsymbol{\beta}}_\Omega - \mathbf{d}) / r,$$

where $r = k$ in this case.

- **F v R Approach.**

$$F = \frac{(RSS_\omega - RSS_\Omega)/(p_\Omega - p_\omega)}{RSS_\Omega/(n - p_\Omega)},$$

where, in this case, $p_\Omega = k + 1$ and $p_\omega = 1$.

ANOVA. The second, RSS or F v R computations have traditionally been presented in an “analysis of variance (ANOVA)” table (granted, a very simple one for this overall F test).

- When discussing R^2 in §2.9, we discussed a **null model**, with just a constant, β_0 —no predictors—which is associated with a total sum of squares, $TSS = \sum_i (y_i - \bar{y})^2$. This null model was presented in the context of a current model of interest with predictors x_1, \dots, x_k and associated parameters β_1, \dots, β_k (and β_0); we denoted this model’s residual sum of squares as $RSS = \sum_i (\hat{y}_i - \bar{y})^2$. Obviously, these are **reduced and full models**, respectively.
- In the current context, $RSS_\omega = TSS$, and $RSS_\Omega = RSS$, and, now, we see

$$\begin{aligned} R^2 &= \frac{\text{var. due to regression}}{\text{total variability}} \\ &= \frac{\sum_i (\hat{y}_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} \\ &= \frac{TSS - RSS}{TSS} \\ &= \frac{(RSS_\omega - RSS_\Omega)}{RSS_\omega} \end{aligned}$$

as the proportion/fraction of this particular reduced (null) model’s variability that is (linearly) associated with this particular full model, as before, now using our more general F v R notation.

- Also, when discussing R^2 (§2.9), we wrote down a decomposition of sums-of-squares, which we re-write using the notation of our current context of the overall F -test.

$$\begin{aligned} \sum_i (y_i - \bar{y})^2 &= \sum_i (\hat{y}_i - \bar{y})^2 + \sum_i (y_i - \hat{y}_i)^2 \\ TSS &= (TSS - RSS) + RSS \\ RSS_\omega &= (RSS_\omega - RSS_\Omega) + RSS_\Omega \\ RSS_\omega &= SS_{reg} + RSS_\Omega \end{aligned}$$

- Obviously, we have a corresponding decomposition of the total degrees of freedom, again using notation for our current context.

$$\begin{aligned}(n - 1) &= (p - 1) + (n - p) \\ (n - p_\omega) &= (p_\Omega - p_\omega) + (n - p_\Omega)\end{aligned}$$

- **F v R Approach.** Now, for our overall F test, we have

$$\begin{aligned}F &= \frac{(RSS_\omega - RSS_\Omega)/(p_\Omega - p_\omega)}{RSS_\Omega/(n - p_\Omega)} \\ &= \frac{(TSS - RSS)/(p - 1)}{RSS/(n - p)} \\ &= \frac{SS_{reg}/(p - 1)}{RSS/(n - p)} \\ &= \frac{MS_{reg}}{MSE}\end{aligned}$$

- **SS Regression.** $SS_{reg} = TSS - RSS$, the sum of squares associated with (consumed by or “explained” by) the full (regression) model over and above the null model.
- **MS Regression.** And, the F numerator may be thought of as a mean square due to regression.

$$MS_{reg} = \frac{SS_{reg}}{p - 1}.$$

- **MS Error (MSE).** Notice the denominator is the mean squared error ($\hat{\sigma}^2$, p. 37) of the full model,

$$MSE = \frac{RSS}{n - p}.$$

- **ANOVA Table.** The decomposition of sums of squares and of degrees of freedom, and the mean squares and F , traditionally have been presented in an ANOVA table, e.g., [Far14, Tab. 3.1, p. 35], and we often see similar output from modern computing packages. The ANOVA table

was originated simply as a way to organize computations towards testing and seems to me to be mostly a hold-over from a time before modern computation. Still, ANOVA tables can be very convenient, especially for models with multiple factor predictors [Far14, Chaps. 16 & 17].

- **Example**, to be discussed in class.
- We use the **stats::anova** function to implement the F v R (extra sum of squares approach).
- We use the **gmodels::glh.test** function to implement the C β approach.
- Again, the “hand” computations are meant to tie things to our typeset discussion; they are generally not as computationally efficient as the computations implemented in the **anova** or **glh.test** functions.

```
> library(faraway)
> data(gala, package="faraway")
> ## Full model (Omega)
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+             gala)
> summary(lmod)
```

Call:

```
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
   data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-111.68	-34.90	-7.86	33.46	182.58

Coefficients:

```

            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.06822   19.15420    0.37   0.7154
Area        -0.02394   0.02242   -1.07   0.2963
Elevation   0.31946   0.05366    5.95 0.0000038 ***
Nearest     0.00914   1.05414    0.01   0.9932
Scruz       -0.24052   0.21540   -1.12   0.2752
Adjacent    -0.07480   0.01770   -4.23   0.0003 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Residual standard error: 61 on 24 degrees of freedom
 Multiple R-squared: 0.766, Adjusted R-squared: 0.717
 F-statistic: 15.7 on 5 and 24 DF, p-value: 6.84e-07

```

> ## The generic ANOVA Table 3.1, p. 35, of your textbook, does not
> ## decompose SSreg like the following (Sequential/Type I) ANOVA table
> ## for overall F test does.
> anova(lmod)

```

Analysis of Variance Table

```

Response: Species
          Df Sum Sq Mean Sq F value Pr(>F)
Area       1 145470 145470 39.13 0.0000018 ***
Elevation  1 65664  65664 17.66 0.00032 ***
Nearest    1     29     29  0.01  0.93007
Scruz      1 14280  14280  3.84  0.06173 .
Adjacent   1 66406  66406 17.86 0.00030 ***
Residuals 24 89231   3718
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> ## Similar ANOVA, but with explicit
> ## reduced model (omega)
> nullmod <- lm(Species ~ 1, gala)
> ## ANOVA table for overall F-test
> anova(nullmod, lmod)

```

Analysis of Variance Table

```

Model 1: Species ~ 1
Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
Res.Df   RSS Df Sum of Sq   F Pr(>F)

```

```

1      29 381081
2      24 89231  5     291850 15.7 6.8e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> ## Cb approach using F result B.9 (repeated in this chapter, above)
> p0mega<- 6; pomega<- 1
> (r<- p0mega - pomega)

[1] 5

> Cmat<- cbind(0,diag(r))
> d<- rep(0, r)
> gmodels::glh.test(reg=lmod, cm=Cmat, d=d)

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 15.699, df1 = 5, df2 = 24, p-value = 6.838e-07

```

```

> ## By ``hand'' using F Result B.9 (repeated in this chapter, above)
> x<- model.matrix(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+                     gala)
> y<- gala$Species
> bhat<- solve(xtx<-crossprod(x), crossprod(x,y))
> Cbhat<- Cmat%*%bhat
> (n<- length(y))

[1] 30

> mse<- sum((y - x%*%bhat)^2)/(n-p0mega)
> Vbhat<- mse*solve(xtx)
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

[,1]
[1,] 15.699

> pf(q=F,df1=r,df2=n-p0mega, lower=FALSE)

```

```
[,1]
[1,] 6.8379e-07

> ## Note: F in Details section of help(glh.test) is missing an inverse
> ## operation.
```

3.2.2 Testing one predictor

$H_0: \beta_{Area} = 0$.

```
> ## Using summary of full model fitted in previous chunk
> summary(lmod)

Call:
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
    data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-111.68	-34.90	-7.86	33.46	182.58

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.06822	19.15420	0.37	0.7154
Area	-0.02394	0.02242	-1.07	0.2963
Elevation	0.31946	0.05366	5.95	0.0000038 ***
Nearest	0.00914	1.05414	0.01	0.9932
Scruz	-0.24052	0.21540	-1.12	0.2752
Adjacent	-0.07480	0.01770	-4.23	0.0003 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 61 on 24 degrees of freedom
Multiple R-squared: 0.766, Adjusted R-squared: 0.717
F-statistic: 15.7 on 5 and 24 DF, p-value: 6.84e-07

```
> ## F v R approach (drop Area)
> lmods <- update(lmod, . ~ . -Area)
> anova(lmods, lmod)
```

Analysis of Variance Table

```
Model 1: Species ~ Elevation + Nearest + Scruz + Adjacent
Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1      25 93469
2      24 89231  1      4238 1.14    0.3
```

```
> ## Cb approach
> p0mega<- 6; pomega<- 5
> (r<- p0mega - pomega)

[1] 1

> Cmat<- matrix(c(0,1,0,0,0,0), nrow=r)
> d<- rep(0, r)
> gmodels::glh.test(reg=lmod, cm=Cmat, d=d)
```

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 1.1398, df1 = 1, df2 = 24, p-value = 0.2963

```
> ## By hand using F Result B.9 (using previously created objects)
> Cbhat<- Cmat%*%bhat
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

[,1]
[1,] 1.1398

> pf(q=F,df1=r,df2=n-p0mega, lower=FALSE)

[,1]
[1,] 0.29632
```

```
> ## By hand using t Result B.10 or B.11
> (sebArea <- sqrt(VCbhat[1,1]))

[1] 0.022422
```

```

> (tstat<- (Cbhat - d) / sebArea)
      [,1]
[1,] -1.0676

> 2*pt(q=abs(tstat), df=n-p0mega, lower=FALSE)
      [,1]
[1,] 0.29632

> ## Aside:  $t^2 = F$ 
> tstat^2

      [,1]
[1,] 1.1398

> F ## from above

      [,1]
[1,] 1.1398

> ## One sided/tailed test?

```

```

> ## Generally, a very different test of bArea = 0 (different full model)
> summary(lm(Species ~ Area, data=gala))

```

Call:

```
lm(formula = Species ~ Area, data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-99.50	-53.43	-29.04	3.42	306.14

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	63.7829	17.5244	3.64	0.00109 **
Area	0.0820	0.0197	4.16	0.00027 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 91.7 on 28 degrees of freedom

```
Multiple R-squared:  0.382, Adjusted R-squared:  0.36
F-statistic: 17.3 on 1 and 28 DF,  p-value: 0.000275
```

3.2.3 Testing a pair of predictors

$H_0: \beta_{Area} = \beta_{Adjacent} = 0.$

```
> ## Drop Area and Adjacent
> lmods <- update(lmod, . ~ . - Area - Adjacent)
> anova(lmods, lmod)

Analysis of Variance Table

Model 1: Species ~ Elevation + Nearest + Scruz
Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     26 158292
2     24  89231  2      69060 9.29  0.001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## Cb approach using F Result B.9 (using previously created object)
> p0omega<- 6; pomega<- 4
> (r<- p0omega - pomega)
```

```
[1] 2
```

```
> Cmat<- matrix(c(0, 1, 0, 0, 0, 0,
+                  0, 0, 0, 0, 0, 1),
+                  nrow=r, byrow=TRUE)
> d<- rep(0, r)
> gmodels::glh.test(reg=lmod, cm=Cmat, d=d)
```

```
Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 9.2874, df1 = 2, df2 = 24, p-value = 0.00103
```

```
> ## By hand using F Result B.9 (using previously created objects)
> Cbhat<- Cmat%*%bhat
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

      [,1]
[1,] 9.2874

> pf(q=F,df1=r,df2=n-p0mega, lower=FALSE)

      [,1]
[1,] 0.0010297
```

3.2.4 Testing a subspace

$$H_0: \beta_{Area} = \beta_{Adjacent}$$

```
> lmods <- lm(Species ~ I/Area+Adjacent + Elevation + Nearest + Scruz, gala)
> anova(lmods, lmod)

Analysis of Variance Table

Model 1: Species ~ I/Area + Adjacent + Elevation + Nearest + Scruz
Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     25 109591
2     24  89231  1     20360 5.48  0.028 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## Cb approach using F Result B.9 (using previously created object)
> p0mega<- 6; pomega<- 5
> (r<- p0mega - pomega)

[1] 1

> Cmat<- matrix(c(0,1,0,0,0,-1), nrow=r)
> d<- rep(0, r)
> gmodels::glh.test(reg=lmod, cm=Cmat, d=d)
```

```

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 5.476, df1 = 1, df2 = 24, p-value = 0.02793

> ## By hand Cb approach using F Result B.9 (using previously created objects)
> Cbhat<- Cmat%*%bhat
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

      [,1]
[1,] 5.476

> pf(q=F,df1=r,df2=n-p0mega, lower=FALSE)

      [,1]
[1,] 0.027926

> ## By hand using t Result B.10 (but not B.11)
> (sebAeqE <- sqrt(VCbhat[1,1]))

[1] 0.021737

> (tstat<- (Cbhat - d) / sebAeqE)

      [,1]
[1,] 2.3401

> 2*pt(q=abs(tstat), df=n-p0mega, lower=FALSE)

      [,1]
[1,] 0.027926

> ## Aside: t^2 = F
> tstat^2

      [,1]
[1,] 5.476

> F ## from above

      [,1]
[1,] 5.476

> ## One sided/tailed test?

```

3.2.5 Non-Zero Hypothesized Value H_0 : $\beta_{Elevation} = 0.5$

```
> lmods <- lm(Species ~ Area + offset(0.5 * Elevation) + Nearest + Scruz +
+               Adjacent, gala)
> anova(lmods, lmod)
```

Analysis of Variance Table

Model 1: Species ~ Area + offset(0.5 * Elevation) + Nearest + Scruz + Adjacent					
Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent					
Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	25	131312			
2	24	89231	1	42081	11.3 0.0026 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1					

```
> ## Cb approach using F Result B.9 (using previously created object)
> p0mega<- 6; pomega<- 5
> (r<- p0mega - pomega)
```

[1] 1

```
> Cmat<- matrix(c(0,0,1,0,0,0), nrow=r)
> d<- 0.5
> gmodels:::glh.test(reg=lmod, cm=Cmat, d=d)
```

Test of General Linear Hypothesis

Call:

```
gmodels:::glh.test(reg = lmod, cm = Cmat, d = d)
F = 11.318, df1 = 1, df2 = 24, p-value = 0.002574
```

```
> ## By hand Cb approach using F Result B.9 (using previously created objects)
> Cbhat<- Cmat%*%bhat
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

[,1]
[1,] 11.318
```

```
> pf(q=F,df1=r,df2=n-p0mega, lower=FALSE)
[1,]
[1,] 0.0025738

> ## By hand using t Result B.10 or B.11
> (sebE0.5 <- sqrt(VCbhat[1,1]))
[1] 0.053663

> (tstat<- (Cbhat - d) / sebE0.5)

[1,]
[1,] -3.3643

> 2*pt(q=abs(tstat), df=n-p0mega, lower=FALSE)
[1,]
[1,] 0.0025738

> ## Aside:  $t^2 = F$ 
> tstat^2

[1,]
[1,] 11.318

> F ## from above

[1,]
[1,] 11.318

> ## One sided/tailed test?
```

- Whew!
- Questions/discussion?
- Two-sided vs. one-sided tests and one- or two-tails. (TBD)

3.2.6 Tests we cannot do (in INF 511)

- Non-linear hypotheses like $H_0: \beta_{Area}\beta_{Adjacent} = 1$. (However, we might appeal to the Delta method and asymptotic normality. See, e.g., the `n1WaldTest` library, used in INF 512.)
- Non-nested models like (Area and Elevation) vs (Area, Adjacent and Cruz). But, see model selection criteria in [Far14, Chap. 10].
- Models with different data (that might arise from the way different predictors are missing in different ways), even if the models otherwise appear nested.

Power

Power is not really an inferential endeavor; it's more a planning and design activity. So, it does not really fit into [Far14, Chap. 3]. But, power computations, at least, share some expressions we've been using here. And, you need to know a bit more about power. We only scratch the surface here. (We may skip this section in favor of other, fun stuff.)

- **Power.** From Table 3.1, we see that the power to reject a null hypothesis is $\text{power} = 1 - \beta$, **one minus the probability of a type II error**. In other words, power is **the probability of rejecting the null hypothesis when the true distribution differs from the null distribution**, thus, unlike α , power ($1 - \beta$) may not be computed with the null distribution but with the (assumed, of course) true distribution. (Note that, α , the probability of a type I error, is also a probability of rejecting the null hypothesis but when the null distribution is true, and may be computed using the null distribution as we know, hopefully.)
- **Power Depends on Things.** Generally speaking, the power to reject a null distribution is a **function of**

1. **Effect.** How “far” the true distribution is from that under the null hypothesis. I will loosely refer to this as an “**effect**,” though we will have a relatively specific definition of effect in later chapters.
2. The **variability of the data**, represented by σ^2 in our models. The power to detect an effect decreases with increased “noise,” and vice versa.
3. The **sample size**, n . As we obtain more information in the form of an increasing (random) sample size, n , the power to detect an effect increases, and vice versa.
4. **Type I Error Rate.** α . The probability of a type I error determines the **rejection region**. A large α means a larger rejection region. Thus, because power is a probability of rejection, power increases with α . In other words, the probability of a type II error decreases with an increase in type I error probability, and vice versa. Thus, we have a trade-off between α and β . Want both error probabilities small? Choose a larger sample size or a smaller effect.
5. **Specify All But One.** Typically, we specify all of the above, but one, which we then solve for. Often, we solve for sample size to achieve detect a desired effect with high power at a specified α and guesstimated σ .

Non-Central F Distribution

- **When Null is Not True.** Under the alternative hypothesis (and our usual linear model assumptions), our F statistic (Result B.9) no longer follows a (central) F but, instead, follows a **non-central** F distribution, with **non-centrality parameter** (ignoring a few details)

$$\begin{aligned} ncp &= (\mathbf{C}\boldsymbol{\beta}_1 - \mathbf{C}\boldsymbol{\beta}_0)^t \text{Var}(\hat{\boldsymbol{\beta}})^{-1} (\mathbf{C}\boldsymbol{\beta}_1 - \mathbf{C}\boldsymbol{\beta}_0) \\ &= (\mathbf{C}\boldsymbol{\beta}_1 - \mathbf{C}\boldsymbol{\beta}_0)^t (\sigma^2 \mathbf{C}(\mathbf{X}^t \mathbf{X})^{-1} \mathbf{C}^t)^{-1} (\mathbf{C}\boldsymbol{\beta}_1 - \mathbf{C}\boldsymbol{\beta}_0) \\ &= (\mathbf{d}_1 - \mathbf{d}_0)^t (\sigma^2 \mathbf{C}(\mathbf{X}^t \mathbf{X})^{-1} \mathbf{C}^t)^{-1} (\mathbf{d}_1 - \mathbf{d}_0) \end{aligned}$$

where, under the null hypothesis, we specify

$$\mathbf{C}\boldsymbol{\beta}_0 = \mathbf{d}_0,$$

but, for the power computation purposes, we assume that the true model is, instead,

$$\mathbf{C}\boldsymbol{\beta}_1 = \mathbf{d}_1,$$

the difference being, of course,

$$\mathbf{d}_1 - \mathbf{d}_0.$$

Unfortunately, ncp depends on the unknown error variance (as mentioned), σ^2 . And, as mentioned, it depends on the “effect,” or difference between null and alternative models (and it depends on \mathbf{X} (regression or design matrix), which masks dependence on n).)

- **When Null is True.** If the null is true, then, of course, $\mathbf{d}_1 = \mathbf{d}_0$ and $ncp = 0$, and our F statistic follows our more familiar **central** F distribution. (The remaining parameters, $df_1 = r = p_\Omega - p_\omega$ and $df_2 = n - p_\Omega$, are the same, central or non-central.)

Example: Power for Effect of One Predictor

- **Simple E.g..** To introduce power computations, suppose that we are interested in the power to detect an **effect of elevation** in our running island biogeography example with $k = 5$ covariates. More precisely, suppose we have

$$\begin{aligned} H_0: \mathbf{C}\boldsymbol{\beta} &= \mathbf{0} \\ H_1: \mathbf{C}\boldsymbol{\beta} &\neq \mathbf{0}, \end{aligned}$$

where $\mathbf{C} = (0, 0, 1, 0, 0, 0)^t$ picks off β_{elev} and $\mathbf{0} = 0$. That is,

$$H_0: \beta_{elev} = 0 \quad \text{vs.} \quad H_1: \beta_{elev} \neq 0,$$

the remaining 5 parameters in our regression model not constrained.

- **Effect.** Further, suppose $\beta_{elev} = 0.05$ is considered practically important, regardless of sign, as determined from the expertise of island biogeographers. Thus,

$$C\beta_1 = d_1 = 0.5.$$

- **Variability.** And, suppose we know, from experience or the literature, that the error standard deviation is $\sigma \approx 60$ species (we may want to increase this to ensure sample size computations are conservative).
- **Sample Size and Design.** For simplicity, we will assume our current \mathbf{X} matrix, which is based on $n = 30$ observations.
- **Type I Error Rate.** $\alpha = 0.05$.

```
> data(gala, package="faraway")
> ## Refit previous Galapagos model to access objects
> lmod<- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+            data=gala)
> pOmega<- 6; pomega<- 5 ## model sizes for this e.g. when testing a
> ## single df (r=1), i.e., testing omission of
> ## single predictor, e.g., elevation
> (r <- pOmega - pomega) ## single df (for omitting elev)

[1] 1

> Cmat<- matrix(c(0,0,1,0,0,0), nrow=r) ## picks off b_elev
> b0<- rep(0,pOmega) ## only elev entry matters upon C%*%b0 in this e.g
> (d0<- Cmat%*%b0) ## hyp mean under null

[,1]
[1,] 0

> b1<- b0; b1[3]<- 0.05 ## only elev entry matters upon C%*%b1 in this e.g
> (d1<- Cmat%*%b1) ## specific hyp mean under alternative

[,1]
[1,] 0.05
```

```

> sig<- 60 ## error std dev (sqrt sigma^2) from educated guess, from
>           ## similar studies (may want to inflate to be conservative,
>           ## i.e., get larger sample size or smaller power or smaller
>           ## effects...)
> X<- model.matrix(lmod)
> (n<- dim(X)[1])

[1] 30

> XtX<- crossprod(X)
> ## Non-centrality parameter for R's F distribution
> (ncpF<- t(d1-d0)%*%solve(Cmat%*%solve(XtX)%*%t(Cmat))%*%(d1-d0) / sig^2)

[,1]
[1,] 0.8966

> ## Reject for F >= Fcrit determined by Pr(Type I) = alpha
> alpha<-0.05 ## type I error prob
> (Fcrit<- qf(p=alpha, df1=r, df2=n-p0mega, lower=FALSE))

[1] 4.2597

> ## Power Pr(reject Ho / ncp) = Pr(F >= Fcrit / ncp)
> pf(q=Fcrit, df1=r, df2=n-p0mega, ncp=ncpF, lower=FALSE)

[1] 0.14878

```

Thus, we have a power of about $1 - \beta = 0.15$ to detect a departure of $\beta_{elev} = 0.5 (\pm)$ from the null, $\beta_{elev} = 0$, assuming $\sigma^2 = 60^2$, $n = 30$ and $\alpha = 0.05$ (and our current \mathbf{X}).

pwr Package

- **Power Software.** Here, we go only a little bit further, still only scratching the surface of power analysis. We use the R package **pwr**, whose functionality appears similar to G*Power (not an R package), which uses

a nice GUI; both seem to follow Cohen's text [Coh88]. In particular, we use `pwr::pwr.f2.test`, which performs power analyses for the general linear model. Search the CRAN Task Views for Clinical Trials and for Experimental Design for “power” for several other packages (<https://cran.r-project.org/web/views/>).

- **More or Less Intuitive Effects.** For many linear models, especially multi-factor predictor models [Far14, Chaps. 16 & 17], and simpler models, effect size is relatively straightforward to specify in terms of familiar parameters, as illustrated in our previous elevation example for the biogeography data. More generally, using our regression model, however, as we have been considering so far, specifying effects may not as straightforward or intuitive. In short, we will specify an effect size in terms of R^2 , which is often attributed to [Coh88]. I need to do more work to show how effects in terms of β translate to effects in terms of R^2 (TBD!). For now, notice the connection between the overall F test statistic and R^2 to help convince you of the connection (without further explanation).
- **Connection to R^2 .** It is perhaps easier to use the RSS (F v R or extra sum of squares) approach to show a connection to R^2 (instead of the $C\beta$ approach), which moves us towards specifying “effects” for power analysis. (Again, this power analysis section is still under development.)

- For the overall F test, we see

$$\begin{aligned}
 F &= \frac{(RSS_\omega - RSS_\Omega)/(p_\Omega - p_\omega)}{RSS_\Omega/(n - p_\Omega)} \\
 &= \frac{(RSS_\omega - RSS_\Omega)}{RSS_\Omega} \left(\frac{n - p_\Omega}{p_\Omega - p_\omega} \right) \\
 &= \frac{\frac{(RSS_\omega - RSS_\Omega)}{RSS_\omega}}{\frac{RSS_\Omega}{RSS_\omega}} \left(\frac{n - p_\Omega}{p_\Omega - p_\omega} \right) \\
 &= \frac{\frac{(RSS_\omega - RSS_\Omega)}{RSS_\omega}}{\frac{RSS_\omega - (RSS_\omega - RSS_\Omega)}{RSS_\omega}} \left(\frac{n - p_\Omega}{p_\Omega - p_\omega} \right) \\
 &= \frac{\frac{(TSS - RSS)}{TSS}}{\frac{TSS - (TSS - RSS)}{TSS}} \left(\frac{n - p_\Omega}{p_\Omega - p_\omega} \right) \\
 &= \frac{R^2}{1 - R^2} \left(\frac{n - p_\Omega}{p_\Omega - p_\omega} \right).
 \end{aligned}$$

A similar expression may be made for other testing situations, aside from the overall F test. (Again, TBD.)

- Options.

- **pwr::pwr.f2.test** named options **u** and **v** are the F distribution's numerator and denominator degrees of freedom, previously denoted df_1 and df_2 . In terms of our notation, $u = r = p_\Omega - p_\omega$ and $v = n - p_\Omega$. This allows us to specify the sizes of null and alternative models and sample size, n .
- The effect size named option is **f2**, which is specified in terms of R^2 , as mentioned, $f2 = R^2/(1 - R^2)$, where R^2 is interpreted as the **increase** in R^2 from a null model to a particular alternative model, which we may formalize as

$$H_0 : R^2 = 0 \quad \text{vs} \quad H_1 : R^2 > 0.$$

- We can also specify power and α .

- Generally, as mentioned previously, we omit one of the above options to have `pwr::pwr.f2.test` compute it for us, as illustrated, below.

Example: A Priori Power Analysis

Here, we specify an effect, power, and α for two models, which we may think of as a null/reduced model, with $p_\omega = 1$ parameter (intercept, no covariates), and a fuller model, with $k = 5$ covariates for $p_\Omega = 6$. Assume, from experience, we expect/are interested in our $k = 5$ covariate model giving (**effect**) $R^2 = 0.30$ (modest, I know), which we may view as an increase in R^2 from the null model ($R^2 = 0$). Thus, as indicated, above, we may think of the corresponding hypotheses,

$$H_0 : R^2 = 0 \quad \text{vs} \quad H_1 : R^2 = 0.3$$

(generally, we have to specify a particular alternative to compute power, etc.).

What is the sample size required to achieve power = 0.8 at level $\alpha = 0.05$?

We call this an **a priori** power analysis because it is the sort of thing we would do to help us determine our sample size, n , **before** collecting the data. Note that σ^2 (and \mathbf{X}) is somehow accounted for in the R^2 approach to power (again, more TBD!).

```
> ## install.packages("pwr", depend=TRUE)
>
> ## Relationship of our notation to pwr package notation:
> ## u = df1 = pOmega - pomega, v = df1 = n - pOmega
> ## (or u = df1 = pF - pR, v = df2 = n - pF)
>
> #####
>
> ## Example 1. A priori sample size example.
```

```

>
> ## H0 : R2=0, H1 : R2>0, k = 5 covariates.
>
> ## What sample size is required to detect (additional) R^2=0.30 (over
> ## and above reduced model, null here) with power 0.8 at alpha = 0.05?
>
> pOmega<- 6; pomega<- 1 ## full and reduced model sizes
> alpha<- 0.05; beta<- 0.2 ## type I and II error probs
> R2<- 0.3 ## determine sample size to detect this R2
> (res<- pwr::pwr.f2.test(u = pOmega-pomega, f2 = R2/(1 - R2),
+                           sig.level = alpha, power = 1-beta))

```

Multiple regression power calculation

```

u = 5
v = 29.891
f2 = 0.42857
sig.level = 0.05
power = 0.8

```

```

> ## v = n - pOmega --> n = v + pOmega
> (n<- ceiling(res$v + pOmega))

```

[1] 36

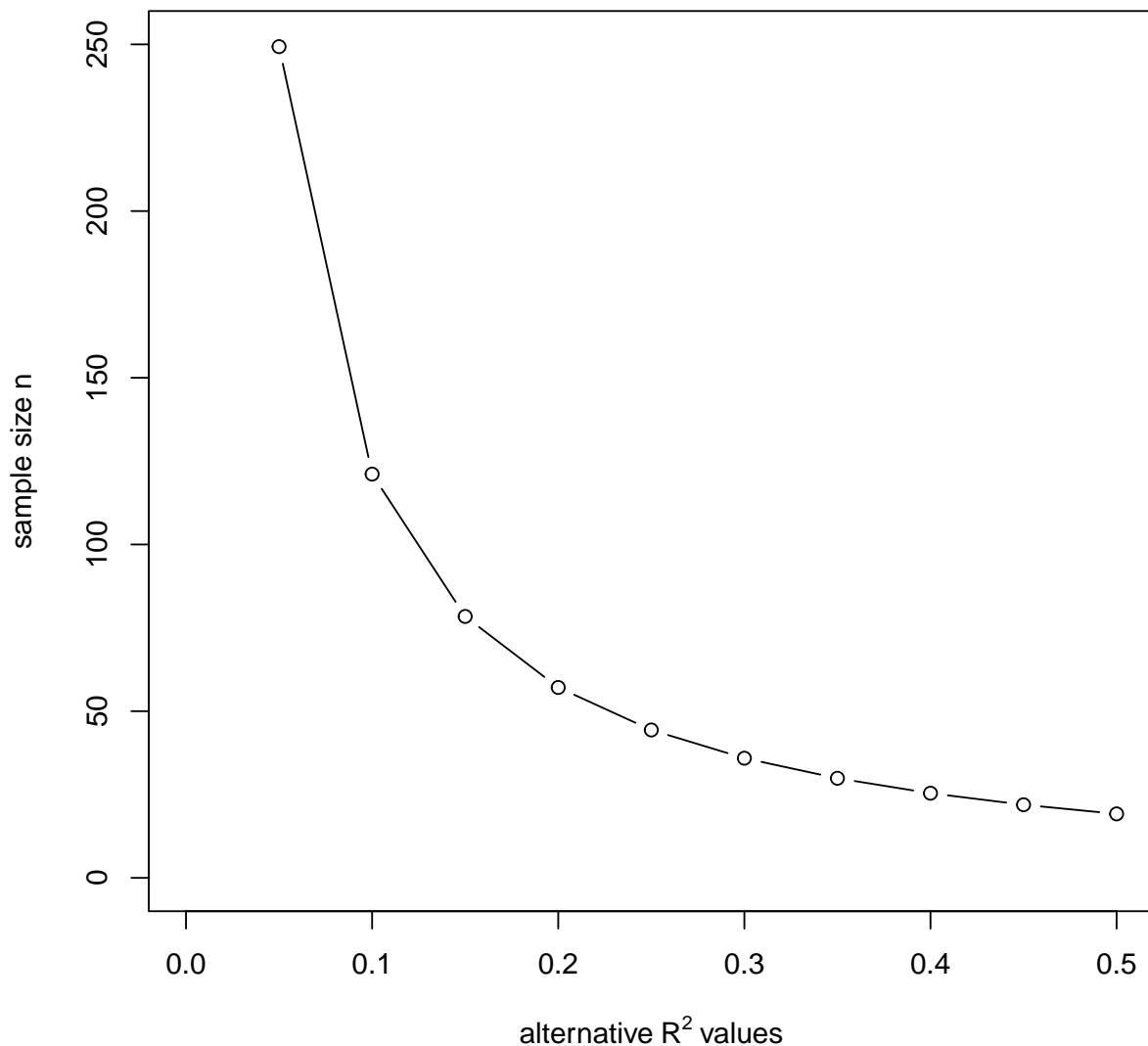
- Thus, we need $n = 36$ observations to obtain (an increase of) $R^2 = 0.30$ (over and above the null model, $R^2 = 0$), assuming $1 - \beta = 0.80$, $\alpha = 0.05$; a very modest sample size for a modest R^2 .
- What about other “effects” and sample sizes? Let’s compute the required sample sizes for a sequence of R^2 values (effects), with other quantities held at their values in the previous code.

```

> #####
>
> ## Example 2. Similar to above, but for sequence of alternative R^2

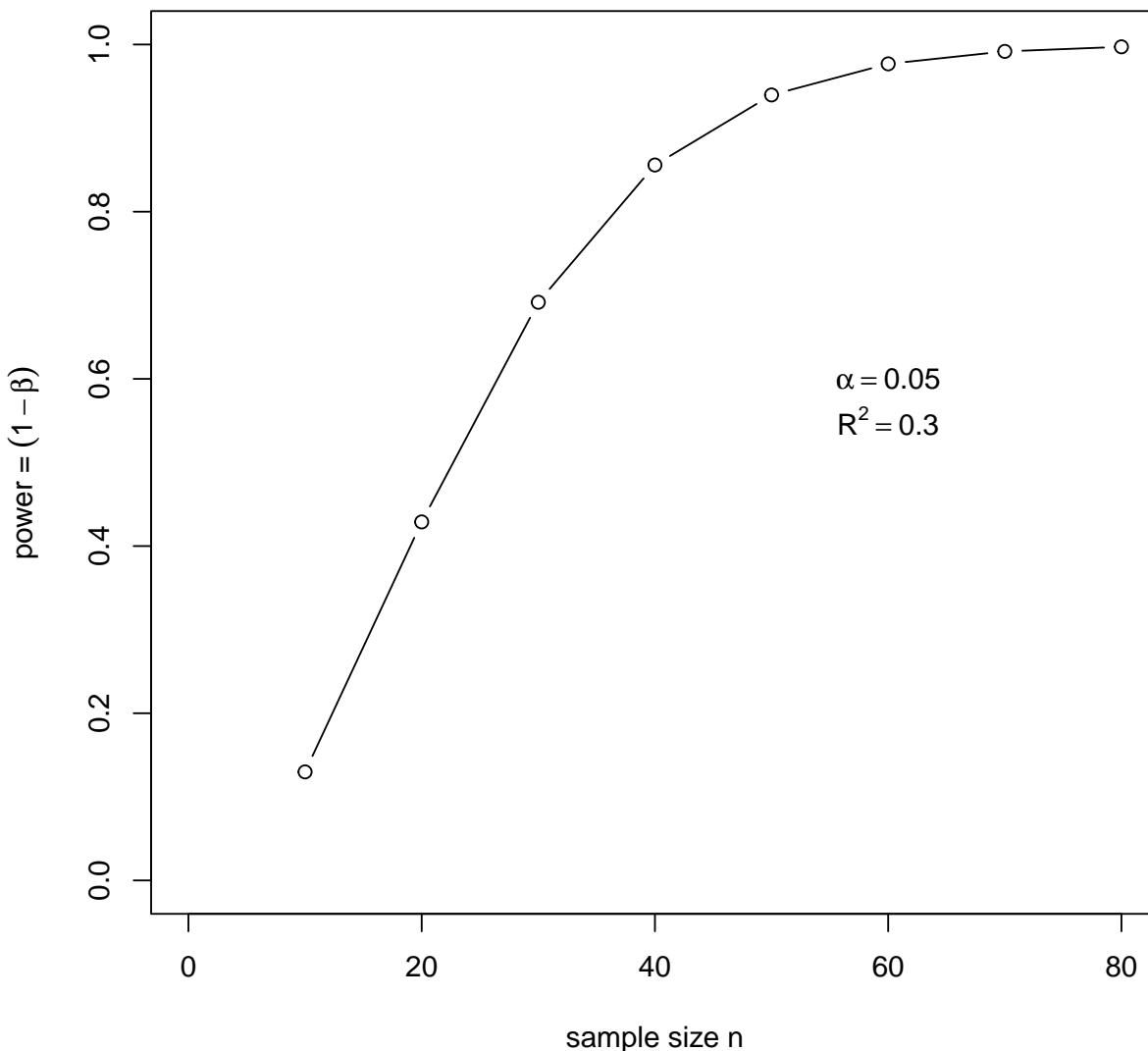
```

```
> ## values.  
>  
> ## A priori sample size determination to detect increase in R2 from  
> ## reduced model ( $H_0:R^2=0$ ) to various alternative models resulting in  
> ## additions of  $R^2$  in  $H_1:R^2 > 0$  at  $\alpha=0.05$  with power =  $1-\beta =$   
> ## 0.8. Same model sizes as previous chunk.  
>  
>  
> p0mega<- 6; pomega<- 1 ## full and reduced model sizes  
> alpha<- 0.05; beta<- 0.2 ## type I and II error probs  
> R2<- seq(0.05,0.5,0.05) ## determine sample sizes to detect these R2  
> ## additions over and above small  
> ## model (here, null model  $R^2=0$ )  
> n<- NULL  
> for(r2 in R2) {  
+   res<-pwr::pwr.f2.test(u = p0mega-pomega, power=1-beta,  
+                           f2 = r2/(1 - r2),  
+                           sig.level = alpha)  
+   n<-c(n, res$v + p0mega)  
+ }  
>  
> plot(n ~ R2, type="b",  
+       xlim=c(0,0.5), ylim=c(0,250),  
+       ylab=expression("sample size" ~ n),  
+       xlab=expression("alternative" ~ R^2 ~ "values"))
```



Let's create a different sort of power curve from that above, now as a function of sample size, fixing $R^2 = 0.30$, with other options held at their previous values.

```
> #####  
>  
> ## Example 3. Power curve. (as function of sample size)  
>  
> ## Post hoc power analysis for a given effect size, various  
> ## sample sizes, i.e., power curve. H0:R^2=0, H1:R^2 = 0.30  
> p0mega<- 6; pomega<- 1 ## full and reduced regression model size  
> alpha<- 0.05; beta<- 0.2 ## type I and type II error probs  
> R2<- 0.30 ## determine sample size to detect this R^2 values  
> f2<- R2/(1-R2) ## Cohen 1988 effect size  
> n<- seq(10, 80, 10)  
> res<- pwr::pwr.f2.test(u = p0mega-pomega,  
+                           v = n - p0mega,  
+                           f2 = f2,  
+                           sig.level = alpha)  
> plot(res$power ~ n, type="b",  
+       ylim=c(0,1), xlim=c(0,80),  
+       ylab=expression("power =" ~ (1-beta)),  
+       xlab="sample size n")  
> text(60,0.6,labels=expression(alpha == 0.05))  
> text(60,0.55,labels=expression(R^2 == 0.30))
```



Example: Post Hoc Power Analysis

Let's say we already have $n = 30$ observations and are wondering about our power to detect certain effects, again in terms of R^2 . Now, we specify sample size, and we omit power from the named options to have the software compute

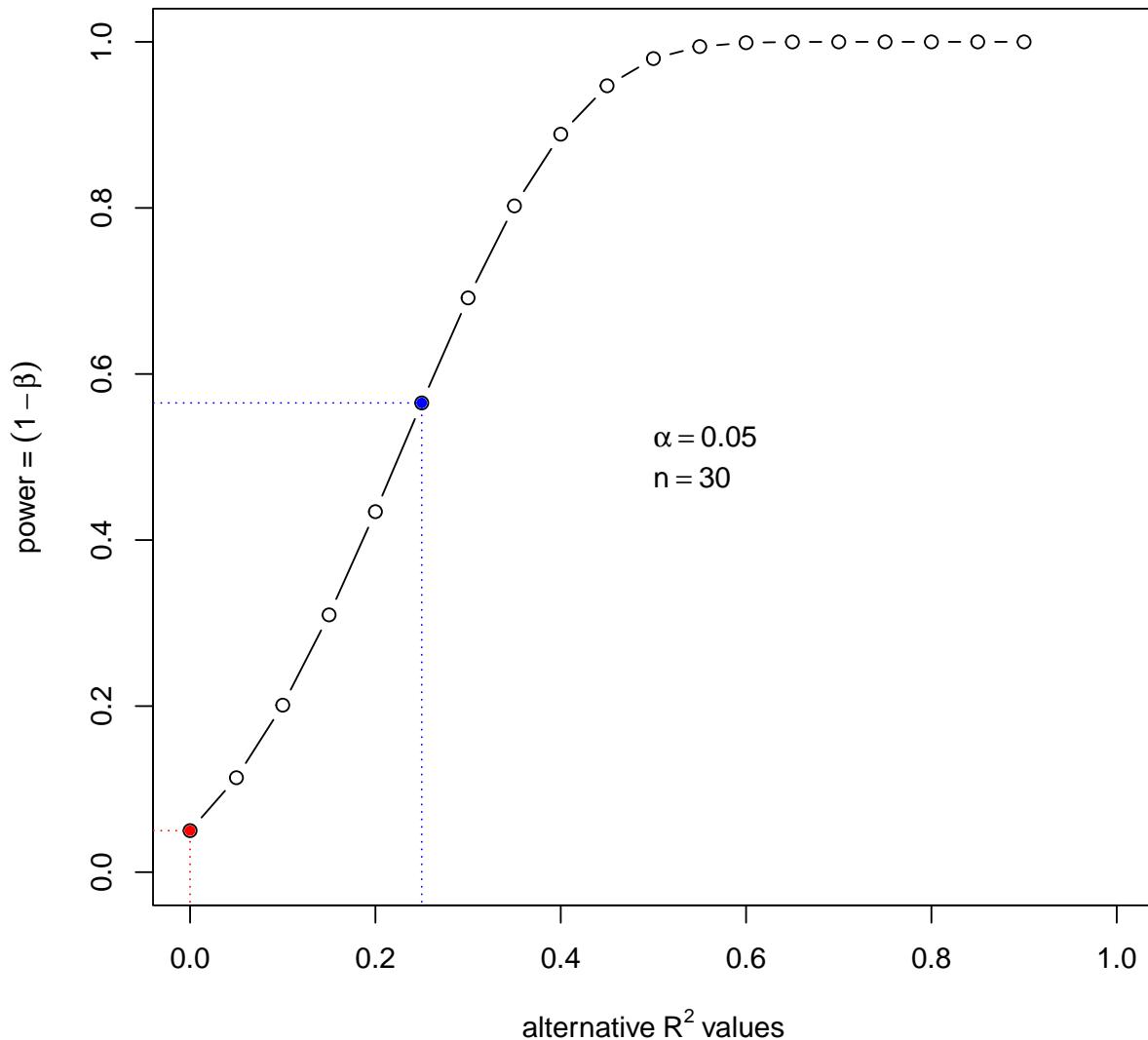
power corresponding to the effects. We refer to this as a **post hoc** analysis because the scenario assumes we have already collected the data (and likely already looked at it and obtained results) before doing the power analysis.

```
> #####
>
> ## Example 4a. Power curve. (as function of effect size)
>
> ## Post hoc power analysis for a given sample size, various
> ## additional R^2 effect sizes, i.e., power curve. H0:R^2=0, H1:R^2 > 1
> p0mega<- 6; pomega<- 1 ## full and reduced regression model size
> alpha<- 0.05 ## type I error prob
> n<- 30 ## sample size
> R2<- seq(0.0,0.90,0.05) ## determine power to detect these R^2 values
> f2<- R2/(1-R2) ## Cohen 1988 effect size(s)
> res<- pwr::pwr.f2.test(u = p0mega-pomega, v = n - p0mega,
+                         f2 = f2,
+                         sig.level = alpha)
> plot(res$power ~ R2, type="b",
+       ylim=c(0,1), xlim=c(0,1),
+       ylab=expression("power" ~ (1-beta)),
+       xlab=expression("alternative" ~ R^2 ~ "values"))
>
> ## Add Pr(reject H0 | H0 true) = Pr(type I error) = alpha (red)
> points(0,alpha, col="red", pch=20)
> segments(0,-1,0,alpha,col="red",lty=3)
> segments(-1,alpha,0,alpha,col="red",lty=3)
>
> ## Add power = 1 - Pr(reject H0 | H1:R^2=0.25) (blue)
> R2<- 0.25; f2<- R2 / (1-R2)
> (Fcrit<- qf(p=alpha, df1=p0mega-pomega, df2=n-p0mega, lower=FALSE))
[1] 2.6207

> (power<- pf(Fcrit,df1=p0mega - pomega, df2=n-p0mega,
+                 ncp= n*f2, lower=FALSE))
[1] 0.56514

> points(R2, power, col="blue", pch=20)
```

```
> segments(R2,-1,R2,power,col="blue",lty=3)
> segments(-1,power,R2,power,col="blue",lty=3)
>
> text(0.5,0.525,labels=expression(alpha == 0.05), adj=0)
> text(0.5,0.475,labels=expression(n == 30), adj=0)
```



```
> #####
```

Let's "zoom in" on one (two, really) of the effects in the above plot, indicated by red and blue. We've seen the plot below, before, right (sans code)?

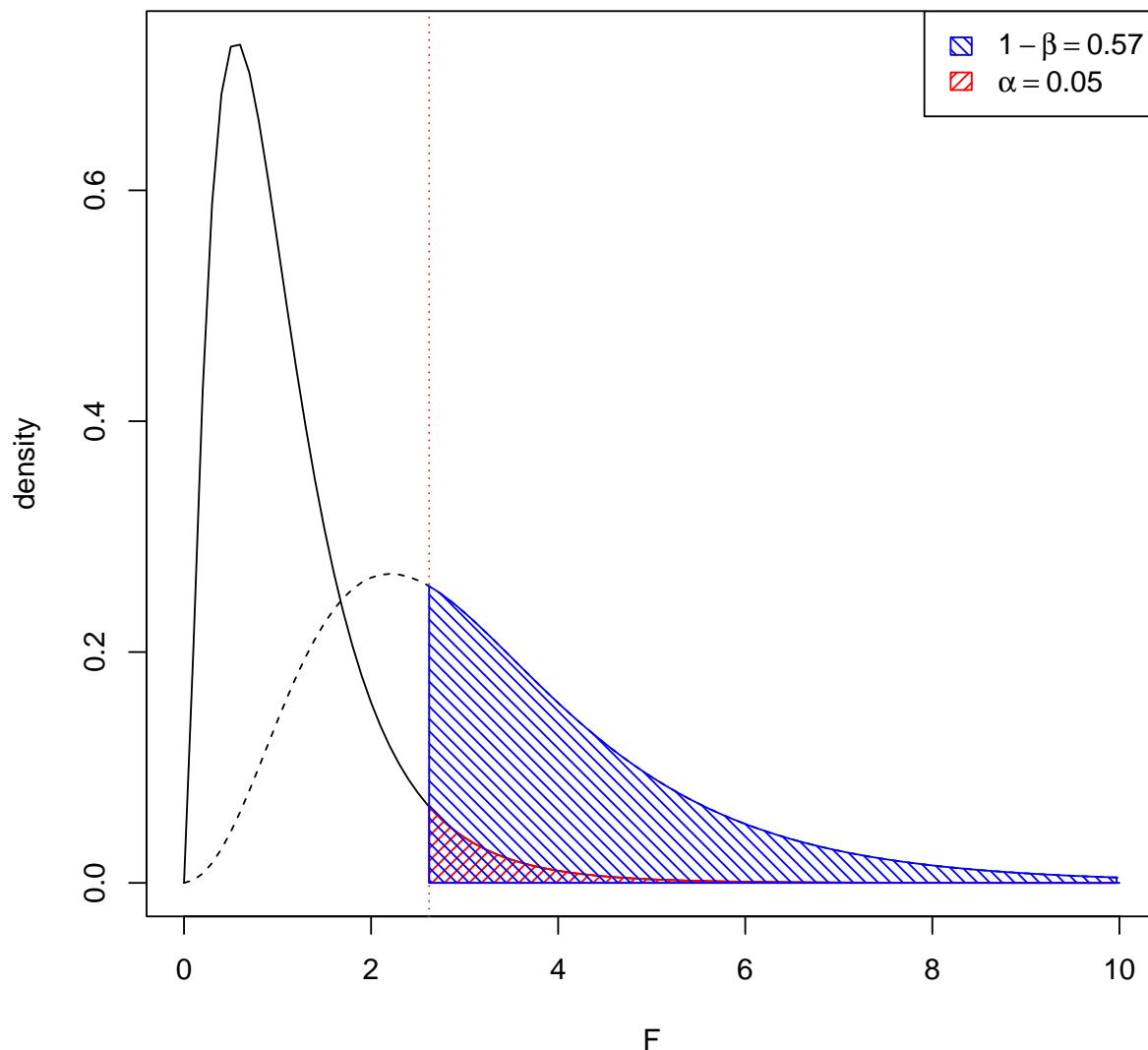
```

> #####
>
> ## Example 4b.
>
> ## Plot alpha and power for H1:R^2 = 0.25 (one of the values in 4a)
> R2<- 0.25; f2<- R2 / (1-R2)
>
> ## H0:R^2=0 central F pdf
> curve(df(x,df1=p0mega - pomega, df2=n-p0mega), ## pdf under H0
+       from=0, to = 10, ylab="density", xlab="F")
>
> ## Add alpha and Fcrit rejection region of F values (red)
> Fcrit<- qf(1-alpha, df1=p0mega - pomega, df2=n-p0mega)
> abline(v=Fcrit, lty=3, col="red")
> Fseq<- seq(Fcrit, 10, 0.05)
> pdfFseq<- df(Fseq,df1=p0mega - pomega, df2=n-p0mega)
> polygon(x=c(head(Fseq,1),Fseq,tail(Fseq,1)), y=c(0,pdfFseq,0),
+           col="red", density=20, angle=45)
>
> ## H1:R^2=0.25 non-central F pdf
> curve(df(x,df1=p0mega - pomega, df2=n-p0mega, ncp= n*f2),
+       from=0, to = 10, lty=2, add=TRUE)
>
> ## Add power (blue)
> pdfFseq<- df(Fseq,df1=p0mega - pomega, df2=n-p0mega, ncp= n*f2)
> polygon(x=c(head(Fseq,1),Fseq,tail(Fseq,1)), y=c(0,pdfFseq,0),
+           col="blue", density=20, angle=135)
> (power<- pf(Fcrit,df1=p0mega - pomega, df2=n-p0mega,
+             ncp= n*f2, lower=FALSE)) ## same as in 4a

[1] 0.56514

> legend("topright",
+        legend=c(expression(1-beta == 0.57),
+                 expression(alpha == 0.05)),
+        density=c(20,20),fill=c("blue","red"),
+        angle=c(135,45), border=c("blue","red"))

```



Test Sidedness, Distribution Tails & Power

- **Two-Sided Alternative.** The hypotheses discussed so far are called **two-sided** in that the alternative, $H_1: C\beta \neq d$, suggests rejection for

large $C\beta$, **positive or negative**, which makes more sense when $C\beta$ is a scalar, i.e., when C consists of a single ($r = 1$) row and $C\beta$ is a number, not a vector, as when testing, e.g., a single parameter $H_1: C\beta = \beta_j \neq 0$, as illustrated previously with test involving β_{area} and with our introductory power computations using β_{elev} .

- **Two-Sided Single df t or F Test.** In the two-sided, scalar case, we may use an F test or a t test, again as illustrated previously. This is because $t^2(df = n - p) = F(df_1 = 1, df_2 = n - p)$, and the **two-tailed rejection region** of the t distribution “folds over” to a **one-tailed** rejection region for the F distribution, with the same with type I error probability, α , folded over, too, as depicted in the following figure, which is characteristic of a such a (numerator) single ($r = 1$) degree of freedom ($df_1 = r = 1$) test, like in the Galapagos example, above, $H_0: \beta_{elev} = 0$.

```
> pOmega<- 6; pomega<- 5 ## like in  $H_0: b_{elev} = 0$ , above.
> alpha<- 0.05 ## type I error prob
> (r= pOmega - pomega) ## single (numerator) df test

[1] 1

> n<- 30
>
> ## t plot (notice alpha/2 splits alpha into lo/up tails
> curve(dt(x, df=n-pOmega), ## pdf under  $H_0$ 
+       from=-3, to = 9, ylab="density", xlab="t or F",
+       col="red")
> (tcrit<- qt(1-alpha/2, df=n-pOmega)) ## same as F df2 (denom)

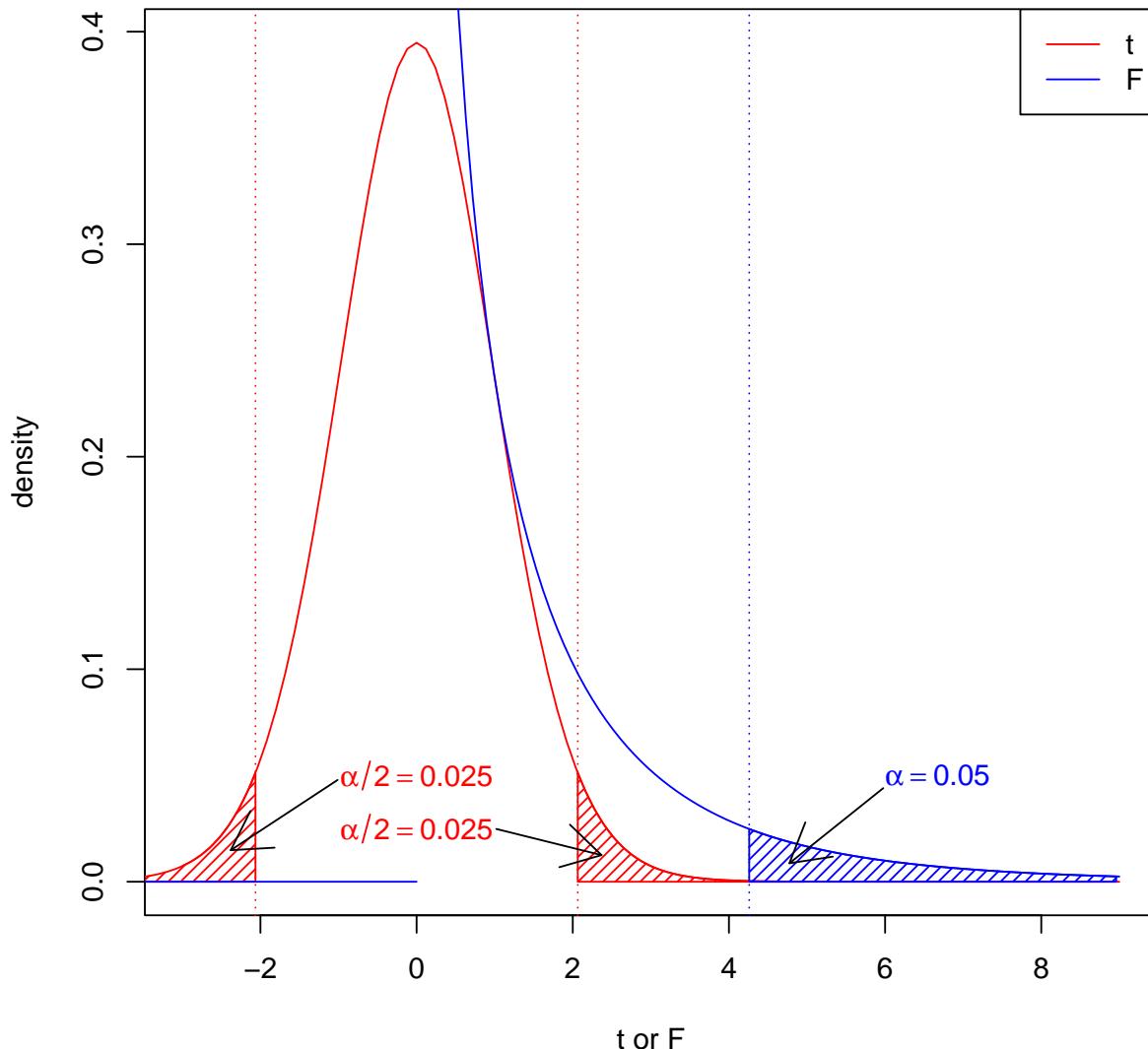
[1] 2.0639

> tseq<- seq(tcrit, 5, 0.05)
> pdftseq<- dt(tseq, df=n-pOmega)
> polygon(x=c(head(tseq,1),tseq,tail(tseq,1)),
+           y=c(0,pdftseq,0),
+           col="red", density=20, angle=45)
> polygon(x= -rev(c(head(tseq,1),tseq,tail(tseq,1))),
```

```
+           y=c(0,rev(pdftseq),0),
+           col="red", density=20, angle=45)
> abline(v=c(-1,1)*tcrit, lty=3, col="red")
> text(0,0.05,expression(alpha/2 == 0.025), col="red")
> text(0,0.025,expression(alpha/2 == 0.025), col="red")
> text(6,0.05,expression(alpha == 0.05), col="blue", adj=0)
>
> ## F plot
> curve(df(x,df1=r, df2=n-p0mega), ## pdf under H0
+        from=0, to = 9, add=TRUE, col="blue")
> segments(-5,0,0,0, col="blue")
> (Fcrit<- qf(1-alpha, df1=p0mega - pomega, df2=n-p0mega))

[1] 4.2597

> abline(v=Fcrit, col="blue", lty=3)
> Fseq<- seq(Fcrit, 9, 0.05)
> pdfFseq<- df(Fseq,df1=p0mega - pomega, df2=n-p0mega)
> polygon(x=c(head(Fseq,1),Fseq,tail(Fseq,1)), y=c(0,pdfFseq,0),
+           col="blue", density=20, angle=45)
>
> ## arrowends<- as.data.frame(locator(n=6))
> arrows(-1.0116, 0.0478939, -2.3803, 0.0148401)
> arrows(1.0206, 0.0248331, 2.3685, 0.0125340)
> arrows(5.9768, 0.0440504, 4.7741, 0.0086905)
>
>
> legend("topright",
+        legend=c(expression(t), expression(F)),
+        lty=1,col=c("red","blue"))
```



```
> round(tcrit^2,10) == round(Fcrit,10)
[1] TRUE
```

- **Increased Power of One-Sided Test.** If we are able to specify a one-sided alternative **a priori**, before looking at the data, then we can gain

power to detect model/hypothesis differences using a t test. (F tests are not convenient for 1-sided alternatives.) For example, let's compare the power for a 1-sided alternative, $\beta_{elev} > 0$, vs. 2-sided alternative, $\beta_{elev} \neq 0$, the latter already considered in a previous, introductory power computation example. Again, we choose $\beta_{elev} = 0.05$ as the particular alternative.

```
> data(gala, package="faraway")
> ## Refit previous Galapagos model to access objects
> lmod<- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+            data=gala)
> p0Omega<- 6; pomega<- 5 ## model sizes for this e.g. when testing a
>                      ## single df (r=1), i.e., testing omission of
>                      ## single predictor, e.g., elevation
> (r <- p0Omega - pomega) ## single df
[1] 1

> Cmat<- matrix(c(0,0,1,0,0,0), nrow=r) ## picks off b_elevation
> b0<- rep(0,p0Omega) ## only elev entry matters upon C%*%b0 in this e.g
> d0<- Cmat%*%b0 ## hyp mean under null
> b1<- b0; b1[3]<- 0.05 ## only elev entry matters upon C%*%b1 in this e.g
> d1<- Cmat%*%b1 ## specific hyp mean under alternative
> sig<- 60 ## error std dev (sqrt sigma^2) from educated guess, from
>           ## similar studies (may want to inflate to be conservative,
>           ## i.e., get larger sample size or smaller power or smaller
>           ## effects...)
> X<- model.matrix(lmod)
> XtX<- crossprod(X)
> ## Non-centrality parameter for R's F distribution
> (ncpF<- t(d0-d1)%*%solve(Cmat%*%solve(XtX)%*%t(Cmat))%*%(d0-d1) / sig^2)
[1,]
[1,] 0.8966

> ncpt<- sqrt(ncpF) ## Non-centrality parameter for R's t
>                      ## distribution. Be sure to negate if alternative d1
>                      ## < d0 of null (or otherwise make sure you know
>                      ## what's going on) (not here)
> (n<- dim(X)[1])
```

```
[1] 30

> ## Reject for  $F \geq F_{crit}$  determined by  $Pr(Type\ I) = alpha$ 
> alpha<-0.05
> (Fcrit<- qf(p=alpha, df1=r, df2=n-p0mega, lower=FALSE))

[1] 4.2597

> ## Equivalently, for 2-sided test, reject for small/large t (note alpha/2)
> (tcrit<- qt(p=alpha/2, df=n-p0mega, lower=FALSE))

[1] 2.0639

> round(Fcrit, 10) == round(tcrit^2, 10)

[1] TRUE

> ## Power  $Pr(reject H_0 | lam) = Pr(F \geq F_{crit} | lam)$ , same as in
> ## a previous example:
> (pwrF<- pf(q=Fcrit, df1=r, df2=n-p0mega, ncp=ncpF, lower=FALSE))

[1] 0.14878

> ## Power  $Pr(reject H_0 | lam) = Pr(|t| \geq tcrit | lam) (= Pr(t^2 \geq
> ## tcrit^2 | lam) = Pr(F \geq F_{crit} | lam), right?!)$ , same as pwrF in
> ## 2-sided case
> (pwrt <- pt(q=tcrit, df=n-p0mega, ncp=ncpt, lower=FALSE) +
+     pt(q=-tcrit, df=n-p0mega, ncp=ncpt, lower=TRUE)) ## little pwr

[1] 0.14878

> ## gain in lower
> ## tail for this
> ## e.g
>
> ## If, however, we perform a 1-sided test,  $H_0:b_elev = 0 \ v H_1:b_elev > 0$ 
> ## 0, then alpha buys all probability in the upper tail, making the
> ## upper tail rejection region larger for a 1-sided test giving more
> ## power for  $H_1:b_elev > 0$ .
>
> ## RR for 1-sided test, reject for only large t (note NOT alpha/2)
> (tcrit<- qt(p=alpha, df=n-p0mega, lower=FALSE))

[1] 1.7109
```

```

> (pwrt <- pt(q=tcrit, df=n-p0mega, ncp=ncpt, lower=FALSE))
[1] 0.23436

> curve(dt(x, df=n-p0mega), from=-4, to=4, ylab="density",
+         xlab="t(n-p) (central and non-central)")
> (tcrit2side<- qt(p=alpha/2, df=n-p0mega, lower=FALSE))

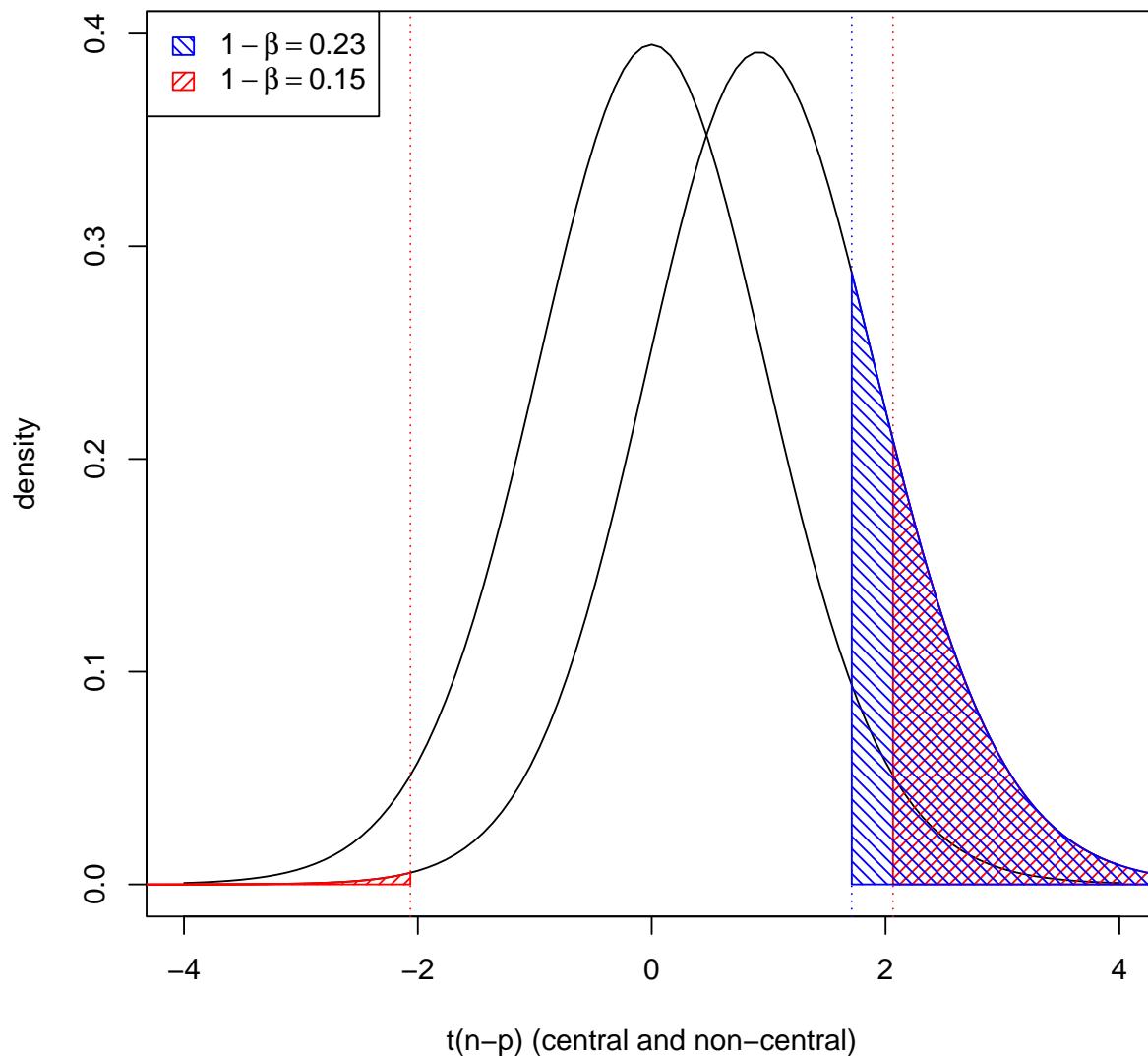
[1] 2.0639

> abline(v=c(-1,1)*tcrit2side, col="red", lty=3)
>
> curve(dt(x, df=n-p0mega, ncp=ncpt), from=-4, to=4, add=TRUE)
> tseq<- seq(tcrit2side, 5, 0.05)
> pdftseq<- dt(tseq, df=n-p0mega, ncp=ncpt)
> polygon(x=c(head(tseq,1),tseq,tail(tseq,1)),
+           y=c(0,pdftseq,0),
+           col="red", density=20, angle=45)
>
> tseq<- -rev(tseq)
> pdftseq<- dt(tseq, df=n-p0mega, ncp=ncpt)
> polygon(x=c(head(tseq,1),tseq,tail(tseq,1)),
+           y=c(0,pdftseq,0),
+           col="red", density=20, angle=45)
>
> (tcrit1side<- qt(p=alpha, df=n-p0mega, lower=FALSE))

[1] 1.7109

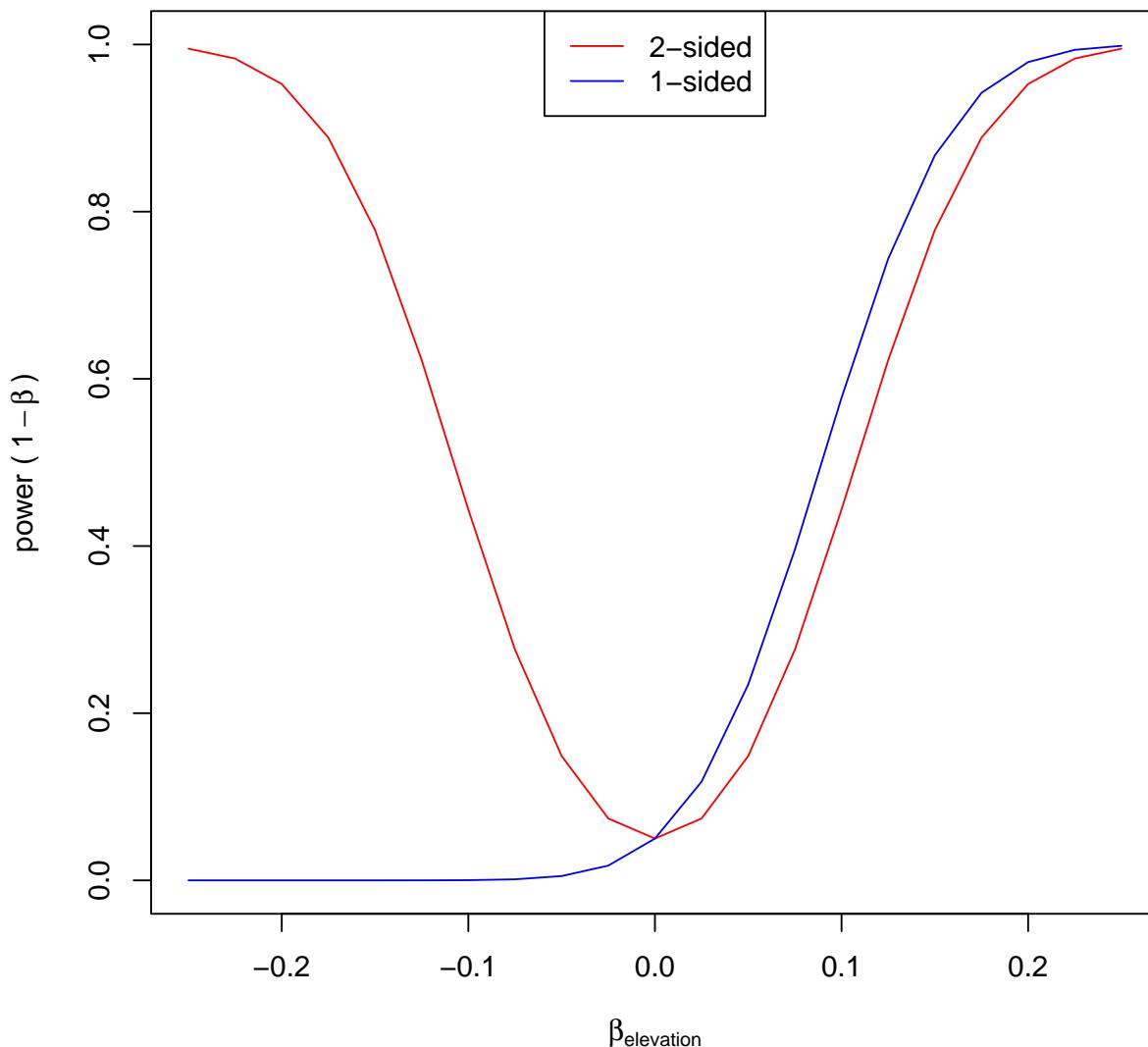
> abline(v=tcrit1side, col="blue", lty=3)
>
> tseq<- seq(tcrit1side, 5, 0.05)
> pdftseq<- dt(tseq, df=n-p0mega, ncp=ncpt)
> polygon(x= c(head(tseq,1),tseq,tail(tseq,1)),
+           y=c(0,pdftseq,0),
+           col="blue", density=20, angle=135)
>
> legend("topleft",
+         legend=c(expression(1-beta == 0.23),
+                  expression(1-beta == 0.15)),
+         density=c(20,20),fill=c("blue","red"),
+         angle=c(135,45), border=c("blue","red"))

```



Similar to the previous code, let's compare power vs. effect curves between a 2-sided test and a 1-sided test, i.e., not just for a single alternative, but for multiple alternative values of β_{elev} .

```
> d1<- NULL; lamt<- NULL; b1<- b0
> b1elevseq<- seq(-0.25, 0.25, 0.025)
> for(b1elev in b1elevseq) {
+   b1[3]<- b1elev ## only elev entry matters upon C%*%b1 in this e.g
+   d1<- Cmat%*%b1 ## specific hyp mean under alternative
+   lamt<- c(lamt, (d1-d0)%*%sqrt(solve(Cmat%*%solve(XtX)%*%t(Cmat))/ sig^2))
+ }
> pwr2side <- pt(q=tcrit2side, df=n-p0mega, ncp=lamt, lower=FALSE) +
+   pt(q=-tcrit2side, df=n-p0mega, ncp=lamt, lower=TRUE)
> pwr1side <- pt(q=tcrit1side, df=n-p0mega, ncp=lamt, lower=FALSE)
> plot(b1elevseq, pwr2side, type="l", col="red", ylim=c(0,1),
+       xlab=expression(beta[elevation]),
+       ylab=expression("power (^1-beta^)"))
> lines(b1elevseq, pwr1side, type="l", col="blue")
> legend("top", legend=c("2-sided","1-sided"),
+         lty=1, col=c("red","blue"))
```



3.3 Permutation Tests

- **Small and Abnormal?** When we cannot reasonably assume normality, and we do not have or are not sure what constitutes a “large” sample

size to invoke the Central Limit Theorem (CLT), then the permutation test offers an **alternative to testing that does not require the assumption of normality** (or large sample size).

- **Randomize to Break Relationship.** It is easy to imagine, **if response values occurred randomly without regard to the predictors**, then there is no relationship between the response and the predictors, and we would tend to see this lack of relationship in our particular observed responses and predictors. We would not expect to see estimated model coefficients significantly different from zero, except by chance, of course.
- **Permutation Test.** This is the idea behind the permutation test: **randomly permute** the observed values and, for each permutation, **compute a reasonable test statistic**, like a t or F , to measure association between the permuted values and the predictors, then compare the statistic for the actual, observed permutation to the **permutation distribution** of statistics by **computing a p-value** to help us decide if observed data is consistent with the random permutation distribution or not.
- **Mechanically Similar to Randomization Test.** We will see the permutation test in [Far14, Chap. 5] again, in the context of **experimental design**, where some people may call it a **randomization test** for reasons to be discussed when we get there.
- **Too Many Permutations?** As we will discuss then, for all but relatively small sample size, n , it is computationally infeasible to enumerate all possible permutations of the response and to compute all possible values of our test statistic to get the actual permutation distribution. (For large samples, we may be able to rely on the CLT.)
- **Sample Permutations.** Instead, we randomly sample from the permutation distribution to get a **Monte Carlo (MC)** approximate permutation distribution and MC approximate p-value.

- **Same Names, Different Distribution.** We choose to use the F or t statistics as our measures of association, getting them from the summary of the resulting `lm` object for convenience (we could compute “by hand,” right?!), but we will not compare them to the F/t distribution based on normal theory discussed above but will, of course, compare to their (MC approximate) permutation distributions under the null hypothesis. (You may choose another reasonable measure of association, but you may have to justify its use, and you may not be able to easily compare its permutation results to other results, e.g., normal theory results.)
- **Null Hypothesis.** If we permute the response, this implies we are interested in the null hypothesis of no association between response and any of the predictors, analogous to the overall F test, above. If we permute a single predictor, then the null is that the predictor is not associated with the response, similar to single df t or F tests done previously. If we permute each predictor in a group of predictors...you get the idea, right?!

3.3.1 Example: Permutation Test of Overall Association

```
> ## We will use the F statistic (but not the F distribution).
> ## Let lm/summary compute F for us. We just need to know where to get it.
> ## For example:
> lmod <- lm(Species ~ Nearest + Scruz, gala)
> lms <- summary(lmod)
> lms$fstatistic

  value    numdf    dendf
0.60196  2.00000 27.00000

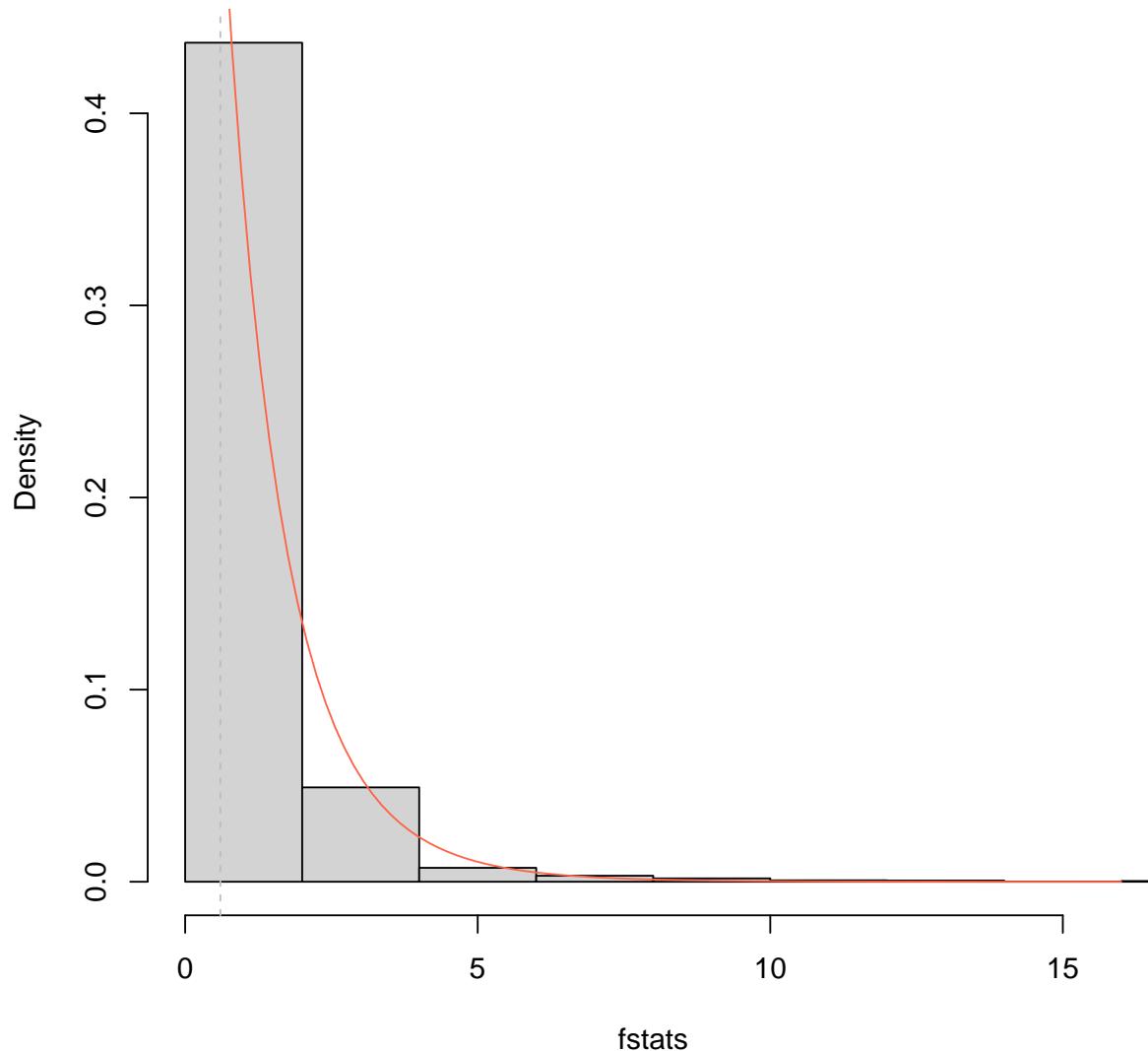
> ## (Okay, we're curious what normal theory F says about the p-value):
> 1-pf(lms$fstatistic[1],lms$fstatistic[2],lms$fstatistic[3])

  value
0.55493
```

- Now, randomly resample the response values (without replacement to get permutations of the observed values) (again, too many permutations to enumerate),
- compute (get) your test statistic for each sample to form a sample of of test statistics from your statistic's permutation distribution (not normal theory F or t distribution),
- then compare the actual observed test statistic to its permutation distribution with a p-value.

```
> nreps <- 4000
> set.seed(123)
> fstats <- numeric(nreps)
> for(i in 1:nreps){
+   lmods <- lm(sample(Species) ~ Nearest+Scruz, gala)
+   fstats[i] <- summary(lmods)$fstat[1]
+ }
> ## Overall test of association, analogous to our overall F test,
> ## via (MC approximate) permutation distribution p-value
> mean(fstats > lms$fstat[1]) ## similar to normal theory F dist
[1] 0.55825
```

```
> ## Histogram of MC approximate permutation distribution:
> hist(fstats, prob=TRUE, xlim=c(0,16))
> ## Compare to normal theory F distribution (just curious):
> curve(df(x,lms$fstat[2],lms$fstat[3]), add=TRUE,
+        col="tomato")
> ## Our observed test statistic:
> abline(v=lms$fstat[1], lty=2, col="grey")
```

Histogram of fstats

3.3.2 Example: Permutation Test of Single Predictor

- The above test was analogous to the overall F test. As mentioned, we can also test **other hypotheses**.

- Let's test for association with a **single predictor** by randomly sampling the value of that predictor (without replacement) and getting a (MC approximate) permutation distribution of a reasonable measure of association, like the t statistic (1-sided or 2-sided alternative) or F statistic (2-sided alternative).
- We use the resulting (MC approximate) permutation distribution of the t statistic to compute a (two-sided) p-value; we do not use the normal theory t distribution to compute the p-value.

```
> ## Again, let lm/summary compute t for us. Where to get it?
```

```
> summary(lmod)$coef[3,] ## or see lms object
```

Estimate	Std. Error	t value	Pr(> t)
-0.44064	0.40253	-1.09467	0.28333

```
> ## Now ,compute p-value from MC approximate permutation distribution
```

```
> tstats <- numeric(nreps)
> set.seed(123)
> for(i in 1:nreps){
+ lmods <- lm(Species ~ Nearest+sample(Scruz), gala)
+ tstats[i] <- summary(lmods)$coef[3,3]
+ }
```

```
> ## Test of association with single predictor, analogous to our t test,
> ## now via (MC approximate) permutation distribution p-value
> mean(abs(tstats) > abs(lms$coef[3,3]))
```

```
[1] 0.26825
```

```
> ## Compare to normal theory results
```

```
> summary(lmod)
```

Call:

```
lm(formula = Species ~ Nearest + Scruz, data = gala)
```

Residuals:

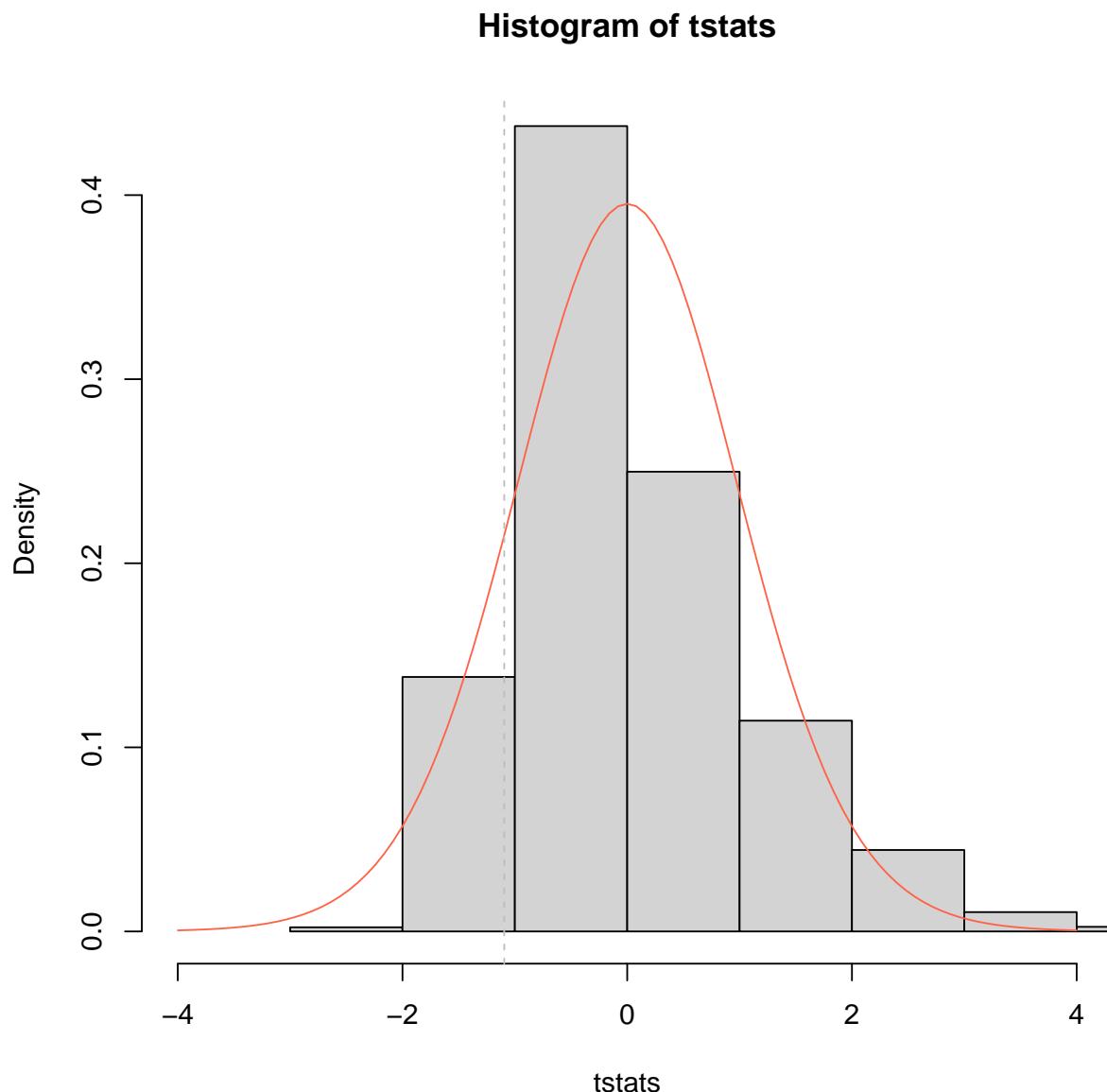
Min	1Q	Median	3Q	Max
-----	----	--------	----	-----

```
-97.9 -73.5 -46.3 18.3 344.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 98.477     28.356    3.47   0.0018 **  
Nearest      1.179      1.918    0.61   0.5439    
Scruz       -0.441      0.403   -1.09   0.2833    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 116 on 27 degrees of freedom
Multiple R-squared:  0.0427, Adjusted R-squared:  -0.0282 
F-statistic: 0.602 on 2 and 27 DF,  p-value: 0.555
```

```
> hist(tstats, prob=TRUE, xlim=c(-4,4))
> curve(dt(x,lms$fstat[3]), add=TRUE, col="tomato")
> abline(v=lms$coef[3,3], lty=2, col="grey")
```



3.3.3 Take-Home Remarks

- Normal theory (parametric) results are generally better at detecting departures from the null hypothesis than the permutation (nonparametric)

test, but we need to justify use of normal theory results by checking normality (and other model assumptions).

- If the normal theory results and permutation results more/less agree, use either result or report both.
- If results disagree substantially, and you cannot make a good case for the normality assumption, then use the permutation test results.

3.4 Sampling

We will discuss this section ([Far14, §3.4]) together with [Far14, Chap. 5], when we get there.

3.5 Confidence Intervals for β

We can use Results B.10 and B.11 to obtain confidence intervals for individual β_j or, more generally, some linear combination of β . In particular, Result B.10 gives us the following. (A picture of a Student's (standard) t distribution may be helpful if you want to follow the algebra.)

$$\begin{aligned}
(1 - \alpha) &= Pr \left(\left| \frac{\mathbf{C}\hat{\beta} - \mathbf{C}\beta}{\sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T}} \right| \leq t \left(1 - \frac{\alpha}{2}, n - p \right) \right) \\
&= Pr \left(t \left(\frac{\alpha}{2}, n - p \right) \leq \frac{\mathbf{C}\hat{\beta} - \mathbf{C}\beta}{\sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T}} \leq t \left(1 - \frac{\alpha}{2}, n - p \right) \right) \\
&= Pr \left(t \left(\frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \leq \mathbf{C}\hat{\beta} - \mathbf{C}\beta \leq \right. \\
&\quad \left. t \left(1 - \frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \right) \\
&= Pr \left(t \left(\frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} - \mathbf{C}\hat{\beta} \leq -\mathbf{C}\beta \leq \right. \\
&\quad \left. t \left(1 - \frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} - \mathbf{C}\hat{\beta} \right) \\
&= Pr \left(\mathbf{C}\hat{\beta} - t \left(1 - \frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \leq \mathbf{C}\beta \leq \right. \\
&\quad \left. \mathbf{C}\hat{\beta} - t \left(\frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \right) \\
&= Pr \left(\mathbf{C}\hat{\beta} - t \left(1 - \frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \leq \mathbf{C}\beta \leq \right. \\
&\quad \left. \mathbf{C}\hat{\beta} + t \left(1 - \frac{\alpha}{2}, n - p \right) \sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T} \right)
\end{aligned}$$

- **Probability or Confidence?** Note that the randomness in the above messy probability statement comes from the data, which are in $\hat{\beta}$ and $\hat{\sigma}$. Thus, before we observe our data, $\hat{\beta}$ and $\hat{\sigma}$ are random variables, and, thus, so are the bounds of the interval; thus, it makes sense to talk about the **probability**, $(1 - \alpha)$, of the (random) intervals containing the quantity of interest, $\mathbf{C}\hat{\beta}$. But, once we compute the bounds based on observed data values, then no randomness remains; thus, it no longer

makes sense to talk of probability. Either the computed interval contains the target, $\mathbf{C}\hat{\boldsymbol{\beta}}$, or it does not, and, in practice, we typically do not know which, of course. But, the derivation of the interval, above, shows how $(1 - \alpha)$, while technically no longer a probability, is somehow a **confidence level** about the interval containing $\mathbf{C}\hat{\boldsymbol{\beta}}$.

- **Confidence/Precision Trade-Off.** More in class.
- **One-Sided Intervals.** One-sided bounds follow in a similar fashion. Similar to one-sided tests vs. two-sided tests, a one-sided interval is somehow more precise than a two-sided interval (all else, including confidence level, equal). So, compute a one-sided interval if it makes sense. More in class.
- **Easy.** If the above algebra seems difficult, the following results are relatively easily implemented.

Result 3.1 (Interval for General Scalar $\mathbf{C}\beta$). *From Result B.10, we can obtain intervals for $\mathbf{C}\beta$,*

$$\begin{aligned} \mathbf{C}\hat{\boldsymbol{\beta}} &\pm t(1 - \alpha/2, n - p) \times \widehat{se}(\mathbf{C}\hat{\boldsymbol{\beta}}) \quad \text{or,} \\ \mathbf{C}\hat{\boldsymbol{\beta}} - t(1 - \alpha, n - p) \times \widehat{se}(\mathbf{C}\hat{\boldsymbol{\beta}}), & \quad \text{lower bound, or,} \\ \mathbf{C}\hat{\boldsymbol{\beta}} + t(1 - \alpha, n - p) \times \widehat{se}(\mathbf{C}\hat{\boldsymbol{\beta}}), & \quad \text{upper bound.} \end{aligned}$$

Result 3.2 (Interval for the Mean $E(Y | \mathbf{x}) = \mathbf{x}^T \boldsymbol{\beta}$).

- An important special case of the above Result 3.1 is when $\mathbf{C} = \mathbf{x}_0^T$, some chosen vector of covariates (including 1 if the intercept is in the model), not necessarily among the observed covariates \mathbf{x}_i , $i = 1, \dots, n$.
- That is, in this special case, we are inferring about $E(Y | \mathbf{x}_0) = \mathbf{x}_0^T \boldsymbol{\beta}$, i.e., about the mean of $Y | \mathbf{x}_0$ at some chosen covariate value \mathbf{x} .

- Your author postpones such intervals to [Far14, Chap. 4].

Result 3.3 (Interval for β_j).

Specializing the above Result 3.1 even further to a single parameter, we have intervals for β_j

$$\begin{aligned}\hat{\beta}_j &\pm t(1 - \alpha/2, n - p) \times \widehat{se}(\hat{\beta}_j) \quad \text{or,} \\ \hat{\beta}_j &- t(1 - \alpha, n - p) \times \widehat{se}(\hat{\beta}_j), \quad \text{lower bound, or,} \\ \hat{\beta}_j &+ t(1 - \alpha, n - p) \times \widehat{se}(\hat{\beta}_j), \quad \text{upper bound.}\end{aligned}$$

```
> ## 2-sided 95% CIs for bArea and bAdjacent
> ## ``By hand (with assistance from lm)'' using Results 3.13/3.15 (What's C?)
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, gala)
> bhat <- coef(lmod)
> sehatbhat <- sqrt(diag(vcov(lmod)))
> tcrit <- qt(0.975, 30-6)
> bhat[2] + c(-1,1) * tcrit * sehatbhat[2]
[1] -0.070216  0.022339

> bhat[6] + c(-1,1) * tcrit * sehatbhat[6]
[1] -0.111336 -0.038273
```

```
> ## Or, more automatically.
> confint(lmod)

              2.5 %    97.5 %
(Intercept) -32.464101 46.600542
Area        -0.070216  0.022339
Elevation    0.208710  0.430219
Nearest      -2.166486  2.184774
Scruz        -0.685093  0.204044
Adjacent     -0.111336 -0.038273
```

```
> ## 95% CI for mean  $E(Y | x_0=\text{median})$  using predict function...
> (newx<- as.list(apply(gala[,3:7], 2, median)))

$Area
[1] 2.59

$Elevation
[1] 192

$Nearest
[1] 3.05

$Scruz
[1] 46.65

$Adjacent
[1] 2.59

> predict(lmod, newdata=newx, interval="confidence")

      fit     lwr     upr
1 56.957 28.524 85.39
```

```
> ## ...or use the gmodels package...
> (Cmat<- matrix(c(1,unlist(newx)), nrow=1))

 [,1] [,2] [,3] [,4] [,5] [,6]
[1,] 1 2.59 192 3.05 46.65 2.59

> gmodels::estimable(lmod, cm=Cmat, conf.int=0.95)

              Estimate Std. Error t value DF
(1 2.59 192 3.05 46.65 2.59) 56.957    13.777 4.1344 24
                                 Pr(>|t|) Lower.CI Upper.CI
(1 2.59 192 3.05 46.65 2.59) 0.00037501   28.524   85.39
```

```
> ## ...or, by hand
> x<- model.matrix(Species ~ Area + Elevation + Nearest + Scruz +
+                     Adjacent, gala)
> y<- gala$Species
> bhat<- solve(xtx<-crossprod(x), crossprod(x,y))
> (n<- length(y))
```

```
[1] 30

> (p0mega<- dim(x)[2])

[1] 6

> (mse<- sum((y - x%*%bhat)^2)/(n-p0mega))

[1] 3718

> Vbhat<- mse*solve(xtx)
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> Cbhat<- as.vector(Cmat%*%bhat)
> sehatCbhat<- as.vector(sqrt(Cmat%*%Vbhat%*%t(Cmat)))
> Cbhat + c(-1,1) * qt(1-0.05/2, n-p0mega) * sehatCbhat

[1] 28.524 85.390
```

- Above intervals are 95% intervals for scalar (single or individual or univariate) quantities (**individual intervals**).
- We can also get **simultaneous confidence regions** for two or more quantities, though visualization is difficult beyond three quantities.
- See [Far14, pp. 44-46] for discussion of test/interval correspondence in the individual CI/test and simultaneous region/test cases, though I think his conclusion of “no disagreement” between individual tests and simultaneous test is wrong in his example.

Correspondence Between Tests & Intervals

- **Contains, Don’t Reject.** If a $1 - \alpha$ confidence interval (region) for $C\beta$ contains the null hypothesized value in a test of $C\beta$ at level α , then we would not reject the null at level α .

- **Doesn't Contain, Reject.** If the interval does not contain the null hypothesized value (or the value is on the boundary), then we would reject the null at level α .
- **E.g.** We use the next chunk to illustrate with 2-sided t tests. Of course, a similar correspondence holds between 1-sided tests and 1-sided intervals.
- **Not Always Exact Correspondance.** This correspondence does not always hold, but **intervals** with unknown corresponding test methods are often presented **from a testing perspective**; see Bootstrap Confidence Intervals, below.

```
> ## Still using previous lmod object
> summary(lmod)

Call:
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
    data = gala)

Residuals:
    Min      1Q  Median      3Q     Max 
-111.68 -34.90   -7.86   33.46  182.58 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 7.06822   19.15420   0.37   0.7154    
Area        -0.02394   0.02242  -1.07   0.2963    
Elevation    0.31946   0.05366   5.95 0.00000038 ***
Nearest      0.00914   1.05414   0.01   0.9932    
Scruz        -0.24052   0.21540  -1.12   0.2752    
Adjacent     -0.07480   0.01770  -4.23   0.0003 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 61 on 24 degrees of freedom
Multiple R-squared:  0.766, Adjusted R-squared:  0.717 
F-statistic: 15.7 on 5 and 24 DF,  p-value: 6.84e-07
```

```
> confint(lmod)

              2.5 %    97.5 %
(Intercept) -32.464101 46.600542
Area         -0.070216  0.022339
Elevation     0.208710  0.430219
Nearest       -2.166486  2.184774
Scruz        -0.685093  0.204044
Adjacent      -0.111336 -0.038273
```

3.6 Bootstrap Confidence Intervals

The bootstrap method can be used as an alternative to the t and F procedures to get intervals and regions that do not depend on normality, which is especially relevant for sample sizes too small for the Central Limit Theorem (CLT).

Simulation

To help develop intuition for the bootstrap method, suppose we know the true model underlying our data. Then, we could

1. Generate ϵ from the known error distribution.
2. Compute \mathbf{y} from $\mathbf{y} = \mathbf{X}\beta + \epsilon$.
3. Compute $\hat{\beta}$.
4. Repeat above many times to get the (MC estimated) distribution of $\hat{\beta}$.
5. Use the resulting distribution to explore properties of $\hat{\beta}$, e.g., mean, standard errors, intervals, etc.

Bootstrap

The (parametric) bootstrap method emulates the way the data were generated in the above simulation procedure.

1. Generate bootstrap error vector $\epsilon^* = (\epsilon_1^*, \dots, \epsilon_n^*)^T$ by **sampling with replacement** from the observed residuals, $\hat{\epsilon}_1, \dots, \hat{\epsilon}_n$
2. Compute bootstrap observations \mathbf{y}^* from $\mathbf{y}^* = \mathbf{X}\hat{\beta} + \epsilon^*$.
3. Compute bootstrap LS estimator $\hat{\beta}^*$ using bootstrap observations \mathbf{y}^* and \mathbf{X} (no star).
4. Repeat above many times to get a bootstrap sample of $\hat{\beta}^*$, which we call the **bootstrap distribution** of β .
5. Form empirical intervals from the bootstrap sample of the $\hat{\beta}^*$ values.

Of course, we have not discussed the theory behind why such a distribution or intervals therefrom are appropriate for the unknown β . We skip it. You can find plenty of additional information on various bootstrap procedures from an online search.

```
> ## We're using the latest lm object lmod
> set.seed(123)
> nb <- 4000
> coefmat <- matrix(NA,nb,6) ## <-- to hold betastar vectors
> resids <- residuals(lmod) ## <-- residual vector epshat
> preds <- fitted(lmod) ## <-- fitted vector yhat
> for(i in 1:nb){
+   ## Step (1) w/replacement and step (2)
+   booty <- preds + sample(resids, rep=TRUE)
+   ## Step (3)
+   bmod <- update(lmod, booty ~ .)
+   coefmat[i,] <- coef(bmod)
+ }
> ## 95% empirical CIs for betas
> colnames(coefmat) <- c("Intercept",colnames(gala[,3:7]))
> coefmat <- data.frame(coefmat)
> (res<-apply(coefmat,2,function(x) quantile(x,c(0.025,0.975))))
```

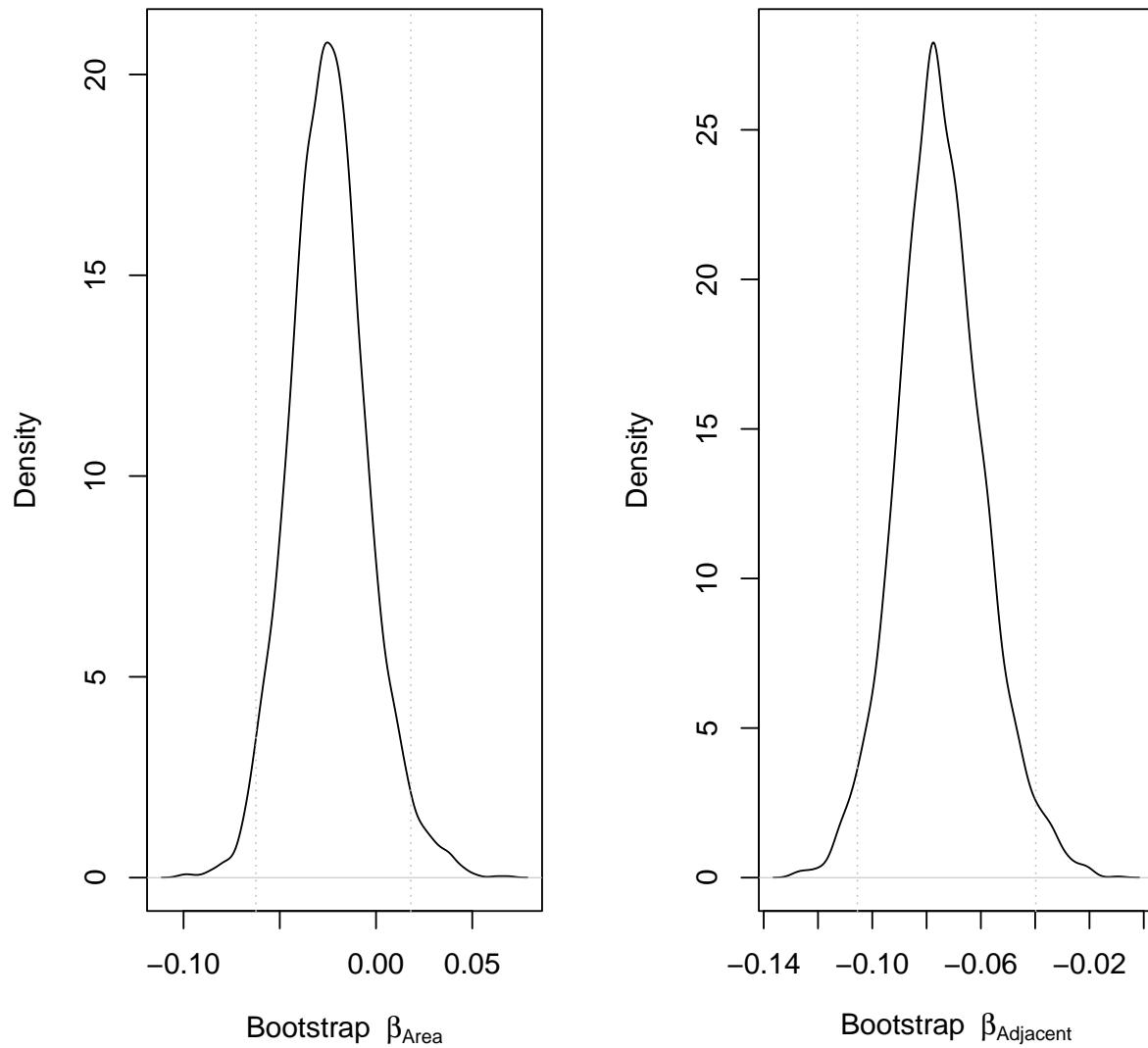
	Intercept	Area	Elevation	Nearest	Scruz	Adjacent
Intercept	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Area	-0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Elevation	-0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Nearest	-0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Scruz	-0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
Adjacent	-0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000

```
2.5%    -25.314 -0.062365   0.23110 -1.7166 -0.60620 -0.105453
97.5%     42.693  0.018074   0.42076  2.1227  0.16777 -0.039797

> ## Compare to normal assumption results ('cuz humans are naturally
> ## curious animals)
> t(confint(lmod))

(Intercept)      Area Elevation Nearest      Scruz Adjacent
2.5 %       -32.464 -0.070216   0.20871 -2.1665 -0.68509 -0.111336
97.5 %        46.601  0.022339   0.43022  2.1848  0.20404 -0.038273
```

```
> par(mfrow=c(1,2))
> plot(density(coefmat[,2]),
+       xlab=expression("Bootstrap " ~ beta[Area]),
+       main=NA)
> abline(v=res[,2], lty=3, col="grey")
> plot(density(coefmat[,6]),
+       xlab=expression("Bootstrap " ~ beta[Adjacent]),
+       main=NA)
> abline(v=res[,6], lty=3, col="grey")
```



```
> par(mfrow=c(1,1))
```

Lecture 4

Prediction

Contents

4.1	Confidence Intervals for Predictions	140
4.2	Example: Predicting Body Fat	143
4.3	Autoregression	148
4.4	What Can Go Wrong with Predictions?	156

4.1 Confidence Intervals for Predictions

- **Two Kinds of “Predictions”.** Your textbook’s author considers two kinds of “predictions.”
 1. **Prediction of a future output**, $Y | \mathbf{x}_0$, at some observed input vector, \mathbf{x}_0 , the zero subscript reminding us that this value does not have to be—often is not—among the original n observations.
 2. **“Prediction” of the mean of a future output**, $E(Y | \mathbf{x}_0)$.
- **Estimation.** Statisticians often prefer to distinguish the prediction of random and fixed quantities, and they often use the term **estimation** in the latter case (2). Picky!
- **And More.** We can also conduct simultaneous inference for unobserved values and their means as well as the prediction of the average of m values of $Y | \mathbf{x}_0$ ([KNNL05, 4.2, 4.3, 6.7]) (or its mean). See [RS13, pg 190 & Sec. 7.4.3] and [RS13, pg 283-4 & Sec. 10.4.2] for similar discussions of these procedures in the SLR and MLR cases, respectively. These can be subsumed into our $C\beta$ framework of note Chapter 3, but we will likely not cover these in more detail.

- **Example of Estimation (2).** In fact, we (not your textbook’s author) already illustrated the second case of “prediction,” above, in Result 3.2 in (note) Chapter 3, along with the (our) code therein; we *estimated* the mean of (a population of values represented by) $Y | \mathbf{x}_0$, i.e., we discussed the interval estimation for $E(Y | \mathbf{x}_0)$.
- **Prediction (1).** But, instead, as we said, we often want an interval for the random variable $Y | \mathbf{x}_0$ itself, which varies about its mean, $E(Y | \mathbf{x}_0)$.

- **Now a Moving Target.** Intuitively, given that $Y | \mathbf{x}_0$ is centered at $E(Y | \mathbf{x}_0)$, but with extra variability according to our model,

$$Y | \mathbf{x}_0 = E(Y | \mathbf{x}_0) + \epsilon,$$

the next result on **prediction intervals** should be somewhat intuitive.

- **Predictor.** In particular, our predictor is (not to be confused with the use of ‘predictor’ to mean input/covariate)

$$\text{Pred}(Y | \mathbf{x}_0) = \mathbf{x}_0^t \hat{\boldsymbol{\beta}}$$

which is the same as the **estimator** of $E(Y | \mathbf{x}_0)$ as mentioned in Result 3.2. (Note that we used different notation from your textbook’s author, who uses \hat{y}_0 to denote both the estimator of the mean and the predictor of an unobserved random output with that mean.) So, our point predictor for $Y | \mathbf{x}_0$ is the same as our point estimator for its mean, $E(Y | \mathbf{x}_0)$.

- **Prediction Error Variance.** But, as we said, we are aiming at a random target, $Y | \mathbf{x}$, with an estimator, $\mathbf{x}^t \hat{\boldsymbol{\beta}}$ (okay, predictor), which itself is random and independent of the target (Why independent?). Thus, we consider the **prediction error** to incorporate both sources of randomness, $\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x}$, and its **variance**

$$\begin{aligned} \text{Var}(\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x}) &= \text{Var}(\hat{\mu}(Y | \mathbf{x})) + \text{Var}(Y | \mathbf{x}) \\ &= \text{Var}(\mathbf{x}^T \hat{\boldsymbol{\beta}}) + \text{Var}(Y | \mathbf{x}) \\ &= \sigma^2 \mathbf{x}^T (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x} + \sigma^2 \\ &= \sigma^2 (1 + \mathbf{x}^T (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x}), \end{aligned}$$

dropping subscripts for convenience.

- **(Estimated) Standard Error of Prediction.** This leads to the estimated standard error of prediction

$$\widehat{\text{se}}(\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x}) = \sqrt{\widehat{\sigma}^2 (1 + \mathbf{x}^T (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x})},$$

where $\widehat{\sigma}^2$ is the MSE (note section 2.4; [Far14, p. 16])

- **Standardize to a t .** And, this leads to an associated t distributional result, similar to what has been given previously

$$\frac{(\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x}) - 0}{\widehat{\text{se}}(\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x})} \sim t(n - p).$$

We might think of

$$\text{"(estimator - target) / se(estimator)"},$$

where the estimator is the prediction error and the target is zero error. (I just write it this way to make it look familiar, like previous t statistics we've used.)

- **Prediction Interval.** And, finally this gives us our **prediction** interval

$$\text{Pred}(Y | \mathbf{x}) \pm t(1 - \alpha/2, n - p) \widehat{\text{se}}(\text{Pred}(Y | \mathbf{x}) - Y | \mathbf{x}),$$

in essentially the same manner as we obtained confidence intervals in our notes §3.5 by “unraveling” a t ratio to get

$$\text{"estimator} \pm \text{multiplier} \times \text{standard error"}$$
.

- Of course, we can get one- or two-sided prediction intervals.

- **Confidence or Prediction Interval?** For each of the following items, explain whether to use a **confidence** interval for a mean response, $E(Y | \mathbf{x})$, or a **prediction** interval for a new observation, $Y | \mathbf{x}$. It's not always easy to know which to use. (Adapted from ALSM Kutner et al.)

1. What will be the humidity in the greenhouse tomorrow if we set the temperature to 25° C?
2. How much do we expect families whose disposable income is \$45,000 to spend, on average, for meals away from home?

3. How many kilowatt hours of electricity will be consumed next month by residential customers in the Orange County service area, given that monthly average temperature will be 65° F?

4.2 Example: Predicting Body Fat

Here, we illustrate prediction of body fat for a “typical” individual, as determined by the median of observed input/covariate values, and compare it to estimation of the mean body fat of a population of typical individuals. The first may be used for an individualized prognosis. (NOTE: We’d actually use a particular individual’s characteristics, x_0 , to do this, but, here, we use the median observed input value for expediency.) In either case, we can use the **stats::predict** function, with different option values as illustrated below.

```
> ## Body fat example
> library(faraway)
> data(fat, package="faraway")
> lmod <- lm(brozek ~ age + weight + height + neck + chest +
+               abdom + hip + thigh + knee + ankle +
+               biceps + forearm + wrist, data=fat)
```

```
> ## Point estimate of the mean  $E(Y | x_0 = \text{median})$ 
```

```
> x <- model.matrix(lmod)
```

```
> (x0 <- apply(x, 2, median)) ##
```

	age	weight	height	neck
(Intercept)	43.00	176.50	70.00	38.00
chest	abdom	hip	thigh	knee
99.65	90.95	99.30	59.00	38.50
ankle	biceps	forearm	wrist	
22.80	32.05	28.70	18.30	

```
> (y0 <- sum(x0 * coef(lmod)))
```

```
[1] 17.493
```

```

> ## Interval estimate of the mean  $E(Y | x_0 = \text{median})$ 
> (est<- predict(lmod,new=data.frame(t(x0)),
+                   interval="confidence", se.fit=TRUE))

$fit
    fit     lwr     upr
1 17.493 16.944 18.042

$se.fit
[1] 0.27867

$df
[1] 238

$residual.scale
[1] 3.988

> ## Estimated standard error of estimation (not yet prediction se)
> ##  $\sqrt{\sigma_{\hat{Y}}^2 (x_0^T X^T X)^{-1} x_0}$ 
> est$se.fit

[1] 0.27867

> ## Width
> est$fit[, "upr"] - est$fit[, "lwr"]

[1] 1.0979

```

```

> ## Now, _prediction_ of  $Y | x = \text{median}$ 
> (pred<- predict(lmod,new=data.frame(t(x0)),
+                   interval="prediction", se.fit=TRUE))

$fit
    fit     lwr     upr
1 17.493 9.6178 25.369

$se.fit
[1] 0.27867

$df
[1] 238

```

```
$residual.scale  
[1] 3.988  
  
> ## Now, estimated standard error of _prediction_  
> ## sqrt{sigmahat^2 (1 + x0^T(X^TX)^{-1}x0)}  
> sqrt(pred$residual.scale^2 + pred$se.fit^2)  
  
[1] 3.9977  
  
> ## Width  
> pred$fit[, "upr"] - pred$fit[, "lwr"]  
  
[1] 15.751
```

• **Prediction Intervals Are Relatively Wide.** The above example's prediction interval for $Y | x_0$ is wide, practically speaking, with a range of nearly 16%, compared to the relatively precise width of about 1% for the estimation interval of the mean ($E(Y | x_0)$) of the population of individuals who share the same characteristics, x_0 . The prediction interval's lower bound vs. its upper bound are akin to a marathon runner vs. average informatics professor. Several different parallel existences can fit between these values (technically speaking), or, at least, a small child.

• **We Estimate Means Relatively Well.** The interval for the mean of (the population of values represented by) $Y | x_0$, i.e., interval for $E(Y | x_0)$ is much more precise, as we might expect when estimating a mean vs. a single (un)observation. This is especially true near the median or typical (close to average) predictor (covariate) value and when we have a largish residual standard error ($\approx \sigma$) and largish sample size, n .

• **Uncertainty Increases Away from the Middle of the Inputs.** In general, both estimation and prediction intervals become wider as we move away from the average of the observed predictors. We illustrate in the next code chunk.

- **Curse of Dimensionality.** In high dimensional data (biggish p), we suffer from the curse of dimensionality in the sense that prediction at most x_0 will tend to be extrapolations near the “edge” of the data and result in wide intervals, unless we have a very large sample size to go along with our big p ([HTF01, §2.5]; more about the curse in INF 504).
- **BTW, Multicollinearity.** (The silly result of a negative coefficient for the weight predictor may be due to multicollinearity. Correlated inputs (columns of \mathbf{X}) cause instability that results in increased variance for $\hat{\beta}$, hence for mean estimates and predictions, too. Also, we are ‘adjusting for other covariates/inputs,’ which usually influences the ‘effect’ of covariate, like weight, especially in observational (non-randomized) studies. See Chapter 5 Explanation for a discussion of ‘effect.’)

```
> ## Illustrate increasing uncertainty as we move away from middle input
> ## values (using weight as and example).
> (x0 <- apply(x, 2, median))

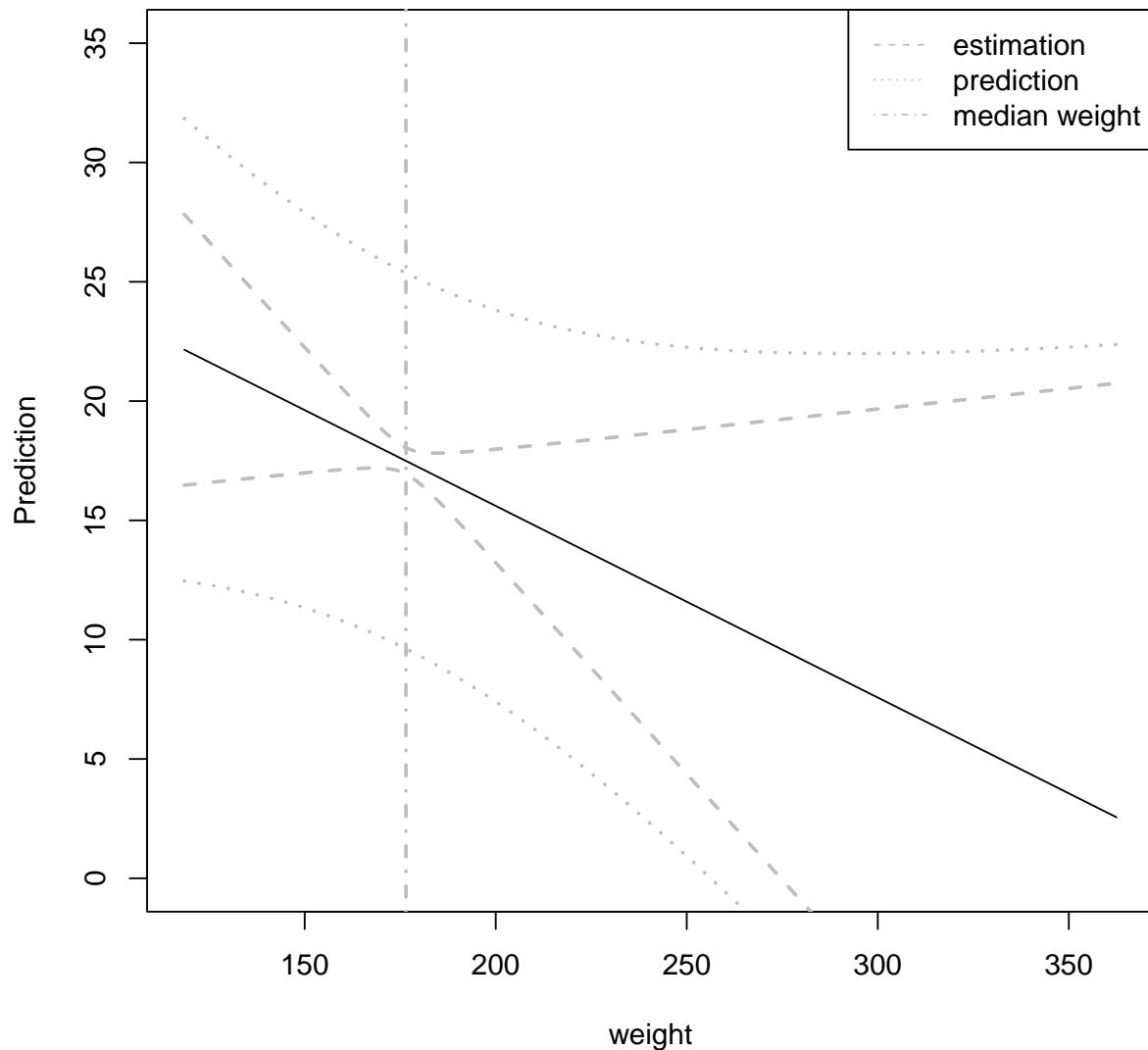
(Intercept)      age     weight    height    neck
      1.00      43.00    176.50     70.00   38.00
      chest     abdom      hip     thigh    knee
      99.65     90.95    99.30     59.00   38.50
      ankle    biceps    forearm    wrist
      22.80     32.05    28.70     18.30

> x0$weight<- seq(min(fat$weight), max(fat$weight), by=1)

Warning in x0$weight <- seq(min(fat$weight), max(fat$weight), by = 1): Coercing
LHS to a list

> est<- predict(lmod, new=data.frame(x0), interval="confidence")
> pred<- predict(lmod, new=data.frame(x0), interval="prediction")
> plot(est[, "fit"] ~ x0$weight, type="l", xlab="weight", ylab="Prediction",
+       ylim=c(0,35))
> lines(est[, "lwr"] ~ x0$weight, lty=2, lwd=2, col="grey")
> lines(est[, "upr"] ~ x0$weight, lty=2, lwd=2, col="grey")
> lines(pred[, "lwr"] ~ x0$weight, lty=3, lwd=2, col="grey")
> lines(pred[, "upr"] ~ x0$weight, lty=3, lwd=2, col="grey")
```

```
> abline(v=median(fat$weight), lty=4, lwd=2, col="grey")
> legend("topright", legend=c("estimation", "prediction", "median weight"),
+         lty=c(2,3,4), col="grey")
```



We skip the “by hand” computation routine for prediction intervals, as we have done before for confidence intervals, but you should be able to do it

using the prediction interval formula provided in the notes.

4.3 Autoregression

Prediction of a future response measured over time is called **forecasting**, which, evidently, causes your author to very briefly discuss in this chapter a traditional time series method known as **autoregression**, whose primary focus is to forecast unobserved, future responses in time.

- **Additive Error Model.** Our underlying conceptual model has been

$$Y = f(\mathbf{x}) + \epsilon,$$

with primary target f , either for explanatory purposes ([Far14, Chap. 5]), or for prediction (forecasting in the current case) (see also notes §1.4); we illustrate the latter, here.

- **Model Bias or Process Error.** If $\mathbf{x}^T \boldsymbol{\beta} \neq f(\mathbf{x})$, we have mean model bias, which is clumped into our error, i.e., our uncertainty about the form of f is clumped into the error. Sometimes, this model uncertainty goes by the term **process error**, suggesting that we are thinking of modeling a physical or mechanistic process, f , that relates the process's inputs and outputs.
- **Unmeasured Predictors.** Obviously, unmeasured predictors in $f(\mathbf{x})$ (but left out of its (linear) model) will tend to result in a model that is biased for $f(\mathbf{x})$.
- **Unmeasured Spatial, Temporal or Biological Scale Predictors.** Often, such unmeasured predictors vary over time, space or some other biological or physical scale. If it is reasonable to assume that such unmeasured predictors somehow vary smoothly over these scales, then it makes

sense that **process error** would also tend to manifest itself smoothly, which would be evident in the residuals; i.e., the model bias, which is clumped into the error, would tend to manifest itself as a smooth function over space, time or biological / physical scale.

- **Time Series or Longitudinal Data & Autocorrelation.** One very important example is when we measure observations over time. In this case, large positive/negative residuals will tend to be close to large positive/negative residuals; we say that the residuals indicate the same phenomenon in the errors, i.e., *positive temporal autocorrelation* in the errors (see also [Far14, §6.1.3]). (Negative autocorrelation is much less common.)
- **Enhanced Error Model.** We can account for this not with the unmeasured covariates, of course—we assume we don't have them—but with a variance model to account for the process error that is clumped into the error term; a popular class of temporal process models are **autoregressive models**, which indirectly induce models for correlated errors by regression on past values of the output, i.e., past outputs (responses) become inputs (predictors) at the current time.

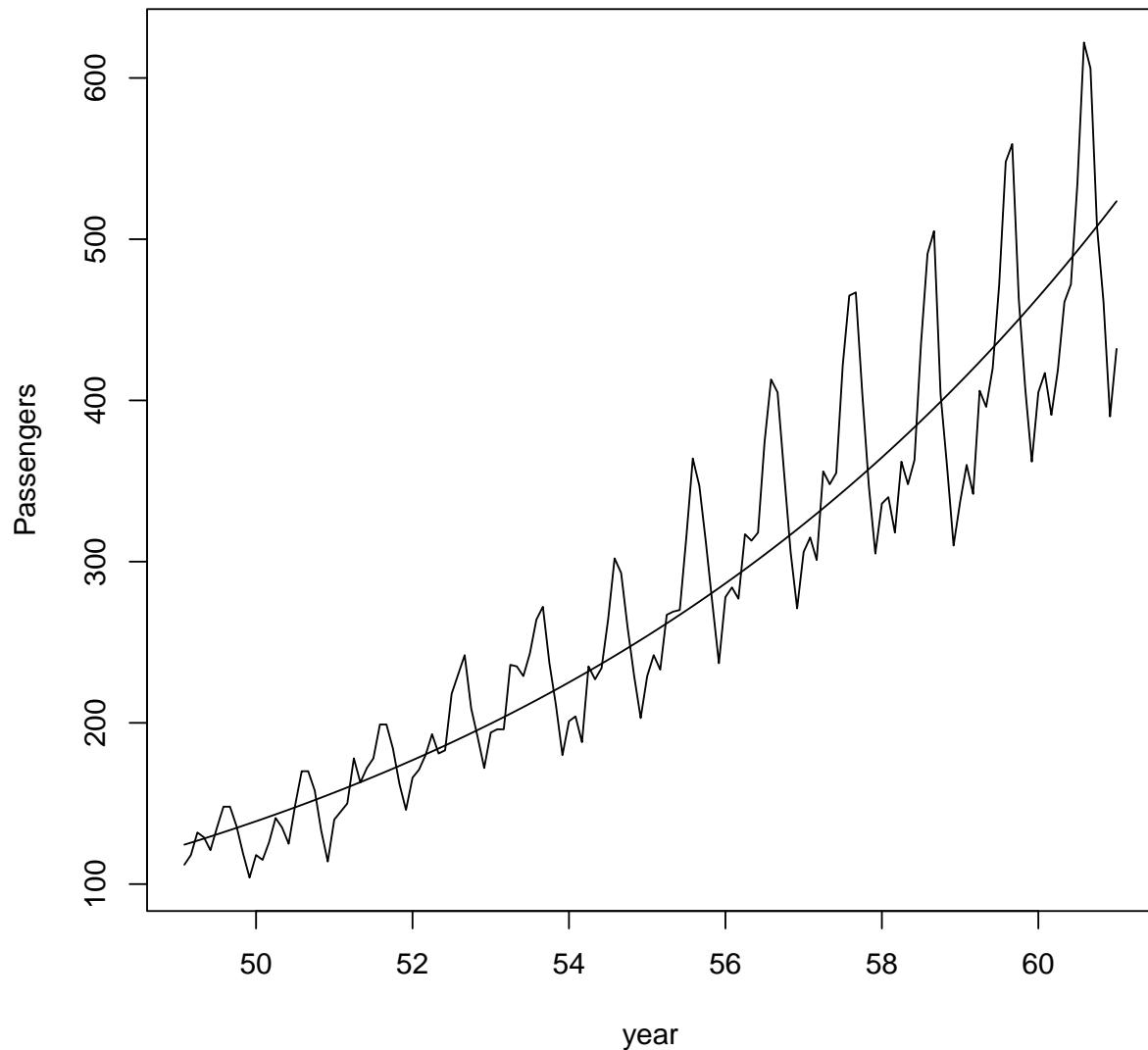
Example: Airline Passenger Data

- **Preliminaries: Remediate Non-Constant Variance.** Increasing error variance with mean response is typical; a traditional way to handle this **non-constant variance** is to transform the output ([Far14, §6.1.1]).
- **Log Transform.** We illustrate with the log transform of the output, which means that we assume that our standard deviation is approximately proportional to our mean (exactly proportional if we assume normality on the log scale as this gives a lognormal distribution of outputs on the log scale, and a log-normal distribution has a **constant coefficient of**

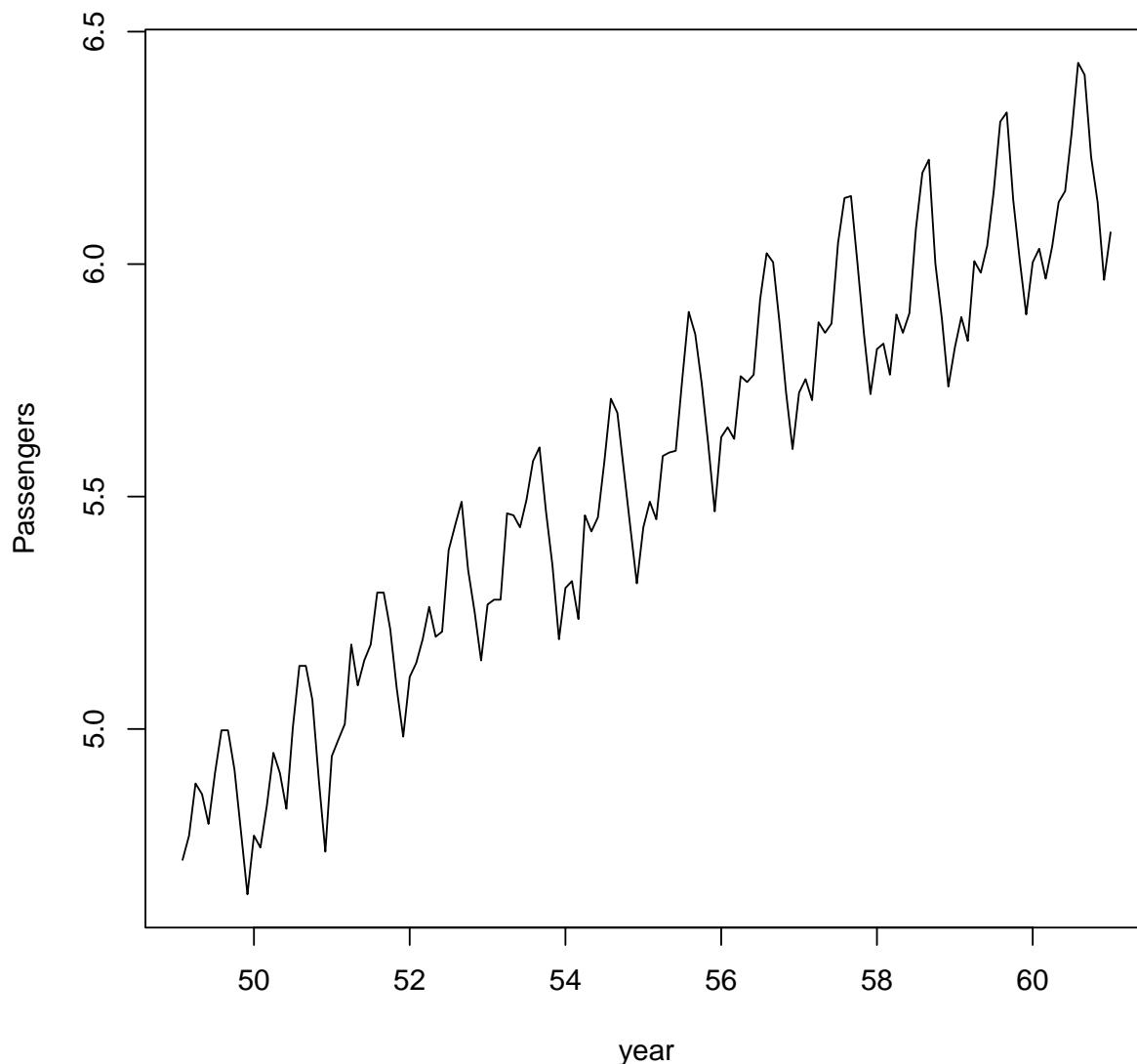
variation (ratio of standard deviation to mean); incidentally, a gamma distribution also has constant cv).

- **Later.** We may discuss alternative ways to handle (co-)variance modeling in [Far14, Chap. 8], and will certainly do so in INF 512.

```
> data(airpass, package="faraway")
> ## Non-constant variance:
> plot(pass ~ year, airpass, type="l", ylab="Passengers")
> ## Feeble, poor first attempt at a model:
> lmod <- lm(log(pass) ~ year, airpass)
> lines(exp(predict(lmod)) ~ year, airpass)
```



```
> ## Note how variance is stabilized after transformation:  
> plot(log(pass) ~ year, airpass, type="l", ylab="Passengers")
```



- **Model.** Your author argues for using **auto-covariates** in an auto-regression model: if we want to predict/forecast/extrapolate passengers for next month, it seems like we should use the current number of passengers (**current month**), and, to capture the obvious seasonal variation in the data, we should use last year's passengers for the month we want

to predict this year (**next month lagged by 12 months**) plus the month previous (**current month lagged by 12 months**), the latter by analogy to using the current month for prediction. Thus, his model for (log-thousands of) passengers at month (time) t is

$$y_t = \beta_0 + \beta_1 y_{t-1} + \beta_2 y_{t-12} + \beta_3 y_{t-13} + \epsilon_t.$$

```
> names(airpass); dim(airpass)

[1] "pass" "year"
[1] 144    2

> ## To get our auto-covariates, we use the embed function to take the
> ## 1D time series to 14D (obs. and 13 lags; see help(embed)).
> tmp<- embed(airpass$pass, dimension=14)
> colnames(tmp)<- c("y",paste0("lag",1:13))
> dim(tmp)

[1] 131   14

> tail(tmp) ## do you see how embed lags?

      y lag1 lag2 lag3 lag4 lag5 lag6 lag7 lag8 lag9 lag10 lag11
[126,] 622 535 472 461 419 391 417 405 362 407 463 559
[127,] 606 622 535 472 461 419 391 417 405 362 407 463
[128,] 508 606 622 535 472 461 419 391 417 405 362 407
[129,] 461 508 606 622 535 472 461 419 391 417 405 362
[130,] 390 461 508 606 622 535 472 461 419 391 417 405
[131,] 432 390 461 508 606 622 535 472 461 419 391 417
               lag12 lag13
[126,]    548   472
[127,]    559   548
[128,]    463   559
[129,]    407   463
[130,]    362   407
[131,]    405   362
```

```

> ## Fit and summarize (data now on log scale)
> lagdf <- embed(log(airpass$pass), 14)
> colnames(lagdf) <- c("y", paste0("lag", 1:13))
> lagdf <- data.frame(lagdf)
> armmod <- lm(y ~ lag1 + lag12 + lag13, data.frame(lagdf))
> summary(armmod)

Call:
lm(formula = y ~ lag1 + lag12 + lag13, data = data.frame(lagdf))

Residuals:
    Min      1Q  Median      3Q     Max 
-0.11117 -0.02485 -0.00244  0.02537  0.13402 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.1385     0.0536   2.58    0.011 *  
lag1         0.6923     0.0619  11.19   < 2e-16 *** 
lag12        0.9215     0.0347  26.53   < 2e-16 *** 
lag13       -0.6321     0.0677  -9.34   4.2e-16 *** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

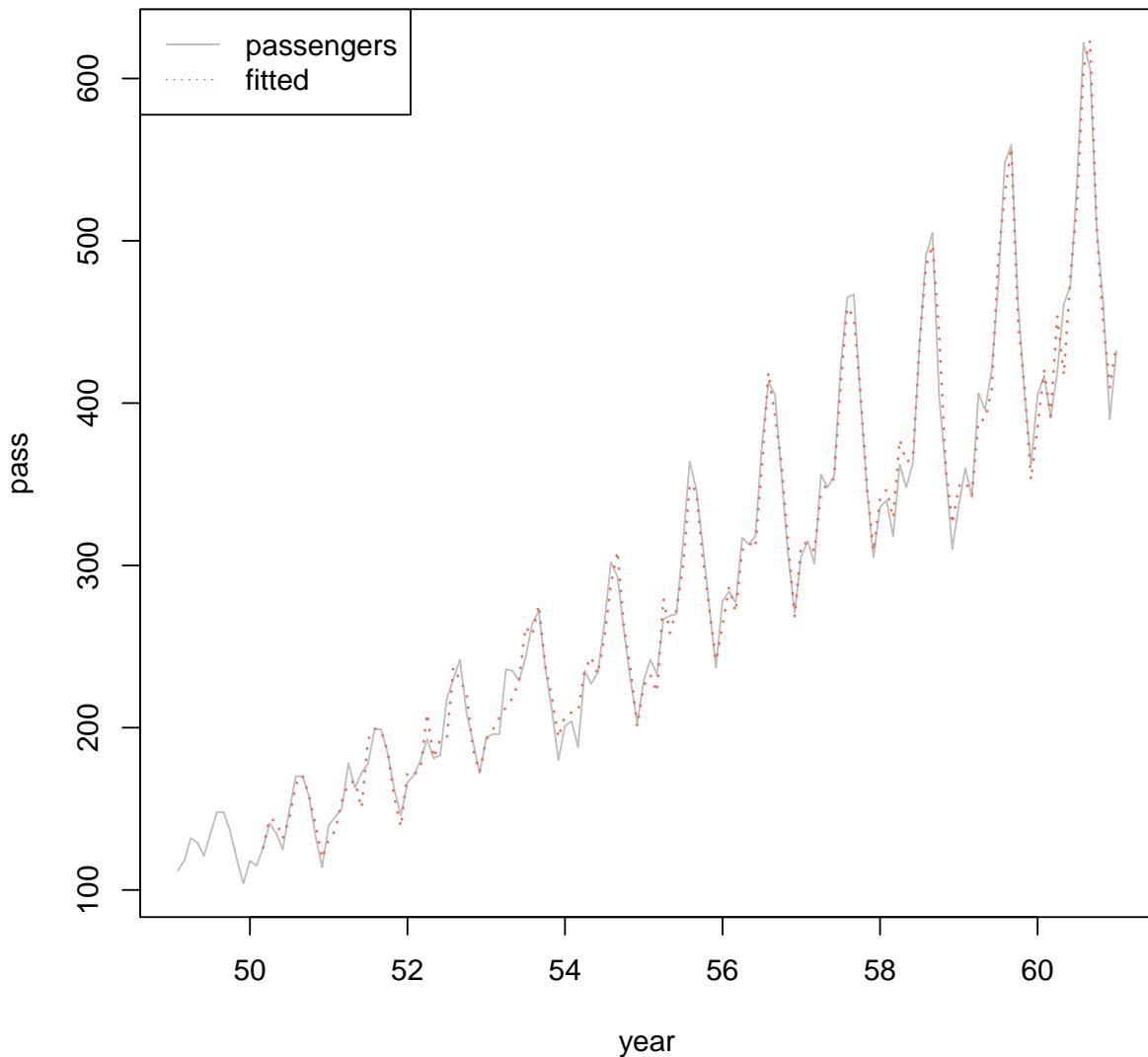
Residual standard error: 0.0416 on 127 degrees of freedom
Multiple R-squared:  0.989, Adjusted R-squared:  0.989 
F-statistic: 3.89e+03 on 3 and 127 DF,  p-value: <2e-16

```

```

> ## Observed and fitted
> plot(pass ~ year, airpass, type="l", col="grey")
> lines(airpass$year[14:144], exp(predict(armod)),
+        lty=3, col="tomato", lwd=1.5)
> legend("topleft", legend= c("passengers", "fitted"),
+        lty=c(1,3), col=c("gray", "tomato"))

```



```
> ## Next month's forecast (prediction) of passengers
> (lastobs<- tail(lagdf,n=1))

      y   lag1   lag2   lag3   lag4   lag5   lag6   lag7   lag8
131 6.0684 5.9661 6.1334 6.2305 6.4069 6.4329 6.2823 6.157 6.1334
      lag9   lag10  lag11  lag12  lag13
131 6.0379 5.9687 6.0331 6.0039 5.8916

> ## To predict yt, t=next month, we need to change labels so that this
```

```
> ## month's response y is covariate lag1, covariate lag1 is lag2, etc.  
> names(lastobs)<- paste0("lag",1:14)  
> (tmp<- predict(armod, newdata=lastobs, interval="prediction"))  
  
      fit     lwr     upr  
131 6.104 6.0206 6.1874  
  
> exp(tmp) ## transform interval back to raw number of passengers  
  
      fit     lwr     upr  
131 447.64 411.83 486.56
```

4.4 What Can Go Wrong with Predictions?

We give a brief summary of [Far14, §4.4].

1. **Bad Model.** We'll do model checking, later.
2. **Quantitative Extrapolation.** E.g., forecasting too far ahead or, more generally, predicting using covariates too far from observed covariate values.
3. **Qualitative Extrapolation.** Different populations: you developed the model under a set of circumstances but apply it to another set of circumstances.
4. **Overfitting.** Overconfidence. Intervals too narrow, p-values too small.
5. **Black Swans.** Your data suggest, "all swans are white" (i.e., suggest data follow normality, etc.), but one "black swan" will become a problem for your model's predictions; a small data set may not reveal observations that are inconsistent with your model (i.e., may not reveal black swans), or, in other words, your model won't predict black swans well because it was developed only using white swans (similar to Qualitative Extrapolation, I suppose).

Lecture 5

Explanation

Contents

Introduction	159
Conditional Mean Model vs. Marginal Mean Model	159
A Justification for Linear Modeling	160
Extrapolation, Meaningful Parameters & Reparameterization	162
Covariates Observed Without Error	164
5.1 Simple Meaning	164
Example: Galapagos Island Biogeography	165
Problems With Our Simple Meaning of Effect	169
5.2 Causality	170
5.3 Designed Experiments	173
Example: Motivation and Creativity	181
t Approximation, Randomization Distribution & Permutation Distribution	184
Formalizing Some Concepts	187
Restricted Randomization	189
Replicate Treatments	191
5.4 Observational Data	193
Voting Example	196
5.5 Matching	202
5.6 Covariate Adjustment	207
5.7 Qualitative Support for Causation	211
Sampling	211
Sampling Distribution: Example	214
Scope of Inference: Summary	223

Introduction

- **The More You Know.** I think we, as informatics graduate students, should know more about our models before we attempt to use them to explain what they're modeling—explanation being the topic of this chapter [Far14, Chap. 5]. I will discuss some aspects of our models, leaving other material in §AppendixB.6 & B.7 for the interested student. Still, I will be happy to answer questions. Please do ask questions.
- **Value Added.** I add a few things, below, in this Introduction, before moving on to material in [Far14, Chap. 5]

Conditional Mean Model vs. Marginal Mean Model

- **Model for Conditional Mean is Much Different for Marginal Mean.** As we've said before, we model the conditional mean, not the marginal mean. To illustrate a difference, consider these two simple models

$$\begin{aligned} E(Y) &= \beta_0 \quad \text{vs.} \\ E(Y | \mathbf{x}) &= \beta_0. \end{aligned}$$

(See the unnumbered section before §2.1 and see §B.7 with Definition B.16 of the regression function, a conditional mean. And, again, more generally, see §B.6 for joint, marginal and conditional distributions.)

- **Obvious.** The first model says only that the mean of the marginal distribution of Y is some constant, which does not seem to be a terribly daring or enlightening model! It's **obvious** that we expect a random variable (representing a population of values) to have a mean value. This is true whether the mean of $Y | \mathbf{x}$ varies with \mathbf{x} or not.

- **Not So Obvious.** Moreover, the first model says nothing about the second model, which, on the other hand, says that the mean of $Y | x$ does not vary with x , which seems to say quite a lot and is **not necessarily obvious** (though it does not say that the entire distribution of $Y | x$ does not vary with x). Typically, we focus effort on a model of the regression function, $E(Y | x)$ (hopefully with more interesting models than $E(Y | x) = \beta_0!$).

A Justification for Linear Modeling

- **1st Order, or Linear, Taylor.** A first order Taylor series approximation of the unknown regression function, $f(x) = E(Y | x)$ provides some justification for linear models, e.g.,

$$\begin{aligned} f(x) &\approx f(x_0) + \left(\frac{df}{dx} \Big|_{x_0} \right) (x - x_0) \\ &\equiv \beta_0 + \beta_1(x - x_0). \end{aligned}$$

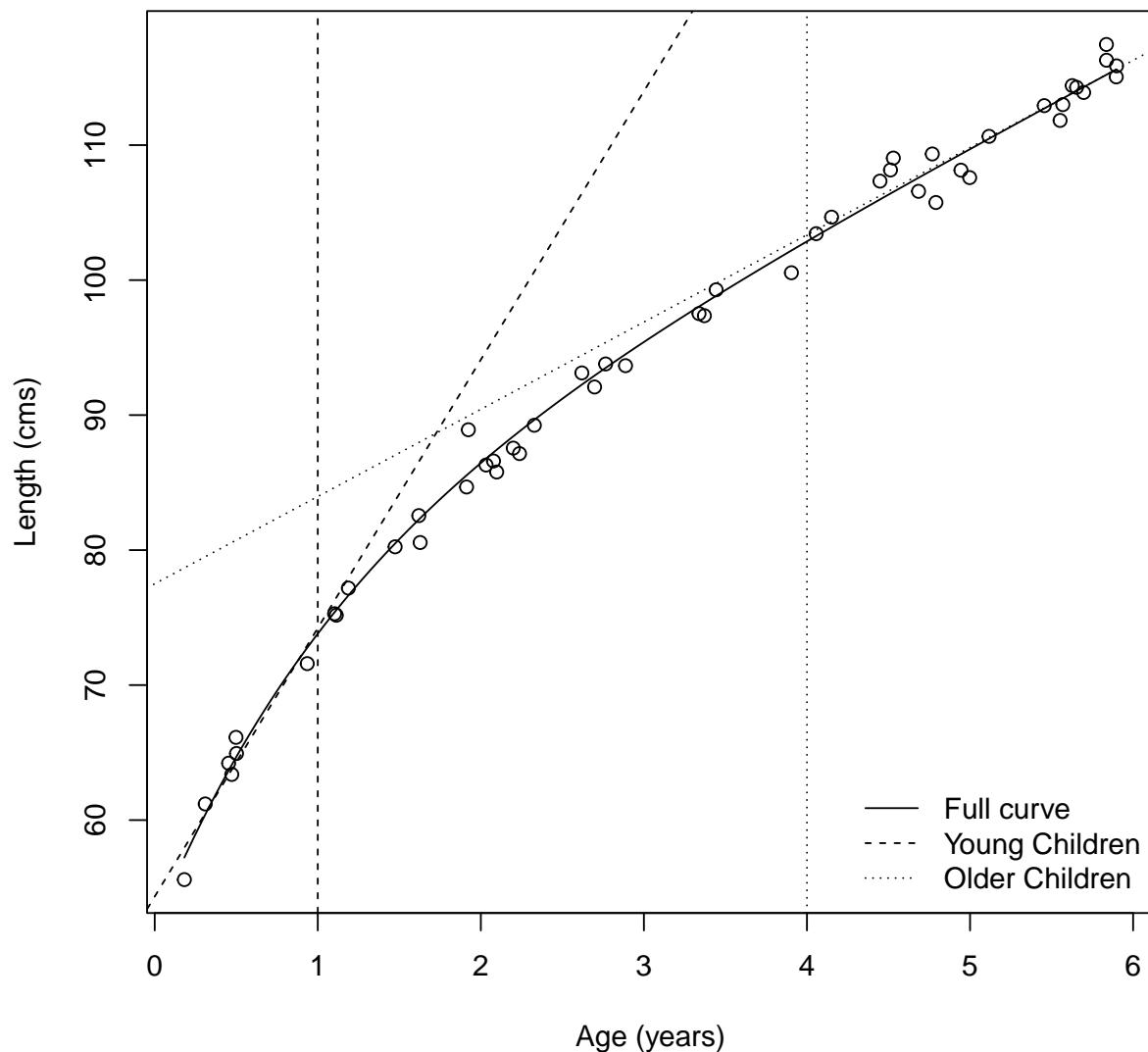
- **E.g., Jenss Growth Curve**

$$E(Y | x) = \beta_0 + \beta_1 x - \exp(\beta_2 + \beta_3 x),$$

illustrated in the nearby code/plot.

```
> ## Modified from Jon Wakefield's BFRM book/website:
> ## For true f (Jenss growth curve, non-linear in two ways):
> f<- function(x) {
+   beta0 <- 78; beta1 <- 6.4; beta2 <- 3.25; beta3 <- -.89
+   beta0 + beta1*x - exp(beta2+beta3*x)
+ }
> ## Data from true model:
> set.seed(20500 + 5150 + 24601)
```

```
> x <- runif(50,.1,6)
> y <- f(x) + rnorm(length(x),0,1.2)
> my.df<- cbind.data.frame(x=x,y=y)
> ## Plot data with true f
> plot(y ~ x, data=my.df, xlab="Age (years)",ylab="Length (cms)")
> curve(f, add=TRUE)
> ## Linear models (in x and beta) okay locally, perhaps
> abline(lm(y ~ x, data=my.df, subset=x>4)$coeff, lty=3, v=4)
> abline(lm(y ~ x, data=my.df, subset=x<1)$coeff, lty=2, v=1)
> legend("bottomright",bty="n",legend=c("Full curve","Young Children",
+ "Older Children"),lty=c(1,2,3))
```



Extrapolation, Meaningful Parameters & Reparameterization

- **Intercept Parameter.** Consider the intercept parameter, β_0 , in the

simple linear model (SLR)

$$E(Y | x) = \beta_0 + \beta_1 x;$$

depending on the data, β_0 **may not make much sense**.

- **Example.** For example, if y is **blood serum cholesterol** and x is weight, we may be more interested in the cholesterol of adults whose observed weights are likely clustered away from zero. So, our interest lies far from zero, not near $x = 0$, and, further, we do not have data near $x = 0$ to help inform the relationship between y and x in this case.
- **Extrapolation.** Generally, without a good theoretical model, **we do not want to infer beyond our data**, i.e., we do not want to use our model to **extrapolate**. (Remember, we may view a linear model as a first order approximation of a potentially more complicated function whose behavior is unknown beyond observed data.)
- **Example.** For the (fake) children's height vs age data, illustrated in a plot, above, the model makes a bit more sense if we observed data for children whose ages are close to zero. Presumably, height at age zero is the length of a child at birth, which seems to be a reasonably interesting quantity.

- **Reparameterization.** We may want to reparameterize our model to get parameters that are more meaningful.
- **A Common Reparameterization.** For example, a very common reparameterization of the above SLR is

$$E(Y | x) = \beta_0^* + \beta_1(x - x^*),$$

where x^* is often chosen as $x^* = \bar{x}$, the **average** of the x observations in the data, the median of x observations, or some other value that is

a convenient/meaningful reference/baseline level of the covariate. So, when a covariate, x , equals the reference/baseline level, x^* , β_0^* has a more familiar interpretation as mean at ‘typical’ characteristics, and we alleviate risk of extrapolating the intercept parameter.

- **Theory.** Theory may suggest a value for x^* where the unknown regression function is known to be approximately linear (as implied by the above discussion about first order Taylor approximation).
- **Other, Common Reference Levels.** Later, we will see reference levels of covariates in the case where a covariate is a categorical variable (i.e., factor), as in ANOVA, leading to relatively intuitive parameter interpretations; e.g., perhaps one level of a factor corresponds to a standard treatment or to a placebo, which some find to be a natural reference ([Far14, Chap. 14-17]).

Covariates Observed Without Error

Further, it is typical to assume that the process of observing the covariates does not introduce error, i.e., the **predictors are assumed to be observed without error!** I’m sure you can/will imagine someone, perhaps yourself, who has used data for which this assumption seems suspect. We may have more to say about this, later ([Far14, §7.1]).

5.1 Simple Meaning

We now begin to follow the numbered sections of [Far14, Chap. 5].

- **Additive Change Interpretation of β_j “Effects”.** The common, simple (simplistic?) mathematical interpretation of a linear regression function parameter, β_j , is the

additive change in the expected value of $Y | \mathbf{x}$ (i.e., in the (assumed linear model of the) regression function) for a one unit increase from x_j to $x_j + 1$, all other covariates $x_{j'}$ $j' \neq j$, held constant.

More generally, $c\beta_j$ is the additive change to the linear regression function with a c unit (additive) change in x .

- **Example.** Considering two covariates, x_1 and x_2 , in a multiple linear regression (MLR) model,

$$E(Y | x_1, x_2) = \beta_0 + \beta_1 x_1 + \beta_2 x_2,$$

we have the difference of means,

$$(\beta_0 + \beta_1(x_1 + c) + \beta_2 x_2) - (\beta_0 + \beta_1 x_1 + \beta_2 x_2) = c\beta_1.$$

- **Effect Jargon.** In short, β_j is called the “effect of covariate x_j ,” though we should be cautious about letting this suggest a cause and effect relationship between x_j and y ; more later. Also, this interpretation often depends on which other covariates are (not) in the model.
- **This Interpretation Has Its Problems.** The following example with the Galapagos biogeography data illustrates this interpretation, and brings to light some problems with the interpretation, which we discuss in a section following the example.

Example: Galapagos Island Biogeography

```
> ## MLR: How do we interpret beta_Elevation?
> library(faraway)
> data(gala, package="faraway")
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
+             data=gala)
> summary(lmod)
```

Call:

```
lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
  data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-111.68	-34.90	-7.86	33.46	182.58

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.06822	19.15420	0.37	0.7154
Area	-0.02394	0.02242	-1.07	0.2963
Elevation	0.31946	0.05366	5.95	0.0000038 ***
Nearest	0.00914	1.05414	0.01	0.9932
Scruz	-0.24052	0.21540	-1.12	0.2752
Adjacent	-0.07480	0.01770	-4.23	0.0003 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 61 on 24 degrees of freedom

Multiple R-squared: 0.766, Adjusted R-squared: 0.717

F-statistic: 15.7 on 5 and 24 DF, p-value: 6.84e-07

> ## SLR: Now how do we interpret beta_Elevation?

> **summary(lm(Species ~ Elevation, gala))**

Call:

```
lm(formula = Species ~ Elevation, data = gala)
```

Residuals:

Min	1Q	Median	3Q	Max
-218.32	-30.72	-14.69	4.63	259.18

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	11.3351	19.2053	0.59	0.56
Elevation	0.2008	0.0346	5.80	0.0000032 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 78.7 on 28 degrees of freedom

Multiple R-squared: 0.545, Adjusted R-squared: 0.529
F-statistic: 33.6 on 1 and 28 DF, p-value: 0.00000318

- **SLR Model Omits Correlated Covariates.** When we look at an island with elevation x_{elev} and compare it to another island with a different elevation, say $x_{elev} + 1$, the other island's (omitted) covariates are likely different from those of the first so that the change, β_{elev} , in the (modeled) species regression function likely does not simply reflect the change in elevation but also reflects changes in the other covariates that are related to elevation and to species but are not included in the SLR model.
- **MLR Model Adjusts For Other Covariates.** The MLR accounts for (i.e., 'adjusts for') the values of islands' other covariates (when not omitted!), and we see our interpretation of β_{elev} given above: the change in (the linear model of) $E(Y | \mathbf{x})$ with one unit change in x_{elev} **given that other covariates are held constant**. Omitting the other covariates in the SLR model of $E(Y | \mathbf{x})$ does not allow for this interpretation. (Orthogonal covariates aside, which do not seem plausible in this observational study.) Still, we might have omitted important covariates in the MLR model. In other words, without knowing "the true model," MLR offer *an* accounting or adjustment, not necessarily *the* accounting. More as we go.
- **Confounding, Fixing & Covariate Adjustment.** The **effect plot** (below, dashed line) illustrates our "additive change with others fixed" interpretation. Super(im)posing the plot of the SLR regression on elevation alone, we suspect that the SLR effect, β_{elev} , is **confounded** by the linear relationship of the output (Species) to the other covariates and by the other covariates' linear relationship to elevation; in other words, the MLR effect plot (for elevation) presents the elevation effect, β_{elev} , after **adjusting for other covariates**.
- **I.E.** In other words, our definition and effect plot illustrate the important idea that if (other) covariates are **fixed**, then they cannot effect the

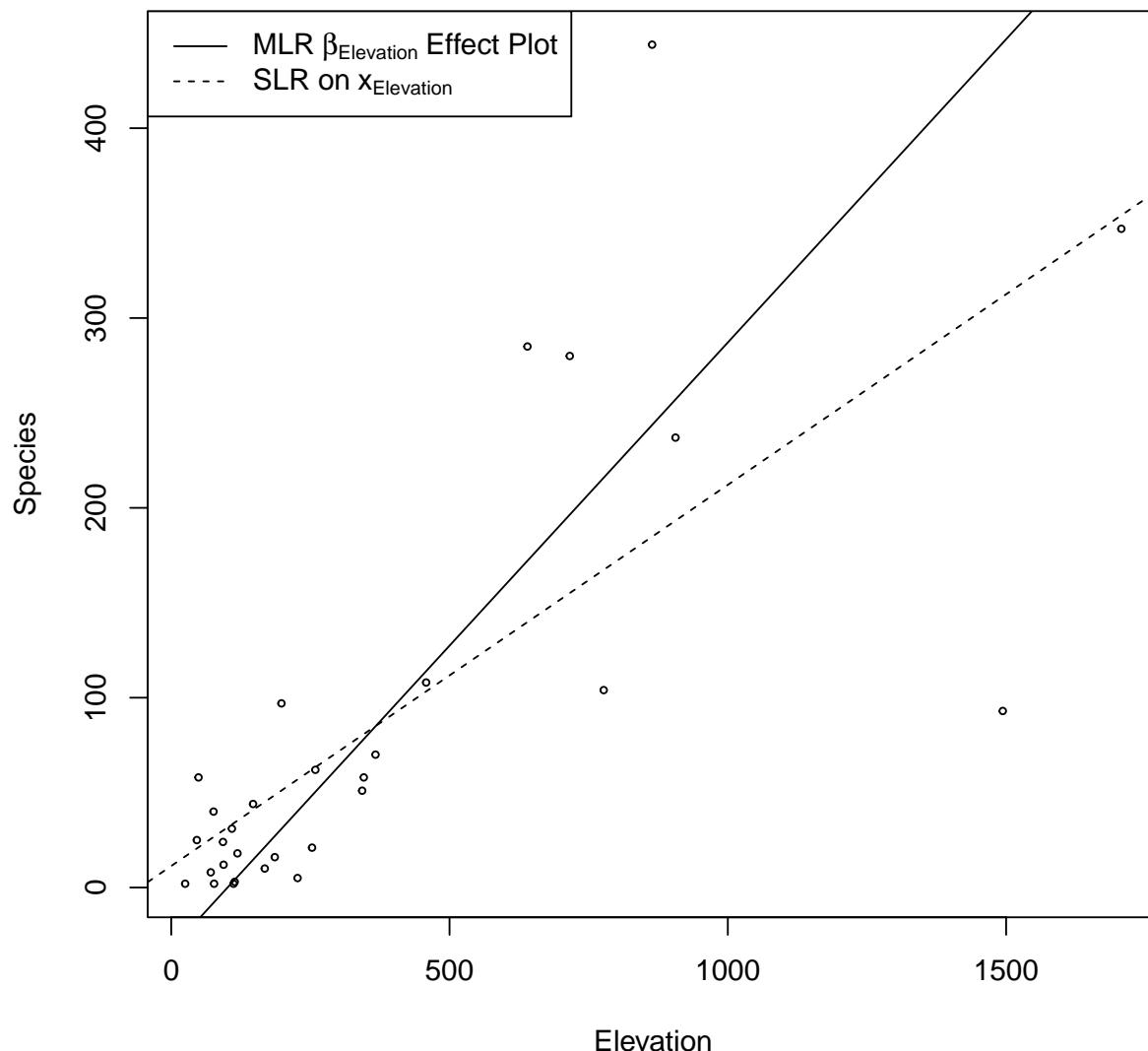
output, but if covariates are not fixed, as in the SLR model, then their covariability with the output and with elevation can **confound** the effect of elevation on output. Still, we may have omitted other important covariates.

- Much of this chapter is aimed at clarifying these **fundamental concepts** of confounding, fixing, and covariate adjustment (and mixing...).

```
> ## Figure 5.1 effect plot in Faraway
> plot(Species ~ Elevation, gala, cex=0.5)
> ## SLR:
> abline(lm(Species ~ Elevation, gala), lty=2)
> ## Now MLR:
> ## Fix other x (e.g., at average values)...
> fixotherx<- colMeans(gala)
> ## ...while varying xElevation.
> fixotherx$Elevation<- gala$Elevation

Warning in fixotherx$Elevation <- gala$Elevation: Coercing LHS to a list

> p <- predict(lmod, data.frame(fixotherx))
> i <- order(gala$Elevation)
> lines(gala$Elevation[i], p[i], lty=1)
> legend("topleft",
+        legend=c(expression("MLR" ~ beta[Elevation] ~ "Effect Plot"),
+                expression("SLR on" ~ x[Elevation])),
+        lty=c(1,2))
```



Problems With Our Simple Meaning of Effect

While our definition of effect gets at the important notion of confounding and suggests how we might address it (e.g., covariate adjustment), our definition is not without problems.

- **Out of Control.** Often, as with our example, covariates are not under our control, and we cannot reasonably hope to hold variables constant as others change
 - Can we change an island's elevation?
 - If we move from one island to another, both with the same elevation or different elevations, by chance, e.g., one with elevation x and the other with elevation $x + 1$, would the islands share the same values of other covariates?
 - Can we control a person's height, weight, gender?
 - What about covariates that we do not observe? Are they confounding the effect of the covariates that we do observe, as suggested by our example?
- **Interactions.** What about the case when a covariate, x , and its square, x^2 , are included in a model? What about interaction terms (e.g., $x_{i1}x_{i2}$)?
- **Causality.** Also, despite our use of the term **effect** there does not seem to be any notion of **cause and effect** in our Galapagos example. (However, we do begin to see such a notion hypothetically, as if we could hold fixed all islands' other covariates while changing only their elevations by the same amount to see how this would cause islands' outputs (Species) to change.)

5.2 Causality

- **Potential Outcomes & Counterfactuals.** In the case of a factor covariate with two levels (two treatments),

$$T = 0 \quad \text{or} \quad T = 1$$

(categorical covariate (input) with two possible values, “ $x = 0$ ” and “ $x = 1$ ”), let

$$y_i^0 \quad \text{and} \quad y_i^1$$

be the **potential outcomes** for subject / patient / unit i under the two different treatment conditions. Almost always, we can observe only one outcome, the outcome that we do not observe is called a **counterfactual**, because, well, considering it an observation is counter to the fact that we actually did not observed it.

- **Causal Effect.** We now define the causal effect of the treatment on subject i as the difference of potential outcomes,

$$\delta_i = y_i^1 - y_i^0.$$

And, because we observe only one of the potential outcomes, we cannot observe causal effects. Note this is defined for an individual (e.g., it's not an average over individuals).

- **Inherent Individual Response.** It may help to look at y_i^0 as the inherent individual response for an “untreated” individual, without the effect, δ_i , of the treatment $T = 1$, as if $T = 0$ is a sort of placebo or reference treatment,

$$y_i^1 = y_i^0 + \delta_i$$

- **No Going Back.** Why do we say that we can't we observe both potential outcomes? Once a patient / unit is subjected to a treatment condition, we take the stance that **we cannot somehow undo the treatment** condition's effect on the patient to then observe the unit's outcome under the other value of the treatment that was not originally assigned, as if the first treatment condition will have no effect on the outcome from the unit when the unit is subsequently subjected to the other treatment condition, as if the first treatment condition did not happen. Besides the effect of the first treatment condition on the patient, **other changes** to the patient may occur to the patient in the time between treatments; it's not the same patient/unit; hence, again, we take the stance that we

cannot observe the counterfactual; hence, we cannot actually obtain an individual's causal effect of treatment by differencing potential outcomes.

- **Pristine Causality.** In other words, the patient's/unit's attributes (aside from treatment conditions) would change between treatments. That is, these attributes, i.e., covariates, are not **fixed** between treatments, i.e., they are not **balanced** between treatments. Thus, the difference in outcomes between treatments may include the effects of these other, **confounding** covariates, thus masking or biasing the causal effect of treatment. Thus, we have established a very reasonable if pristine and **unattainable definition of cause and effect!** (Note how the idea of adjusting (fixing) other covariates, but not unobserved covariates, has begun to creep in here as in the previous discussion of the effect of elevation on the number of species in the Galapagos observational data...)
- **Clear Reference Point, However.** At least we now have a clear (if unattainable) goal to pursue.
- **How Close Can We Get?** Let's see how close we can get to our ideal notion of causality. In the process, we hope to clarify "why?" we may infer causality. This is a **fundamental** question.
- **Basic, Take-Home Idea.** In short, the basic idea is to somehow **fix or balance units' confounding covariates within and across treatments when comparing outcomes across treatments** with the goal of isolating or unmasking treatment effects, unconfounded by effects of other covariates; notice that our notion of fixing covariates in our simple definition of effect, illustrated in the effect plot, in §5.1, above, is akin to this basic idea. If we cannot **fix** values between treatment conditions (if they're not observed), then we try to achieve balance on average by **mixing** covariate values between treatment conditions. More as we go.

5.3 Designed Experiments

Here, in our notes, we go into a bit more detail here than provided in [Far14, §5.3]. Still, there remains an enormous literature on designed experiments. At NAU, you might consider STA 676 to learn more.

Three Types of Covariates. We will briefly consider three types of covariates ('more or less') **with fixing and mixing in mind**:

1. **Control Variables.** These are variables that we can control, like temperature and humidity in a greenhouse, oven temperature, or electrical current and voltage. Of course, we can **fix** these covariates among units assigned to treatments.
2. **Observed But Not Controllable.** These are variables that we can measure or observe but which we cannot control, like gender, height, weight and soil type. We can group units according to such covariates, however, more or less effectively **fixing** covariates among units assigned to treatment.
3. **Unobserved.** These are variables that we do not measure/observe, (potentially) controllable or otherwise. We cannot fix or group what we cannot control or observe. But, we can **mix** these to achieve **balance on average** among units assigned to treatments.

- **Illustration Setup.** To help us see how close we can get to our ideal of causality, consider the situation wherein we have
 - $n = 4$ units,
 - with $n_0 = 2$ units assigned (somehow) to treatment $T = 0$ and
 - the remaining $n_1 = 2$ assigned to treatment $T = 1$.

We may consider T as a factor variable with two levels (treatments), which we may have denoted as $x = 0$ $x = 1$ to be more in line with our previous covariate notation.

Our definition of individual causal effect gives

$$y_i^1 = y_i^0 + \delta_i,$$

so we can write our potential outcomes as in the following table. We observe only **one of the six possible assignments** as indicated by red and . (Why 6?)

$T = 0$	y_1^0	y_2^0	y_3^0	y_4^0
$T = 1$	$y_1^0 + \delta_1$	$y_2^0 + \delta_2$	$y_3^0 + \delta_3$	$y_4^0 + \delta_4$

Let

$$\bar{Y}_0 = \bar{y}_0^0$$

be the average of the $n_0 = 2$ actual observations for units assigned to treatment $T = 0$, and, similarly, let

$$\bar{Y}_1 = \bar{y}_1^0 + \bar{\delta}$$

be the average of the $n_1 = 2$ actual observations for units assigned to treatment $T = 1$ (\bar{y}_0^0 denotes the average of the inherent individual outcomes y_1^0, y_4^0 for those $n_0 = 2$ individuals assigned to $T = 0$, similarly for \bar{y}_1^0 of the $n_1 = 2$ units assigned to $T = 1$).

We consider using the observed difference of averages,

$$\bar{Y}_1 - \bar{Y}_0 = \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta},$$

to somehow estimate causal effects, as we now discuss in more detail.

- **Case 0: Ideal Fixing.** Under the hypothetical scenario that all units are the same (except for treatment conditions), we would have,

$$y_1^0 = \dots = y_4^0$$

(right?!) and

$$\delta_1 = \dots = \delta_4$$

(right?!) so that our statistic would measure a single, common causal effect perfectly

$$\begin{aligned}\bar{Y}_1 - \bar{Y}_0 &= \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta} \\ &= 0 + \bar{\delta} \\ &= \bar{\delta}. \\ &= \delta_i \quad (\text{same for all } i, \text{ right?}).\end{aligned}$$

Thus, we reiterate, if units were exactly **fixed** (i.e., their other covariates, fixed and equal) within and between treatment groups, aside from treatment conditions, we could isolate a single (common) causal effect, $\bar{\delta}$ —no need for Statistics! (You may be hypothetically enthusiastic!)

(To be sure, our pristine definition of individual causal effects in terms of *potential* outcomes and *counterfactual* establish that we should not consider an individual to be the same between treatments, so it would seem strange to admit multiple, identical individuals, other than hypothetically as we do here with Case 0.)

- **Question 1. Do Individual Differences (Variability) Mask an Effect?** More realistically, if units are not identical, then they differ wrt other covariates, and this difference is captured in **differences among inherent individual outcomes**,

$$y_1^0 \neq \dots \neq y_4^0$$

(right?!) and **differences among individual effects**

$$\delta_1 \neq \dots \neq \delta_4$$

(right?!) so that our statistic is

$$\bar{Y}_1 - \bar{Y}_0 = \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta}.$$

But, now, how do we know if our result

$$\bar{Y}_1 - \bar{Y}_0$$

is due to actual (average) treatment effects

$$\bar{\delta}$$

or to inherent variability among (now different) individuals

$$\bar{y}_1^0 - \bar{y}_0^0,$$

which has nothing to do with the treatment effects? Dagnabbit!

- **Question 2. Do Individual Effects Cancel?** Further, how do we know that important individual effects,

$$\delta_2, \delta_3$$

have not been cancelled or otherwise masked in the average

$$\bar{\delta}?$$

One individual may have large, positive effect,

$$\delta_3 > 0,$$

while the other individual may have a large, negative effect,

$$\delta_4 < 0.$$

In other words, there may be very **important individual effects**, of opposite sign, that give a **misleading average effect**

$$\bar{\delta} \approx 0.$$

- **Case 1: Practical (Approximate) Fixing: Similar Units.** If units cannot be identical, make units as similar as possible (i.e., make units' **controlled/observed** covariates the same, though, of course, they may still differ by their **unobserved** covariates; how would we know?). In this case, because individuals are "similar," we may expect the individuals' inherent outcomes to be similar

$$y_1^0 \approx \dots \approx y_4^0$$

and we may expect their effects to be similar, too,

$$\delta_1 \approx \dots \approx \delta_4$$

so that our statistic is

$$\begin{aligned} \bar{Y}_1 - \bar{Y}_0 &= \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta}. \\ &= \text{small} + \bar{\delta}, \end{aligned}$$

where, now,

$$\bar{\delta},$$

may be considered less likely to mislead by cancelling individual effects. (Note that these expectations seem to be implicit in our definition of causal effects in that, if we can't expect this sort of "**continuity**" as **we begin to move away from the unattainable ideal of identical units ever so slightly to "similar" units**, then similar units may result in a large $\bar{y}_1^0 - \bar{y}_0^0$ that masks a $\bar{\delta}$ that may be cancelling potentially important individual effects! This lack of continuity does not seem good to me. (Incidentally, this is akin to the notion of how badly interacting factors can cancel important main effects; we may get to this when discussing ANOVA.))

- **Case 2: Randomized Experiment: One Particular Mix.** Because, in Case 1, we cannot observe all characteristics (covariates) of units, we might randomize otherwise similar units (determined by observed covariates) to treatments with the **expectation of achieving a balance**

among units with respect to unobserved covariates and, hence with the expectation that

$$\bar{y}_1^0 - \bar{y}_0^0$$

is small and

$$\bar{\delta}$$

does not cancel important individual effects. **Still, by chance, we may get a particular randomization (mix) that masks in one or both of the two ways discussed above (Q1 and Q2).** Doh! Patience, young (and old) Padawans. (I just mixed Homer with Obi-Wan.)

- **Case 3: (Hypothetical) Replications of Randomized Experiments: Average of Mixes.** You might think that we can perform many different randomizations so that no one particular mix results in masking an effect with one particularly large

$$\bar{y}_1^0 - \bar{y}_0^0;$$

instead averaging these to get a small difference (as we would expect; why?), unmasking $\bar{\delta}$. But, in the same way, we may get a small

$$\bar{\delta}$$

from canceling individual effects. Besides, we already decided that we cannot get results from repeated randomizations. **Why?** (No going back...)

- **Case 4: Hypothetical Replications of Randomized Experiments Assuming No Treatment Effects: Average (Expected) Balance + Null Effect.** Notice (in the above table), if there are no treatment effects, i.e., $\delta_i = 0$, and we would have

$$\begin{array}{c|c|c|c|c} T = 0 & \textcolor{blue}{y_1^0} & y_2^0 & y_3^0 & \textcolor{blue}{y_4^0} \\ \hline T = 1 & y_1^0 & \textcolor{red}{y_2^0} & \textcolor{red}{y_3^0} & y_4^0 \end{array}.$$

Notice, the non-colored y_i^0 counterfactuals are the same as the colored y_i ; we have the counterfactuals in this null case. Indeed, it would not matter

which of the many possible ways units may be assigned to treatment groups in this null case (only 6 ways in our running example); we would have all of the y_i^0 and would be able to compute **compute**

$$\bar{Y}_1 - \bar{Y}_0 = \bar{y}_1^0 - \bar{y}_0^0$$

for all such possible ways (again, under the null assumption of no treatment effects, $\delta_i = 0$). If this assumption seems impractical, wait a moment.

- **Causal Effect Holy Grail: Null Randomization Distribution and p-value.** Thus, in principle, we can compute the (null) distribution of all possible values of the differences, $\bar{Y}_1 - \bar{Y}_0$, among all possible random assignments (of reds and blues), **under the null of no treatment causal effects**. Now, we ask again (**Question 1**),

“How can we determine if our particular

$$\bar{Y}_1 - \bar{Y}_0 = \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta}$$

is large because of a large $\bar{\delta}$ or because of a large $\bar{y}_1^0 - \bar{y}_0^0$ that results due to the chance imbalanced assignment of units to treatment groups?”

We can't be certain. (Doh!) But, now, our particular observed difference (large or small) is subject to the **random mechanism of repeated randomizations** captured in this **randomization distribution** of differences, and we can now calibrate the “largeness” of our (one, actual) observed $\bar{Y}_1 - \bar{Y}_0$ using probability computed from the randomization distribution. We can formalize our notion of extremeness under the null distribution with a **p-value**, as I hope you've heard about before! Thus, if our actual observed difference $\bar{Y}_1 - \bar{Y}_0$, from our single, actual experiment were “large,” (small p-value) then, as we (should) recall a familiar game, **(i) either we've observed a large difference by (small) chance, having to do with inherent the large inherent individual differences captured in $\bar{y}_1^0 - \bar{y}_0^0$ and having nothing to do with treatment effects, according to our null hypothesis assumption**

of no treatment effects, (ii) or our null hypothesis assumption of no treatment effects is wrong. Report your p-value, as usual, for the test of the null hypothesis of no treatment effect! (Some may argue for this randomization situation to be ‘usual,’ with our normal theory t and F methodology being mathematical approximations (as we will illustrate).)

- **Actual Random Mechanism.** Note, if we had not randomly assigned units to treatments, then we do not have such a randomization mechanism or its associated randomization distribution with which to compute such probabilities (arising by assuming the null under repeated randomizations). We would have no way to calibrate the largeness of our observed $\bar{Y}_1 - \bar{Y}_0$.
- **What about Question 2?** We have yet to answer the question, posed above,

“How do we know that important individual effects have not been cancelled or otherwise masked in the average $\bar{\delta}$? ”

Short answer: Again, we can’t be certain. Doh! We will address this more, shortly.

- **Take Home Message: No Fixy? Then Mixy! Or, Some Combo of Fixy Then Mixy.** With the above discussion, we reformulate our initial take-home message bullet at the beginning of this section: When we cannot balance by fixing other observed covariates across treatments (e.g., between $T = 0$ (“ x_j ”) and $T = 1$ (“ $x_j + 1$ ”)), we try to balance, on average, by mixing units with respect to their other (unobserved) covariates, if possible, and we can calibrate the chance imbalance of a particular randomization by computing a p-value under the assumption of no effects. (We will combine fixy and mixy shortly.)
- **Don’t Over-interpret.** Be careful with causal inference without fixing, mixing, or other carefully laid out assumptions.

Example: Motivation and Creativity

Example 5.1 (Motivation and Creativity Randomized Experiment).

- Here, we use the Motivation and Creativity Experiment presented in [RS13, Section 1.1.1 and 1.3] to illustrate the randomization distribution of the difference of average creative ability scores (a statistic) between two treatment groups of children. (More in class.)

- **Randomly Assign to Two Groups.** There are $n_0 = 23$ subjects that were randomly assigned to the Extrinsic treatment group, $T = 0$, and $n_1 = 24$ subjects assigned to the Intrinsic treatment group, $T = 1$.
- **Observe Data and Compute a Test Statistic.** The difference of average scores between groups for our particular randomization is $\bar{y}_1 - \bar{y}_0 = 4.14$ (an observed value of $\bar{Y}_1 - \bar{Y}_0$, discussed above). This is our **test statistic**. We could have chosen a different statistic to measure the difference between groups, e.g., a t statistic or F statistic (but will would use their randomization distribution or that of whatever statistic we chose to measure a departure from the null of no treatment effect; like in the permutation tests of chapter 3, we would not use the t or F distribution (if we had chosen here to use a t or F statistic)).
- **Extreme Under the (Null) Randomization Distribution?** Akin to our Question 1, above, is a **treatment effect** (false null) making the p-value small compared to the (null's) randomization distribution, or have we just seen a possible but extreme result under the (null) randomization distribution? (And, we may be wondering about Question 2 regarding masking or cancelling individual effects by averaging...more soon...)
- **Monte Carlo Approximate Randomization Distribution.** Comparing our observed test statistic to the randomization distribution of all

possible differences, $\bar{Y}_1 - \bar{Y}_0$, for all possible randomizations, is not practically feasible (even under the null). **How many ways** can we choose $n_0 = 23$ subjects from $n_0 + n_1 = 47$ subjects? See code. Hence, we obtain a **Monte Carlo (MC) sample approximation from/of the actual randomization distribution.**

```
> ## Example
> library(Sleuth3)
> summary(case0101)

      Score          Treatment
Min.   : 5.0   Extrinsic:23
1st Qu.:14.9   Intrinsic:24
Median  :18.7
Mean    :17.9
3rd Qu.:21.2
Max.    :29.7

> attach(case0101)
> tapply(Score,Treatment,length)

Extrinsic Intrinsic
      23         24

> (mscore<- tapply(Score,Treatment,mean))

Extrinsic Intrinsic
     15.739     19.883

> tapply(Score,Treatment,sd)

Extrinsic Intrinsic
      5.2526     4.4395

> diff(mscore)

Intrinsic
      4.1442
```

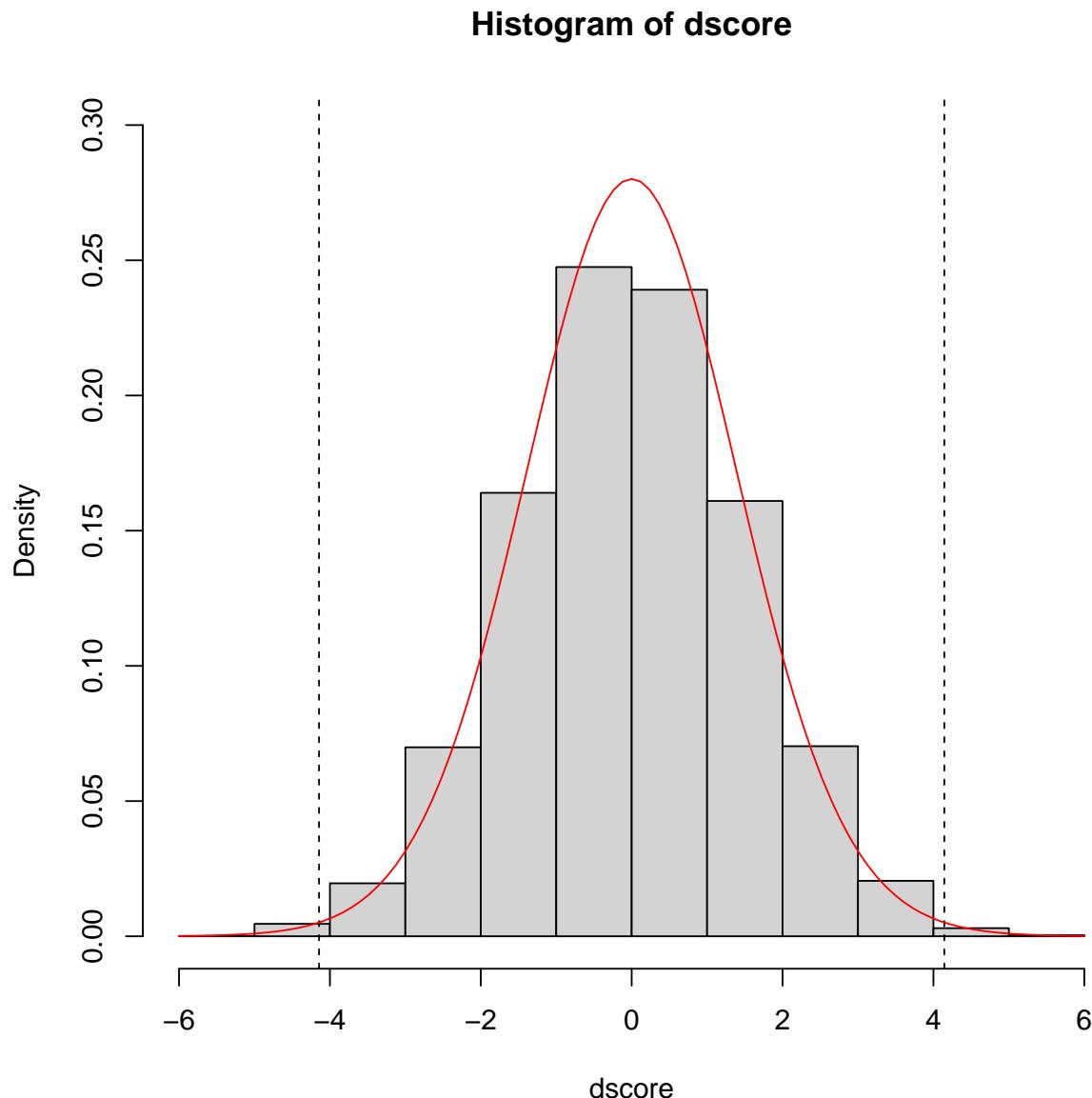
```
> choose(47, 23) ## <-- too many!  
[1] 1.6124e+13  
  
> ## Monte Carlo (MC) approximate randomization distribution:  
> ## Number of MC assignments to sample:  
> M<- 10000  
> dscore<- vector("numeric", length=M)  
> dscore[1]<- diff(tapply(Score, Treatment, mean)) ## ,<-- obs. stat  
> set.seed(8675309)  
> for (i in 2:M){  
+   ## permute/randomize (w/o replacement)  
+   mscore<- tapply(sample(Score), Treatment, mean)  
+   dscore[i]<- diff(mscore)  
+ }  
> (pval<- mean(abs(dscore) >= abs(dscore[1])))  
  
[1] 0.0065
```

- **p-value Seems Small.** So, from the previous code, the observed (approximate) two-sided p-value is 0.0065 and is illustrated as the area in the tails (indicated by the vertical lines) of the randomization distribution histogram in the nearby figure, below. We'll discuss the **red curve**, too. That means **if** the null hypothesis of no treatment effect is true, then the probability of observing a difference of average creativity scores at least as extreme as the one we actually observed (4.14) is about 0.0065, or 65 in 10000.
- **Perhaps Formally Reject/Not Reject.** Of course, such a small p-value is possible under the null, but it tends to makes us want to reject the null hypothesis as false. Anticipating rejection, we may have chosen to make the probability of a false rejection to be small, i.e., we may have chosen the **significance level (Type I error rate or $P(\text{type I error})$)**, e.g., $\alpha = 0.05$. Thus, when we reject when p-value $\leq \alpha$, we control the type I error at α , if we must make such a binary decision to reject or not reject. (We should always choose α before looking at the data or such results, and it may be more appropriate to simply report the p-value.)

- **Confidence Interval?** As mentioned in the unnumbered section, Reporting Test Results, in Lecture 3, perhaps we also would construct a confidence interval for the treatment effect (not shown here) to get an idea of what values of the treatment effect are consistent with our observed data and how these values may or may not be scientifically meaningful (important or “scientifically significant” or “practically significant” so to speak).

t Approximation, Randomization Distribution & Permutation Distribution

```
> ## Plot histogram approximating the randomization distribution and
> ## (just out of curiosity) compare to normal linear model t
> ## distribution (properly scaled in plot for comparison to histogram).
>
> ## For normal theory lm comparison:
> case0101.sum<- summary(case0101.lm<-
+                         lm(Score ~ Treatment, data=case0101))
> ## (next assumes treatment coding...more with ANOVA later)
> se<- case0101.sum$coefficients[2,2]
>
> ## MC approx. randomization distribution
> hist(dscore, freq=FALSE, ylim=c(0,0.30))
> abline(v=c(-1,1)*dscore[1], lty=2)
>
> ## Compare to t from our normal lm (properly scaled)
> curve(dt(x / se,df=23+24-2) / se, col="red", add=TRUE)
```



```
> ## Compare to p-value from normal lm t (red)
> 2*pt(abs(dscore[1])/se, df=23+24-2, lower=FALSE)
[1] 0.0053665
> detach(case0101)
> detach(package:Sleuth3)
```

- Notice the **t distribution** in the plot. We use it here to illustrate that a **mathematical model** is often used to approximate distributions like the randomization distribution. (That the distribution of the differences of averages, $\bar{Y}_2 - \bar{Y}_1$ (divided by the appropriate estimated standard error), appears well approximated by the t distribution should be no surprise to those of us who have heard of the Central Limit Theorem: the difference of averages would be approximately normal and, when scaled by the appropriate estimated se, we get (approximately) a t. See Appendix A for CLT.)
- Indeed, we will often assume such a model approximation (not without checking our assumed model, however!), and we forget about (or never knew) the underlying distribution being approximated.
- But, it may be good to keep this approximation in mind. At least, it is a direct reminder of the underlying hypothetical replication interpretation of frequentist statistics.
- What's the p-value from the t approximation? See code, above.
- See [Far14, §3.3] for a discussion of a **permutation test**, in an observational context, as an alternative to normal-based procedures when normality fails. Operationally, the permutation test is the same as the test from a randomization distribution, but in the observational context, however, there is no real random mechanism—like randomizing subjects to treatments—to justify the permutation test as there is with a randomized experiment, and the scope of inference is thus limited in the observational context. See [Far14, §5.3] for a brief mention of the “permutation test” (our randomization test) in the randomized experiment case. (Some people, including your author, use permutation test / randomization test interchangeably. Not us.)
- In this current, randomized experiment case, the permutation test sometimes goes by **randomization test** as we have called it here to clarify

the actual random mechanism of randomly assigning units to treatment groups.

Formalizing Some Concepts

- Our previous discussion and example illustrate a few basic concepts that we may have discussed without formal definitions. Let's define them, if still a bit loosely. (We repeat the definition of the p-value given previously in Lecture 3, modified to fit the randomization distribution context.) Granted, this may seem a bit of a non-sequitur. Oh well.

Definition 5.1 (p-value). *The probability of a (randomization resulting in a) test statistic value from the (randomization) null distribution being as extreme as or more extreme than the actual value observed (from the actual randomization obtained in a randomized experiment).*

You may want to look back to Lecture 3 for interpretations of the p-value.

Definition 5.2 (Parameter).

- A parameter is an (usually) unknown constant associated with a (mathematical model of a) distribution of values and is often the target of inference. It's typically only hypothetically observable (e.g., population mean, treatment effect, regression effect, etc.).
- Usually, questions (and answers) are phrased in terms of parameters or functions of parameters. ("Is the data consistent with the assumption

$\delta = 0?$,” in the case of a test or p-value, or “What is/are values of δ that are consistent with the observed data?,” in the case of a confidence interval)

- To be more precise, in our experimental example, above, we saw that our parameter may be thought of as $\delta = \bar{\delta}$, an average of individual causal effects. But almost always, discussion simply ignores the underlying individual details, and we simply think of a single effect parameter, δ . In any case, may want to keep the underlying details in mind for later discussion when we (finally!) return to our Question 2, “How do we know that important individual effects have not been cancelled or otherwise masked in the average $\bar{\delta}$?“

Definition 5.3 (Statistic). Some function of data that does not depend on unknown parameters.

Definition 5.4 (Estimator/Estimate). When a statistic is used to infer about a parameter, we often call the statistic an estimator (before plugging in values) or an estimate (after plugging in values).

- Note that, before we plug in our actual data values, $\bar{Y}_1 - \bar{Y}_0$ is uncertain, i.e., it's an example of what we defined to be a random variable, whose uncertain values are characterized by its (randomization) distribution (arising from the randomization mechanism) that we used to answer the question of whether our data is consistent with the assumption $\delta = \delta_0$ (or could have used to construct an interval to see which values of δ are consistent with the data).

- In particular, we used the (randomization) **distribution** of $\bar{Y}_1 - \bar{Y}_0$ under the assumption that $\delta = 0$.
- The statistic, $\bar{Y}_1 - \bar{Y}_0$, is also an **estimator/estimate** because we used it to infer about the parameter, δ .
- We also may call $\bar{Y}_1 - \bar{Y}_0$ a **test statistic**, because we used it to “test” whether our data was consistent with the assumption $\delta = 0$, though this terminology is more consistent with the case where we must make a binary decision to reject or not reject the null hypothesis.

Restricted Randomization

- **Question 2.** Finally, we get to this question in this and the next (un-numbered) sections.
- **Re-iterating Example (See Previous Table).** Suppose male patients inherently tend to show larger responses than females, regardless of any treatment effects and, further, assume that males show positive causal effects to a treatment while females tend to show negative effects. If we attempt to balance by mixing (instead of fixing) males and females among treatment groups, then we may have, by chance, all **males in one group** and **females in the other group**, and we may get large inherent effects,

$$\bar{y}_1^0 - \bar{y}_0^0$$

in

$$\bar{Y}_1 - \bar{Y}_0 = \bar{y}_1^0 - \bar{y}_0^0 + \bar{\delta},$$

due to gender differences across groups, regardless of treatment effects. In addition, because males tend to show positive treatment effects

$$\delta_2, \delta_3 > 0$$

(as in our previous discussion), and females tend to show negative treatment effects

$$\delta_1, \delta_4 < 0.$$

Thus, our result, $\bar{Y}_1 - \bar{Y}_0$ will reflect an effect for males

$$\bar{\delta} > 0$$

because males happened to be assigned to $T = 1$, by chance, which would be badly misleading for the effect of females

$$\bar{\delta} < 0,$$

which our particular randomization did not capture, and the effect for male would appear larger than it should be, $\bar{\delta} > 0$.

We can imagine other unfortunate, chance results. We could have 1 male and 1 female in each group, so that

$$\bar{y}_1^0 - \bar{y}_0^0$$

may be close to zero, but, then

$$\bar{\delta}$$

would cancel, e.g., a positive male effect $\delta_2 > 0$ with a negative female effect $\delta_3 < 0$, which would be misleading for both males and females.

- **Fixing & Restricted Randomization.** Of course, as we've said before, we would fix (observed or controlled) covariates among units prior to assigning treatment (No fixy? Then mixy.) Thus, in this example, of course, we would 'fix' gender at males, (randomly) assigning treatments to males, then fix gender at females, and (randomly) assigning treatment to females. We have **restricted randomization** of treatments to males (or vice versa) and restricted randomization of treatments to females (or vice versa). Then, we compare treatment effects within males and within females to help us avoid misleading conclusions about effects across gender.

- **Question 2 Revisited.** Thus, we have, at least, a partial answer to our question, “How do we know that important individual effects have not been cancelled or otherwise masked in the average $\bar{\delta}$?“ We’ve used restricted randomization. (Still, of course, we might have masking problems of type Q1 and Q2, by chance, as we’ve discussed, despite randomization, but, again, we can calibrate the chance of making a Type I error (using the randomization distribution(s) under the null(s) of no effect(s) (male or female); we can compute p-values.
- **And Restricted Inference!** Restricted randomization of treatments to units with the same values of all observable covariates seems to take a one-sided view of our definition of individual causal effects. In particular, it attempts to emulate the notion of sameness of units among treatment groups, embodied by the counterfactual, to isolate causal effects separate from these potentially confounding covariates. This may sound great, but it ignores the fact that individuals have different causal effects! Thus, our inference of cause and effect, if any, will be **restricted** to relatively **homogenous units**. How can we say what would have happened if the fixed value of the covariates (i.e., units/subjects and their individual causal effects) were different? If you only studied females, what can you say about males (or vice versa)? If you fixed the greenhouse temperature and humidity, what can you say about plants’ response to (e.g., fertilizer) treatments under different temperatures and humidities?

Replicate Treatments

- **Answer: More Fixy & Mixy.** We **replicate the restricted randomization of our basic treatment structure across a range of fixed covariate values (units/subjects) of interest**, thus, increasing the scope of inference to a larger variety of units. In our previous example with males and females, we get inference about each. And, e.g., we get

inference about plants' response to fertilizer treatments across a range of temperatures and humidities, etc.

- **Blocks.** The fixed levels/combinations of covariate values within which our randomization occurs are called **blocks** in a **randomized complete block design (RCBD)** ([Far14, Chap. 17]). If there are only two treatment levels, e.g., then we often only have two units per block, one unit receiving one treatment, the remaining unit receiving the other, blocks are sometimes called **(matched) pairs**. Two males in one block, two females in another.

For another cheap matched pairs example, we might have two eye drop treatments, a new eye drop treatment and a standard eye drop treatment. In this case, we might use a person as a block (matched pair (of eyes)), randomly assigning one eye drop treatment to one eye (observational unit), with the remaining eye (obs. unit) receiving the other treatment. (In this case, the blocks (people) are often considered random, not fixed conditions, but that is getting beyond us in INF 511; perhaps more in INF 512.)

Thus, with multiple blocks, we have multiple replicates of all treatment conditions, which we hope will help us more powerfully detect treatment effects. (But, responses to treatments may differ across blocks! (block x treatment interaction) Doh! Time to take STA 676 Experimental Design!)

- **Control Orthogonality.** If the observable covariates are under our **control**, then we may choose their values to achieve **orthogonality**, which helps to simplify conclusions about causal effects. See [Far14, §2.11, Chap. 16, 17].
- **It's a Big World Out There.** We have only just scratched the surface of the design of randomized experiments (again, consider STA 676), but we have covered fundamental concepts that should serve us well in statistics more generally, as we now move to more discussion on observational (non-experimental) data.

5.4 Observational Data

Definition 5.5. *Observational Data*

- **No Randomization.** *Observational data arise from studies that do not consist of randomized experiments, i.e., that are not observed from units that have been randomly assigned to treatment groups.*
- **Limited Scope of Inference Regarding Causality.** *Importantly, causal inference from observational data is, strictly speaking, not justified; our scope of inference regarding causality is relatively limited without a causal model.*
- **Association (or Correlation) Is Not Causation.** *This is an old adage for observational studies.*

- **E.g. Scholastic Performance and Socio-Economic Factors.** We may compare scholastic performance (response) as measured by, say, **GPA or SAT scores**, across different observed **socio-economic factors**, but we cannot randomly assign subjects to these “treatments” to mix up other, unmeasured, potentially confounding covariates within and among the different socio-economic “treatments” (observational factor (covariate) level combinations).
 - **Effect Not Causal.** An effect of, e.g., household income may be confounded with some other, unmeasured covariates so that we are not on firm ground to claim that the effect of household income is causal: changing income by 1 unit (as if we could) does **not cause** an effect, β_{income} , to (mean) performance, despite our notion of this so-called effect being obtained by **fixing** values of the remaining observed socio-economic factors, as we discussed near the beginning of this chapter in the context of the Galapagos example. (But, now

we see how this interpretation of effect getting at the notion of fixing, perhaps in attempt towards the ideal of causality.)

- **Galapagos Example Not Causal.** Or, in the Galapagos example, while we have observed several covariates, other than elevation, we cannot strictly infer that $\beta_{elevation}$ is caused by a 1-meter change in elevation, despite fixed values of other covariates, because there may be other, unobserved covariates that we did not mix. (How could we randomly assign islands to receive covariate values of elevation, area, etc., to mix up unobserved covariate values?)
- **No Metal.** Thus, **observational studies fall short of our gold standard of randomization and causality.**
- **Is There Any Consolation?** Having no metal, how can we still feel good about observational studies, **how close we can get to the gold standard?** The notions of **fixing or mixing to balance covariates** again play a role, of course. More to come, shortly.

Definition 5.6 (Confounding Variable).

- *Loosely speaking, a confounding variable is a variable that is related to both (treatment) group/level membership (a covariate) (i.e., to an "x" variable) and to an outcome variable (i.e., to the response or "y" variable). See [Far14, p. 65] for an informal discussion of confounder.*
- **Association May Not Be Causation.** *Thus, as we've said in an example, above, an apparent association between group membership (or covariate) and an outcome may not be due to a direct (causal) relationship between group membership (or covariate) and an outcome, but to their shared relationship with other, confounding variables, which we often do not or cannot observe. (Again, 'correlation (or association) is not causation' when it comes to observational studies.)*

- In other words, **confounding variables can make causal inference problematic** in observational studies.
- Lurking variable or unobserved/unmeasured covariate or confounder are frequently used synonyms.
- E.g. Though people living in households with more televisions may tend to be healthier than those living with fewer televisions, we do not say that televisions cause people to be more healthy (or that better health causes more televisions); in this case, factors such as diet, exercise and health care may be confounding factors associated with both health outcomes and material wealth, like TVs.
- More Precisely. There are relatively precise, technical definitions of confounding variable, and these definitions are typically used in causal modeling (see remark below about causal inference). We do not cover causal modeling or more precise definitions.

Remark 5.1 (Causal Inference).

- **Without Randomization, Then What?**
- **Causal Model.** Without the ability to randomize causal information cannot justifiably be drawn from observational data alone. But, if we make assumptions about the nature of causality, i.e., if we assume a causal model for the data, then causal inferences may be drawn from observational data, dependent on causal modeling assumptions being correct.
- **Field of Causal Inference.** Such causal modeling is the subject of the developing field of Causal Inference (e.g., causal diagrams, path analysis, propensity scores, potential outcomes, interventions, counterfactuals) ([Pea09], [PM18]). (You may find that our brief treatment above will give you some insight into causal inference talks/papers.)

- *We will largely ignore causal modeling and the field of causal inference in this course, but this should not be taken to mean that the field of causal inference has nothing to offer!*
- *See also Qualitative Support for Causation ([Far14, §5.7]).*

Voting Example

- We illustrate **confounding** with observational data by following the New Hampshire primary voting example in [Far14, §5.4].
- Of interest is the number of votes for Obama and for Clinton in the January 8th 2008 New Hampshire Democratic primary.
- Contrary to pre-primary polling, Clinton defeated Obama. Some thought that it could be due to different voting technologies, which most people would argue should not matter.
- Among machine-counted votes ("D"igital), Clinton defeated Obama, while Obama had more votes among paper ballots ("H"and).
- Did voting technology have a causal effect on the outcome? Was election integrity somehow compromised? Was there voting fraud?
- (Randomization seems a bit far fetched here, but, we will demonstrate, in subsequent sections, how we can use the notion of **balance by fixing, or covariate adjustment**, in an observational study such as this voting example.)

```
> ## A brief look at the data:  
> data(newhamp, package="faraway")  
> names(newhamp)
```

```
[1] "votesys"      "Obama"        "Clinton"       "dem"          "povrate"
[6] "pci"          "Dean"         "Kerry"         "white"        "absentee"
[11] "population"   "pObama"

> ## `D'igital counts
> colSums(newhamp[newhamp$votesys == 'D', 2:3])

Obama Clinton
86353    96890

> ## `H'and counts
> colSums(newhamp[newhamp$votesys == 'H', 2:3])

Obama Clinton
16926    14471
```

We take as our response the proportion of votes for Obama. As a first attempt, we fit an SLR of proportion of votes for Obama against voting technology “treatment”, coded as 0 for ‘D’igital and 1 for ‘H’and. Thus, the so-called “effect” of voting technology is β_1 , indicating the change in percent of voters for Obama from digital to hand technology, i.e., in some sense, β_1 indicates how much hand counting favors Obama over digital technology. (Note, you may have coded the voting systems in the opposite way, 0 for hand and 1 for digital, so that β_1 would then indicate the change in percent of voters for Clinton from hand to digital technology, because the suspicion was that digital technology favored Clinton. Not a big deal. Potato, potato.)

```
> ## Initial chosen (SLR) model  $E(Y/T) = b_0 + b_1 T$ .
> ## Is percentage voting for Obama affected by technology?
> newhamp$trt <- ifelse(newhamp$votesys == 'D', 0, 1)
> lmodu <- lm(pObama ~ trt, newhamp)
> summary(lmodu)
```

Call:
`lm(formula = pObama ~ trt, data = newhamp)`

Residuals:

Min	1Q	Median	3Q	Max
-0.17215	-0.04764	-0.00452	0.04036	0.22911

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.35252	0.00517	68.15	$< 2e-16$ ***
trt	0.04249	0.00851	4.99	0.0000011 ***
<hr/>				
Signif. codes:	0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1			

Residual standard error: 0.0682 on 274 degrees of freedom

Multiple R-squared: 0.0834, Adjusted R-squared: 0.0801

F-statistic: 24.9 on 1 and 274 DF, p-value: 0.00000106

Remark 5.2. Confounding in Observational Regression Studies

- **Assumed Model.** Let's say that we **choose a model**,

$$Y | T = \beta_0 + \beta_1 T + \epsilon.$$

This is our SLR model in the code, above.

- **Truth.** But, suppose the (unknown) true relationship is (an MLR model, adding Z)

$$Y | T, Z = \beta_0^* + \beta_1^* T + \beta_2^* Z + \epsilon,$$

and assume we are interested in the effect of **observed covariate**, T , on Y , i.e., β_1^* , but we **do not observe** Z . Further, assume that Z and T are **linearly associated** via

$$Z | T = \gamma_0 + \gamma_1 T + \epsilon'$$

- **Confounder.** Thus, unobserved Z is related to both Y and to observed covariate, T ; i.e., Z is a **confounder** by our definition (5.6).

- **Potential Bias.** We want to know β_1^* in the true model, but β_1 , in our chosen model, may not equal β_1^* . In other words, β_1 may contain other effects and may be biased against β_1^* , without randomization. That is, without randomization, we should not think of β_1 as the effect on the population mean of Y caused by a unit increase in the covariate T in the population. We may think of it only as indicating an association with T arising by the association of the confounder Z with both Y and T .
- By the **law of iterated expectations** (just using iterated integrals / sums to compute multiple (double here) integrals/sums)([CB02, Theorem 4.4.3])

$$\begin{aligned}
 E(Y | T) &= E_{Z|T}(E(Y | T, Z)) \\
 &= E_{Z|T}(\beta_0^* + \beta_1^*T + \beta_2^*Z) \\
 &= \beta_0^* + \beta_1^*T + \beta_2^*E(Z | T) \\
 &= \beta_0^* + \beta_1^*T + \beta_2^*(\gamma_0 + \gamma_1T) \\
 &= (\beta_0^* + \gamma_0\beta_2^*) + (\beta_1^* + \gamma_1\beta_2^*)T \\
 &= \beta_0 + \beta_1T.
 \end{aligned}$$

(For the interested student, the first few equalities above may be seen as

$$\begin{aligned}
 \int_z \int_y y[y, z] dy dz &= \int_z \int_y y[y | z][z] dy dz \\
 &= \int_z E(y | z)[z] dz \\
 &= \int_z (\beta_0^* + \beta_1^*T + \beta_2^*z)[z] dz \\
 &= (\beta_0^* + \beta_1^*T + \beta_2^*E(Z | T)),
 \end{aligned}$$

etc., as above, and we have used square bracket notation for joint $([y, z])$, conditional $([y | z])$ and marginal $([z])$ distributions; see Appendix §B.6.)

- **Bias More Precisely.** Thus, we see in this case of confounding that our SLR parameter $\beta_1 = (\beta_1^* + \gamma_1\beta_2^*)$. Thus, unless T and Z are uncorrelated ($\gamma_1 = 0$) or Z does not affect Y ($\beta_2^* = 0$), then Z is a confounder, and

we see its influence on biasing our SLR β_1 away from the true β_{1} by an amount $\gamma_1\beta_2^*$.*

- **Estimated Effect in Our Example.** *The estimated effect in our chosen SLR model is $\hat{\beta}_1 = 0.04249$ (increase in proportion of voters for Obama when voting by hand vs digital), which appears highly inconsistent with the null hypothesis that voting technology had no effect on proportion of votes, and seems to question voting system integrity (p-value = 0.0000011). Hold on, Chicken Little, before you lobby congress to change voting laws, let's consider if there is a third, potential confounding variable, Z, contributing to bias that may be reflected in this estimated effect. (Of course.)*

Adjust for the Confounder. In the code, below, we add the potential confounder, ‘political outlook’ ($Z=$ Dean, proportion of voters for Howard Dean in the 2004 Democratic primary), to our previous SLR model. Note how the “effect of voting technology” (increase in proportion of voters for Obama when voting by hand vs digital) is estimated to be about 10 times smaller and is now no longer significant after having **fixed, or adjusted for, the other covariate** $Z=$ Dean, which is estimated to account for a very highly significantly large proportion of voters for Obama. (Imagine how the conspiracy theory of vote tampering would fly today! (Too well, I’m afraid.))

```
> ## Add potential confounder, `political outlook' (Z=Dean), to get
> ## ``true'' model (e.g.), above,  $E(Y/T, Z) = b*0 + b*1 T + b*2 Z$ 
> lmodz <- lm(p0bama ~ trt + Dean, newhamp)
> summary(lmodz) ## Faraway's sum(m)ary function
```

	Estimate	Std. Error	t	value	Pr(> t)
(Intercept)	0.22112	0.01125	19.65	<2e-16	
trt	-0.00475	0.00776	-0.61	0.54	
Dean	0.52290	0.04165	12.55	<2e-16	

n = 276, p = 3, Residual SE = 0.054, R-Squared = 0.42

Some Discussion. As we expect for a confounder, $Z=\text{Dean}$ is very significantly positively related to our covariate of interest (voting technology), too. Thus, the increase in voters for Obama between hand vs digital technology in the unadjusted SLR is plausibly due to the positive association that the $Z=\text{Dean}$ confounder has with proportion of voters for Obama and with voting technology. (Note that $Z=\text{Dean}$ reflects 2004 primary results, whereas voting technology and voters for Obama reflect 2008 results; it seems implausible that past political preference caused differences in voting technology across voting precincts in a conspiracy to elect Obama over Clinton in 2008; it's more likely that there just happens to be a coincidental tendency for 2004 "Dean precincts" to have used hand technology in 2008. (I'm not sure what the underlying causal mechanism might have been. Could it have been that urbanites favored Clinton and that cities have larger tax bases to better fund election technologies?....)

In any case, we have illustrated the concept of fixing, or covariate adjustment, in the MLR (adding $Z=\text{Dean}$) in the context of an observational study. That is, we are back to our simple meaning of effect where we see β_1 , in the MLR as the change in mean percent voters for Obama with a unit change in a covariate (voting technology) while holding fixed ("controlling for") remaining covariates ($Z=\text{Dean}$).

```
> ## Potential confounder, 'political outlook' (Z=Dean), is related to
> ## covariate of interest, trt:  $E(Z/T) = g_0 + g_1 T$ 
> summary(lm(Dean ~ trt, newhamp)) ## Faraway's sum(m)ary function.
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.25129	0.00599	41.99	<2e-16
trt	0.09034	0.00985	9.18	<2e-16

n = 276, p = 2, Residual SE = 0.079, R-Squared = 0.24

5.5 Matching

- **Matching.** Similar to the notion of fixing and adjusting or controlling for a covariate by including it in the model, as above, we might instead try to find pairs of districts that have the same or similar (almost fixed at the same) values of political preference ($Z=$ Dean, confounder) and compare proportion voting for Obama between pairs of precincts balanced for or (almost) fixed on /matched on $Z=$ Dean, one precinct with hand technology, the other with digital technology. That is, as we discussed above, we make units similar before comparing the effect of “treatments” (voting technologies). Because $Z=$ Dean is a continuous variable, we have to be tolerant and look for districts that are closely similar in the $Z=$ Dean variable, if not fixed at exactly the same value.
- **Matching is Free of Our Model Assumptions.** Matching does not rely on a model (thus is more free of model assumptions), but, as we see, it gets at the same notion of fixing/balancing (perhaps more transparently than covariate adjustment does), and we (your text's author) will compare results of matching to covariate adjusting for $Z=$ Dean in a subsequent section. (As you will see, our model, and it's covariate adjustment, give very similar results to matching, in this case, but matching may be use when we don't have a model.)
- **Matching Library.** We use the `matching` library to do the matching of precincts with similar values of $Z=$ Dean but with different voting technologies. (It uses a genetic algorithm. OOOOooooh! Further discussion omitted!)

```
> require(Matching) ## likely need to download first
```

```
Loading required package: Matching
```

```
Loading required package: MASS
```

```
## 
## Matching (Version 4.10-2, Build Date: 2022-04-13)
## See http://sekhon.berkeley.edu/matching for additional documentation.
## Please cite software as:
##   Jasjeet S. Sekhon. 2011. ‘‘Multivariate and Propensity Score Matching
##   Software with Automated Balance Optimization: The Matching package for
##   R.’’
##   Journal of Statistical Software, 42(7): 1-52.
## 

> set.seed(123)
> mm <- GenMatch(newhamp$trt, newhamp$Dean, ties=FALSE,
+                  caliper=0.05, pop.size=1000)

Loading required namespace: rgenoud
```

```
> ## First 6 matching pairs of districts and their Obama proportions,
> ## first column hand (coded 0) second column digital (1)
> head(mm$matches[,1:2])

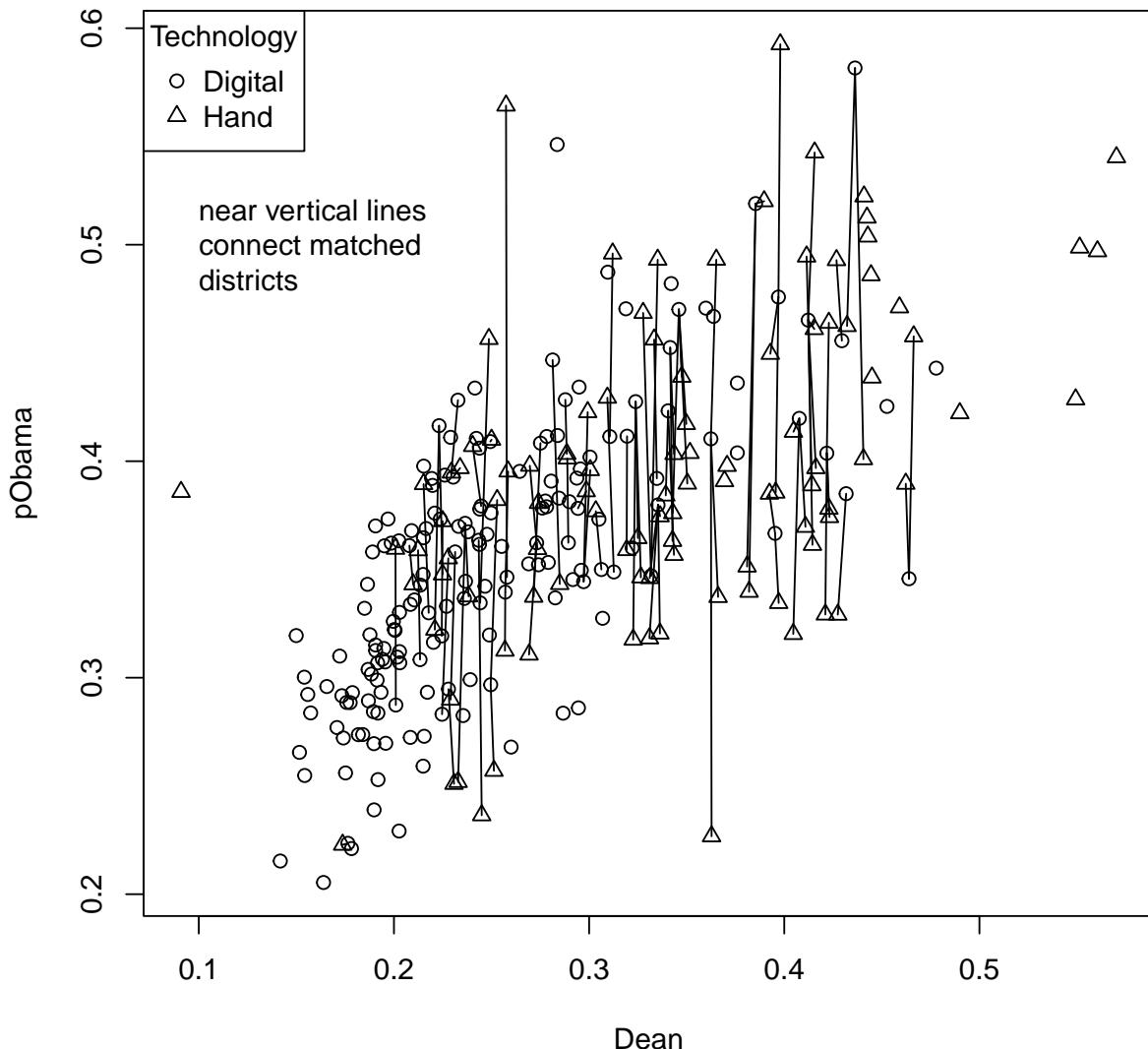
 [,1] [,2]
[1,]    4 107
[2,]   17 242
[3,]   18  53
[4,]   19  83
[5,]   21 246
[6,]   22  95

> ## A closer look at the first pairing:
> newhamp[mm$matches[1,1:2],c('Dean','pObama','trt')]

          Dean pObama trt
CenterHarbor 0.28495 0.34328  1
Amherst      0.28131 0.44676  0
```

```
> ## Let's plot all matching pairs of proportions, along with remaining
> ## proportions as a function of political preference (Z=Dean)
> plot(pObama ~ Dean, newhamp, pch=trt+1)
> with(newhamp, segments(Dean[mm$match[,1]],pObama[mm$match[,1]],
+                         Dean[mm$match[,2]],pObama[mm$match[,2]]))
> legend("topleft", legend=c("Digital","Hand"), pch=c(1,2),
```

```
+       title="Technology")
> text(0.1,0.5,labels=c("near vertical lines\nconnect matched\ndistricts"),
+       adj=0)
```



- **Fraud?** First, without looking at $Z=\text{Dean}$, we see more **hand-counting districts (triangles)** at higher percent voters for Obama and more

digital-counting districts (circles) at lower percent voters for Obama. (Pretend Z=Dean isn't there and project the triangles and circles to the left onto the vertical axis.) This suggests that, like our SLR, voting technology is associated with voting outcome, but we know to be leery about making a causal inference with this observational study.

- **Not So Fast!** But, we see Z=Dean (political outlook) positively associated with voting outcome, too, and with voting technology—we see more hand counting districts (triangles) at higher political outlook levels and more digital counting districts at lower political outlook. This association of Z=Dean with voting technology and with voting outcome is what we called a confounder (when not observed and accounted for), and is likely a more plausible explanation for the variability in voting outcome. Without further evidence of voting fraud, this may be sufficient to drop further investigation.
- If we look at pairs of districts, one digital (circles) and one hand (triangles), matched (connected by line segments) for similar values of Dean (i.e., we fix Dean before differences outcomes between treatments); then it's difficult to see an systematic positive or negative association of technology with outcome; we see a mix of positive and negative differences in percent voters for Obama. Thus, we no longer see an obvious association of voting technology with voting outcome. That is, differences in the percent voter's for Obama between matched districts appear as if the average difference could be about zero. Let's do a single sample t-test on the difference in percent voters for Obama (null of zero difference) to see if it agrees with what we see in the plot (chunk below).
- Yep. When voting preference is taken out of the mix, by fixing it to provide a balanced comparison of outcome between voting technologies, results are no longer terribly inconsistent with the null of no effect of voting technology—no fraud ($p\text{-value}=0.064$)—and the associated confidence interval of differences may be considered to contain small, unimportant differences. (I don't know why our results differ slightly from

[Far14, §5.5]; you might see slight differences in our above plots compared to [Far14, Fig. 5.2], too. Perhaps the random matching algorithm has changed; set.seed did not work as intended!)

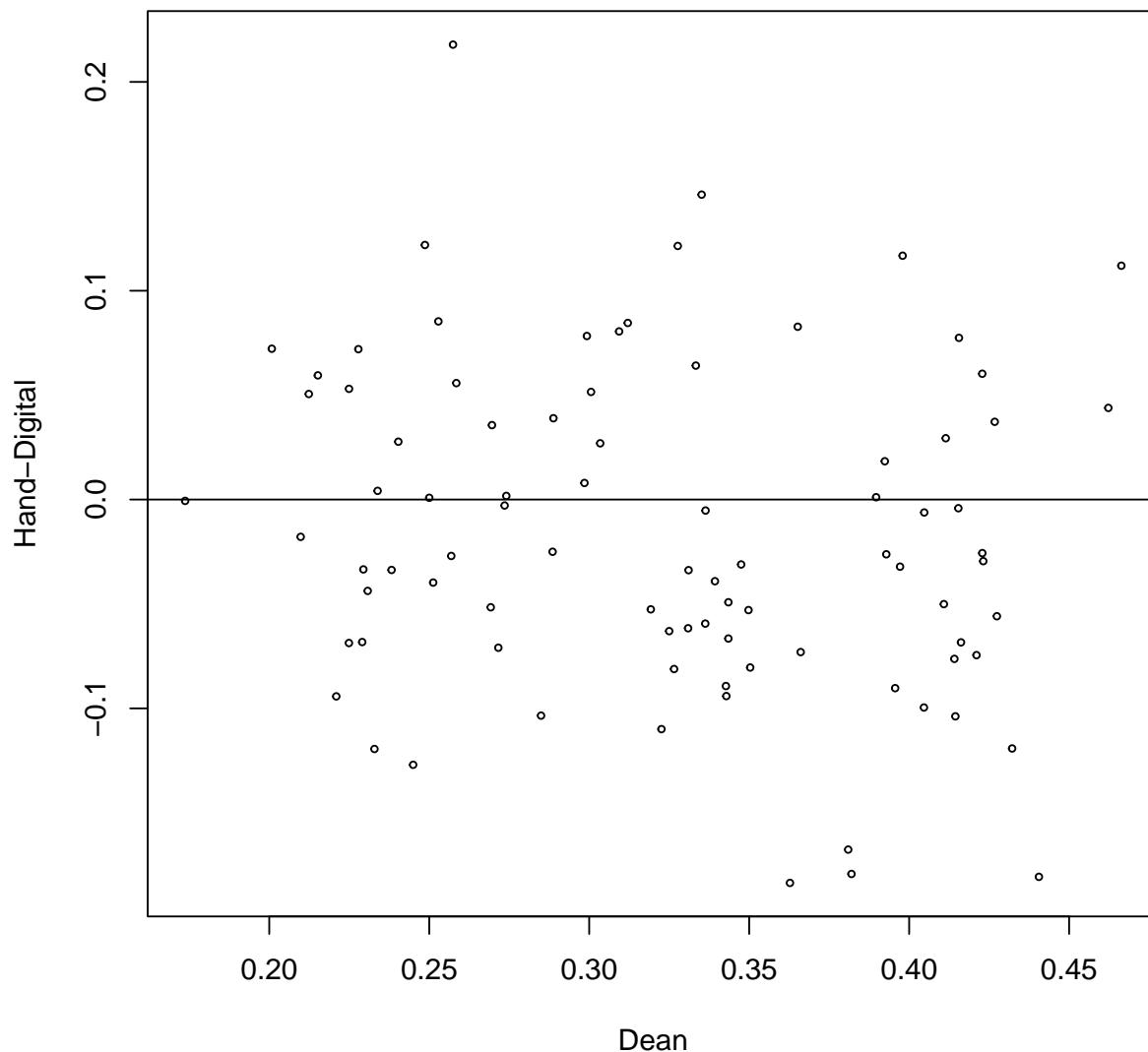
This next chunk just shows that matching on Z=Dean before differencing removes (controls for. or adjusts for) Z=Dean—there's no association with Z=Dean left in the differences, thus Z=Dean can no longer confound the inference about the differences in outcome between treatments. And, we see again the average difference between treatments appears to be about zero. Thus, we may think that it was likely not fraud but association with the lurking variable Z=Dean that cause initial concern.

```
> pdiff <- newhamp$p0bama[mm$matches[,1]] - newhamp$p0bama[mm$matches[,2]]
> t.test(pdiff)

One Sample t-test

data: pdiff
t = -1.92, df = 86, p-value = 0.058
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
-0.03284473 0.00053205
sample estimates:
mean of x
-0.016156

> plot(pdiff ~ newhamp$Dean[mm$matches[,1]], xlab="Dean", ylab="Hand-Digital",
+       cex=0.5)
> abline(h=0)
```

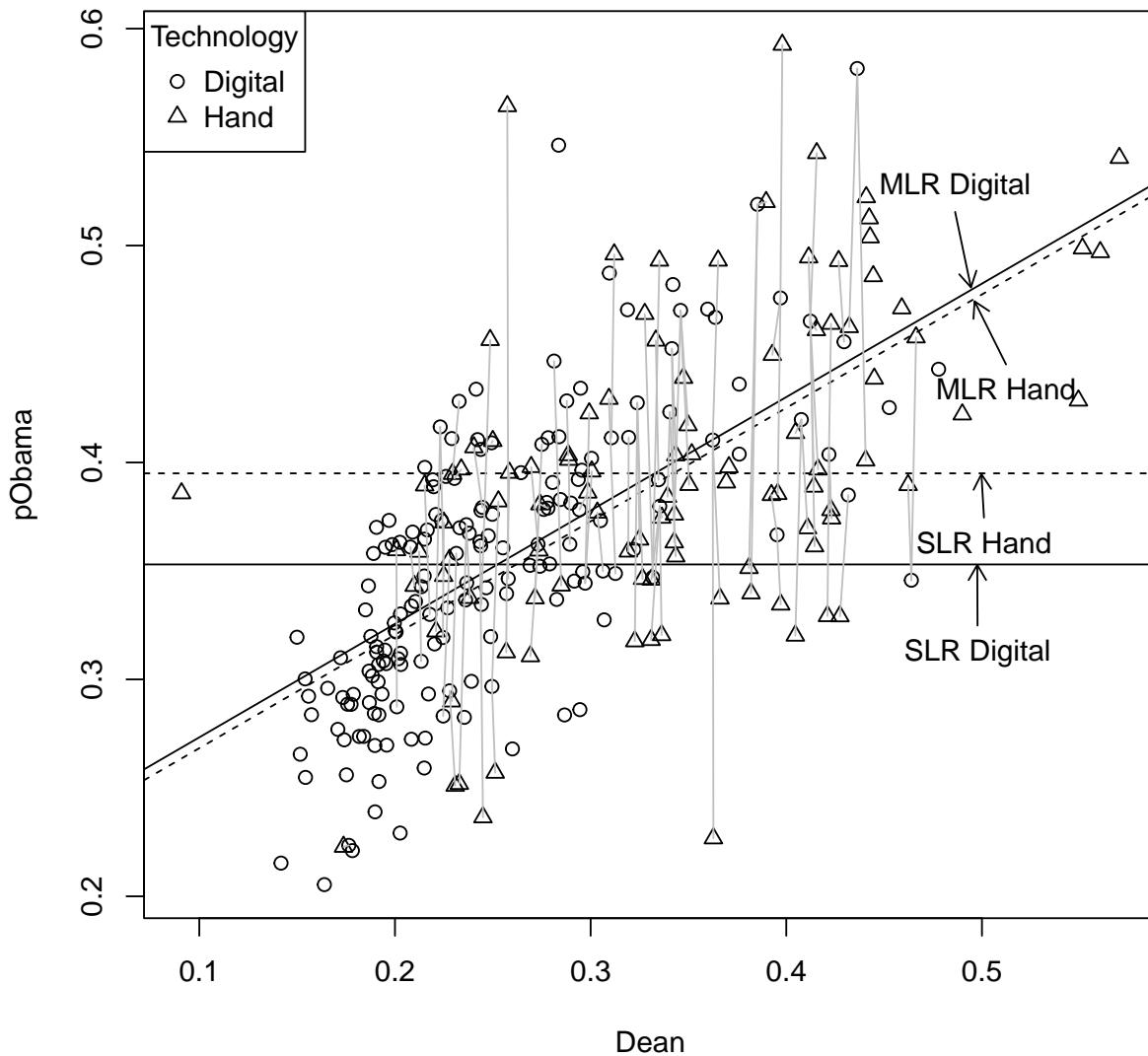


5.6 Covariate Adjustment

The above SLR on treatment compares percent voting for Obama, without considering the potential confounder, $Z = \text{Dean}$. The above MLR adds (**fixes** or adjusts for) $Z = \text{Dean}$ before comparing percent voting for Obama between

hand and digital districts. Big difference in conclusions, eh? (more in class)
 Again, this covariate adjustment is in the same spirit of controlling/adjusting
 for potential confounders as matching.

```
> plot(pObama ~ Dean, newhamp, pch=trt+1)
> ## SLR digital and hand:
> abline(h=c(.353, 0.353+0.042), lty=1:2)
> ## MLR digital
> abline(0.221, 0.5229)
> ## MLR hand
> abline(0.221-0.005, 0.5229, lty=2)
> ## Indicate matching districts
> with(newhamp, segments(Dean[mm$match[,1]], pObama[mm$match[,1]],
+                         Dean[mm$match[,2]], pObama[mm$match[,2]],
+                         col=gray(0.75)))
> legend("topleft", legend=c("Digital", "Hand"), pch=c(1,2),
+        title="Technology")
> ##tmp<- as.data.frame(locator(n=8))
> tmp<- data.frame("x"= c(0.48628, 0.49438, 0.51232, 0.49622,
+                      0.50103, 0.49973, 0.49774, 0.49761),
+                     "y" = c(0.51553, 0.48074, 0.44519, 0.47428,
+                            0.37457, 0.39463, 0.32482, 0.35279))
> arrows(tmp[c(1,3,5,7),"x"],tmp[c(1,3,5,7),"y"],
+         tmp[c(2,4,6,8),"x"],tmp[c(2,4,6,8),"y"],
+         length=0.1)
> text(tmp[c(1,3,5,7),"x"],tmp[c(1,3,5,7),"y"], pos=c(3,1,1,1),
+       labels=c("MLR Digital", "MLR Hand", "SLR Hand", "SLR Digital"))
```



- Our MLR treatment effect is $\beta_1 = E(Y | T = H, Z) - E(Y | T = D, Z)$ and is estimated to be -0.00475 (about -0.005), which is indicated by the relatively small difference between sloped lines (solid digital, dashed hand)—consistent with zero (no effect) as our previous summary output and matched pairs t-test indicated—compared to the large difference

(=??? about 10 times larger) between horizontal lines depicting the difference when $Z=\text{Dean}$ is not fixed in SLR (solid digital, dashed hand).

- Thus, this fixing (covariate adjustment) in the MLR is similar in spirit to matching, but matching does not make an assumption about the form of the association of outcome with $Z=\text{Dean}$ ($E(Y | T = H, Z)$ and $E(Y | T = D, Z)$ are not assumed to be a linear function of Z as in the MLR), though the assumption does not appear to be a poor one, and the estimate of the mean difference between outcomes obtained by matching is comparable (-0.015969, in a previous chunk), especially considering the variability of the matched differences (depicted in the length of the vertical lines) which serve to make this difference not inconsistent with zero as we saw in the previous t-test.

(What about different slopes? What about quadratic terms in $Z=\text{Dean}$? Perhaps on a homework.)

- **So, how would randomization help in such an observational regression study?** (We're getting back to our remark about Confounding in Observational Regression Studies.)
- If we could randomly assign units (with their unobserved values of confounder, Z , attached for the ride) to different levels of T , then, on average, as we have said, there would be balance. Thus, on average, we would expect no difference of units (which have retained their Z values) between levels of T .
- In other words, $E(Z | T)$ would be the same constant value across T values, i.e., $E(Z | T) = \gamma_0$ for all values of T so that we must have $\gamma_1 = 0$ so that $\beta_1 = \beta_1^*$ —**no bias**. Intuitively, mixing up Z across levels of T would break the relationship with T , effectively setting correlation to zero (on average).

- Of course, again, there would appear to be no opportunity in such voting situations to randomly assign districts to values of voting technology T values in order to mix-up political outlook $Z = Dean!$
- So, if we are in an observational setting, and we cannot mix by randomizing (either we can't mix/randomize units within fixed/balanced levels of potential confounders or do the next best thing of unrestricted mixing/randomizing), then we can still try to emulate the ideal of the counterfactual and causal effect by matching or covariate adjustment when we do observe covariates, though there still may be other, unobserved covariates that are unmixed, thus we are still relatively far from justifying causality in an observational setting. (Again, the field of Causal Inference may have something to offer...)

5.7 Qualitative Support for Causation

Despite the limited scope of observational studies with regard to causality, see the list of qualitative rationale for causality in [Far14, §5.7]

Sampling

- **Another Random Mechanism.** We now return to [Far14, §3.4 Sampling], which we had skipped to cover that material here, instead. (I think of sampling as a kind of random mechanism that justifies use of random variables and probability, like a randomized experiment justifies use of random variables and probability.)
- **Scope of Inference** So far, in this chapter, we have discussed the **scope of inference** with regard to **causality**—if we randomized units to treatments, then we may infer causality, otherwise not—but we did not ask

how or if that inference applied just to the **units under study** or if somehow we could extend the scope of inference to include a **broader collection of units**, not just those in the study, experimental or observational.

- **Inductive Inference.** After all, such **generalization is fundamental to science** and to our ability to interact with the world around us?

Definition 5.7 (Population).

- **The complete set of units of interest.** We may denote the total number of units in a population (**population size**) with N , though we do not consider population size much in this class.
- **Ideally, the population is well defined.**
- **In most practical situations, this set of units is almost always beyond us, in some sense, so, instead, we focus on a subset of the population (sample).**
- **Most often (not always), we use a mathematical model as an approximation to the distribution of values from the actual population of N units.** In such cases, we sometimes refer to the mathematical approximation as the **superpopulation** because, for example, a normal distribution approximation represents an uncountable infinity of possible outcomes, which no real population has, as far as I know.

Remark 5.3 (Population of Units or Population of Values?). Note that some people (e.g., statisticians) may prefer to discuss sample and population in terms of the values of the (random) variables (outcomes) that may be observed

from units—a sample/population of values rather than sample/population of units. Context usually clarifies.

Definition 5.8 (Sample).

- A **subset of units (or their values) from a population** (*if well defined*). Data set. We often denote the total number of units/values in a sample (**sample size**) as n .

Definition 5.9 (Simple Random Sample).

- A *simple random sample of size n from a population* is a subset of the population consisting of n members selected in such a way that **every subset of size n is afforded the same chance of being selected**.
- For the methods that we will use in this class, we assume $n \ll N$ else we would have to make a “**finite population correction**.” A rule of thumb is n is less than 5% of N .

Remark 5.4 (We Will Not Cover Sampling Methods).

- **Scope of Inference.** Random sampling from a population is fundamental to extending the scope of inference from a sample to a population, analogous to how random allocation is fundamental to extending the scope of inference from mere association to a cause and effect relationship.

- **Etc.** *In this course, we do not treat the many different ways to obtain samples from populations, but largely leave this important and rich area to other courses.*

Sampling Distribution: Example

- Analogous to the randomization mechanism of randomized experiments and its associated reference (null) distribution (randomization distribution) arising from the notion of hypothetical replication of the mechanism, we also have the **mechanism of random sampling**, which also leads to the notion of **hypothetical replications** and a **reference distribution** with which to conduct statistical inference.
- For random sampling, we **consider (hypothetically) all possible random samples** (of whatever size we are considering, say, generically, n) from the same population.
- Then, we consider all corresponding values of a (test) **statistic**, e.g., sample average if, say, we want to infer about the population mean (like in “STAT 101”), or the difference of averages, if we want to infer about (sub-)population means, or t statistics or F statistics...
- A histogram of these replicated test statistics would represent the **sampling distribution** for that statistic.
- We could then use the **sampling distribution** to test hypotheses, etc., similar to our discussion of the **randomization distribution**, above.
- But, of course, if we could obtain **all possible samples** of size n from the population, then we would have access to the entire population, and we would have its mean and its variance and all other properties; there would be no need for statistics! Curb your enthusiasm!

- We obtained the randomization distribution by considering (an MC sample of) **all possible permutations** of units to treatments (or vice-versa). But, how do we proceed in the random sampling case to get the sampling distribution?
- In the random sampling case, we appeal to **(1) theory**. In particular, the **Central Limit Theorem (CLT)** tells us that, practically speaking, no matter what population we consider, as long as our (random) sample size, n , is “large enough”, then **sample averages are approximately normally distributed**. (Assume that sample averages are appropriate statistics/estimators, as if we want to estimate population means. And, assume $n \ll N$.) (TBD: Bootstrap)
- Or—this may sound familiar—we might **(2) assume** at the outset that our (super)population distribution is normal, which is an obvious approximation. (Why?) (We might offer plots, e.g., histograms, and summary statistics to help justify this assumption.)
- So, we appeal to **theory or assumptions** to get us to an **approximate sampling distribution**, as you likely did in a previous statistics course using **typical normal/ $\chi^2/t/F$ based procedures**, which we introduced in Chapter 3 ([Far14, Chap. 3]), with more details in Appendix B.
- As mentioned, if the normal (or related) approximations are poor, we may appeal to a **permutation distribution/test** ([Far14, §3.3]), in the case of observational data, which is what we (not your author) called a **randomization distribution/test** in the context of a randomized experiment ([Far14, §5.3]).

Example 5.2 (Illustrating a Sampling Distribution via CLT).

- *To illustrate, we generate $N = 1000$ values from a uniform distribution (**superpopulation**) to mimic the responses from a (finite) **population***

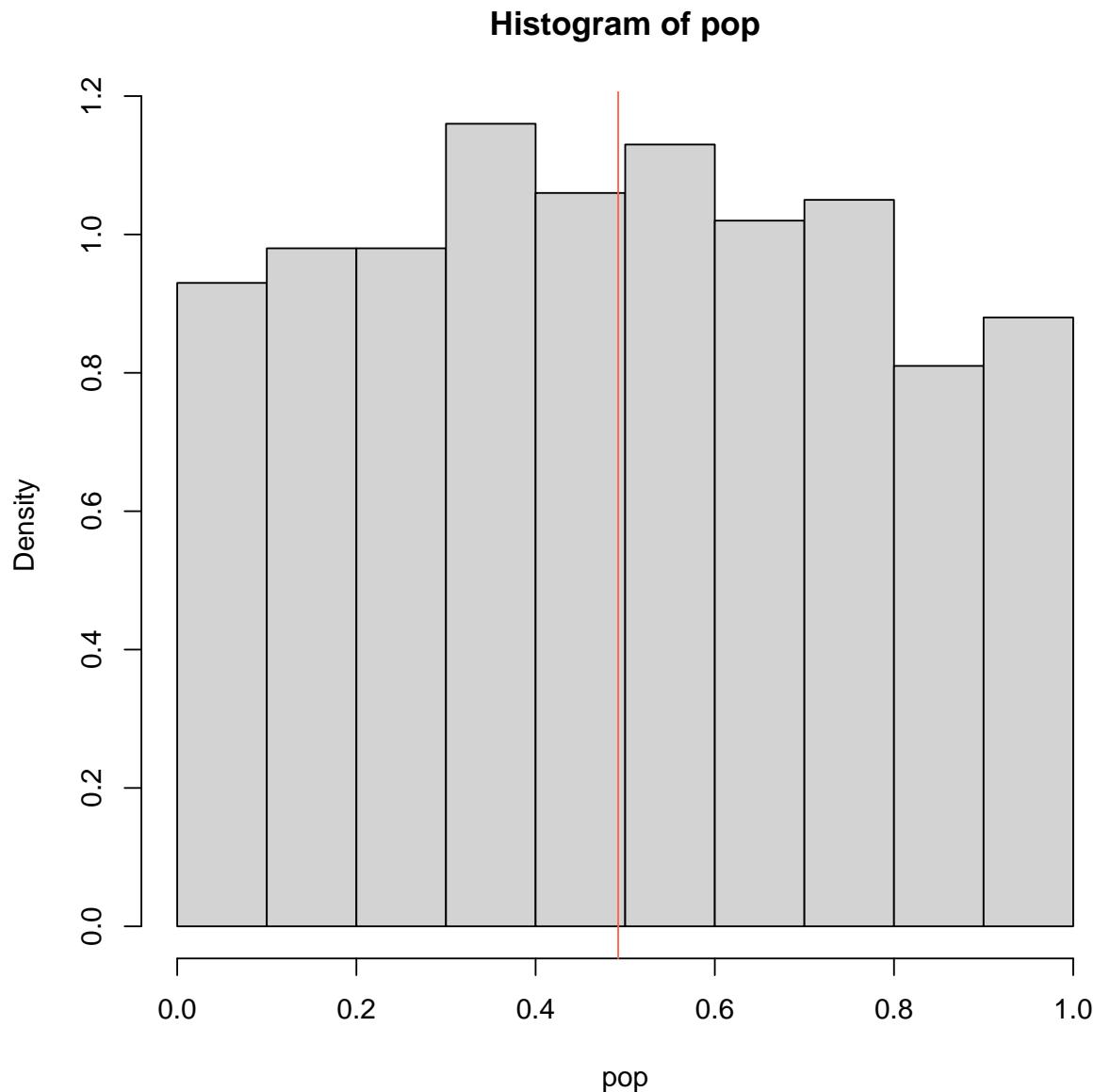
of N units.

- Further, assume we are interested in inferring the **mean of the population**, which we know in this example, to be $\mu = \sum_{i=1}^N Y_i/N = 0.49215$, where Y_i denotes the response of unit i in the population.
- We also compute the **population standard deviation** (below).
- Of course, we usually cannot compute the population mean/sd in practice.

```
> ## CLT
> N<- 1000 ## <-- Finite population size
> set.seed(20500 + 5150 + 24601) ## <-- for reproducibility
> pop<- runif(n=N,min=0,max=1) ## <-- Uniform superpopulation, mean 0.5
> (mu<- mean(pop)) ## <-- ``Actual'' pop. mean
[1] 0.49215

> (sigma<- sd(pop)) ## <-- ``Actual'' pop. sd
[1] 0.27847

> hist(pop, prob=TRUE)
> abline(v=mu, col="tomato")
```

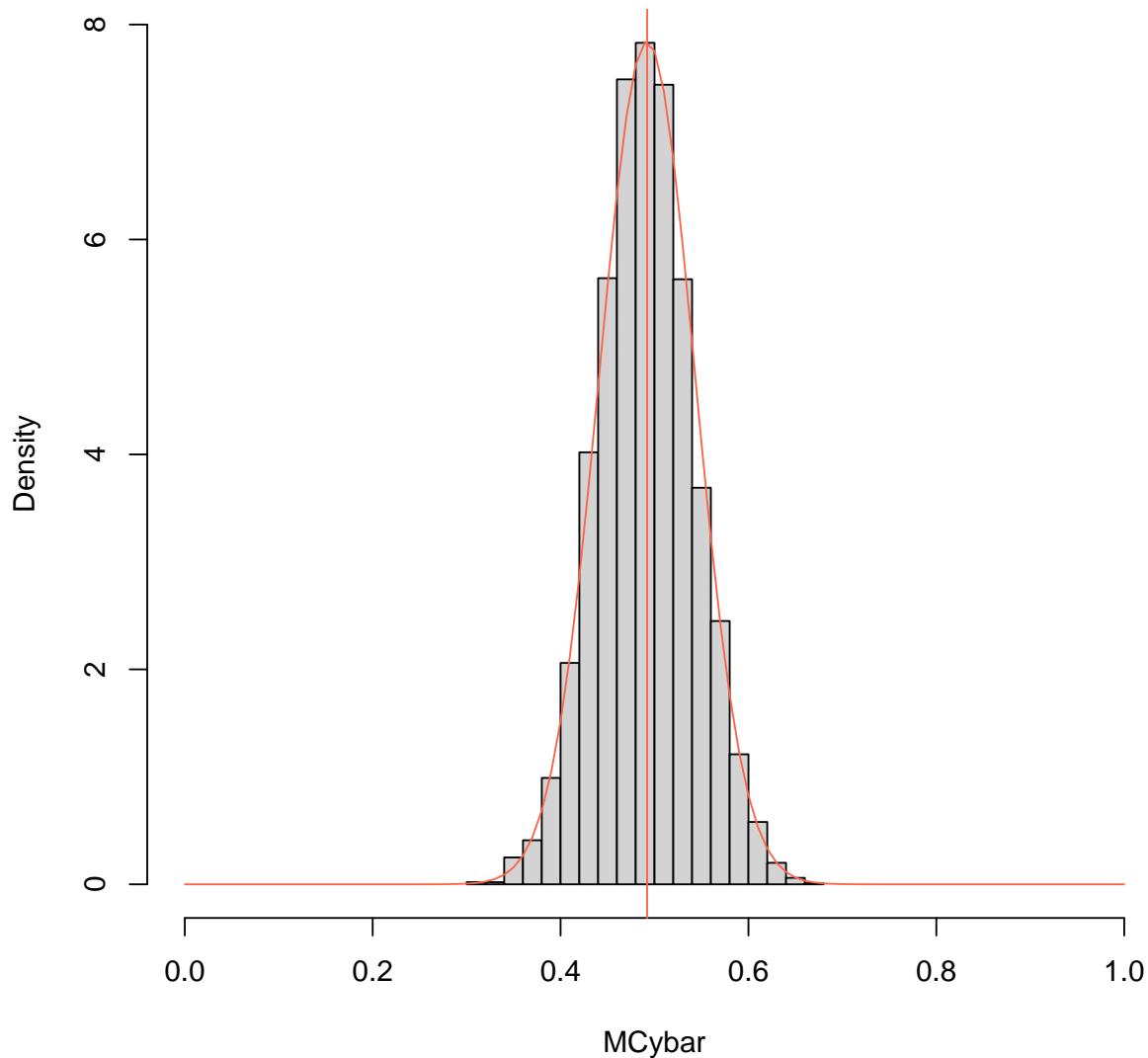


- (E.g., cont'd) Now, proceeding as in practice, without knowledge of the population, we mimic a simple random sample of size $n = 30$ and compute the average, \bar{Y} , as a **statistic**. Note we obey the 5% rule of thumb.

```
> n<- 30 ## sample size  
> oursampley<- sample(x=pop, size=n, replace=FALSE)  
> (ourybar<- mean(oursampley))  
  
[1] 0.50073
```

- (E.g. cont'd) Similar to the randomization distribution, the sampling distribution is the distribution of a statistic—here, \bar{Y} —computed from each of the possible ways to choose $n = 30$ units from $N = 1000$, a big number of ways; see code below. (And, as we just said, above, we cannot do this in practice, and, if we could, there would be little need for statistics in this regard.)
- Instead, we obtain a Monte Carlo (MC) sample of, size, say, $M = 5000$, from all possible random samples of size n (a sample of samples), compute the average for each of the M samples, then display the resulting histogram of M averages as an approximation to the (practically and computationally) unobtainable sampling distribution.
- In practice, as we said, we appeal to theory/assumptions. In particular, we appeal to the **Central Limit Theorem** in this example, which says, loosely, that sample means (or sums) are approximately normal (despite the fact that the population distribution, as depicted by a histogram, does not appear normal). We will compare this normal approximation to the (MC approximate) sampling distribution.
- What do we notice? Does theory seem to be operating here? You may recall the CLT tells us that $\bar{Y} \sim N(\mu, \sigma^2/n)$, at least approximately.
- Certainly, assuming Y_i to be normal is not tenable (the sample histogram would look like the population histogram, which looks uniform, not normal).

```
> ## How many possible random samples?  
> choose(N,n) ## <-- far too many to enumerate  
  
[1] 2.4296e+57  
  
> ## So, we obtain M MC samples of samples of size n as approximation.  
> M<- 5000  
> MCybar<- vector("numeric",length=0)  
> for (i in 1:M) {  
+   MCsampley<- sample(x=pop,size=n, replace=FALSE)  
+   MCybar<- c(MCybar, mean(MCsampley))  
+ }  
> hist(MCybar, prob=TRUE, xlim=c(0,1))  
>  
> ## Compare to CLT (mu and sigma obtained previously)  
> curve(dnorm(x,mean=mu, sd=sigma/sqrt(n)), from=0, to=1,  
+         add=TRUE, col="tomato")  
> abline(v=mu, col="tomato")
```

Histogram of MCybar

- (E.g. cont'd) Of course, we used the population mean and standard deviation, μ and σ , to illustrate that the CLT is working as an approximation to the sampling distribution of \bar{Y} .
- In practice, again, we don't know μ and σ , but, still, the CLT says \bar{Y} is approximately normal. In this case, we play a similar game to that used

to get our randomization distribution, and we assume that we know μ , which we formulate in a **null hypothesis**,

$$H_0 : \mu = 0.4,$$

where we choose 0.4 merely for illustration.

- Again, similar to our randomization distribution discussion, the idea is that, if our actual, observed statistic, \bar{Y} is large/small (formalized via a p-value) relative to those values in the null distribution, then this may be due to chance. Did we by chance sample extreme units in the population? Did we not get a good **mix** from the population? Or, perhaps our null assumption about μ is wrong.
- Still, in our current example, we don't know the population standard deviation, σ . Sparing you the remaining details (Chapter 3), we end up using the **Student's t distribution** (with $n-1 = 29$ degrees of freedom) as the (approximate) sampling distribution of the statistic,

$$\frac{\bar{Y} - 0.4}{\frac{s}{\sqrt{n}}},$$

for which our observed value is 1.9807, etc.

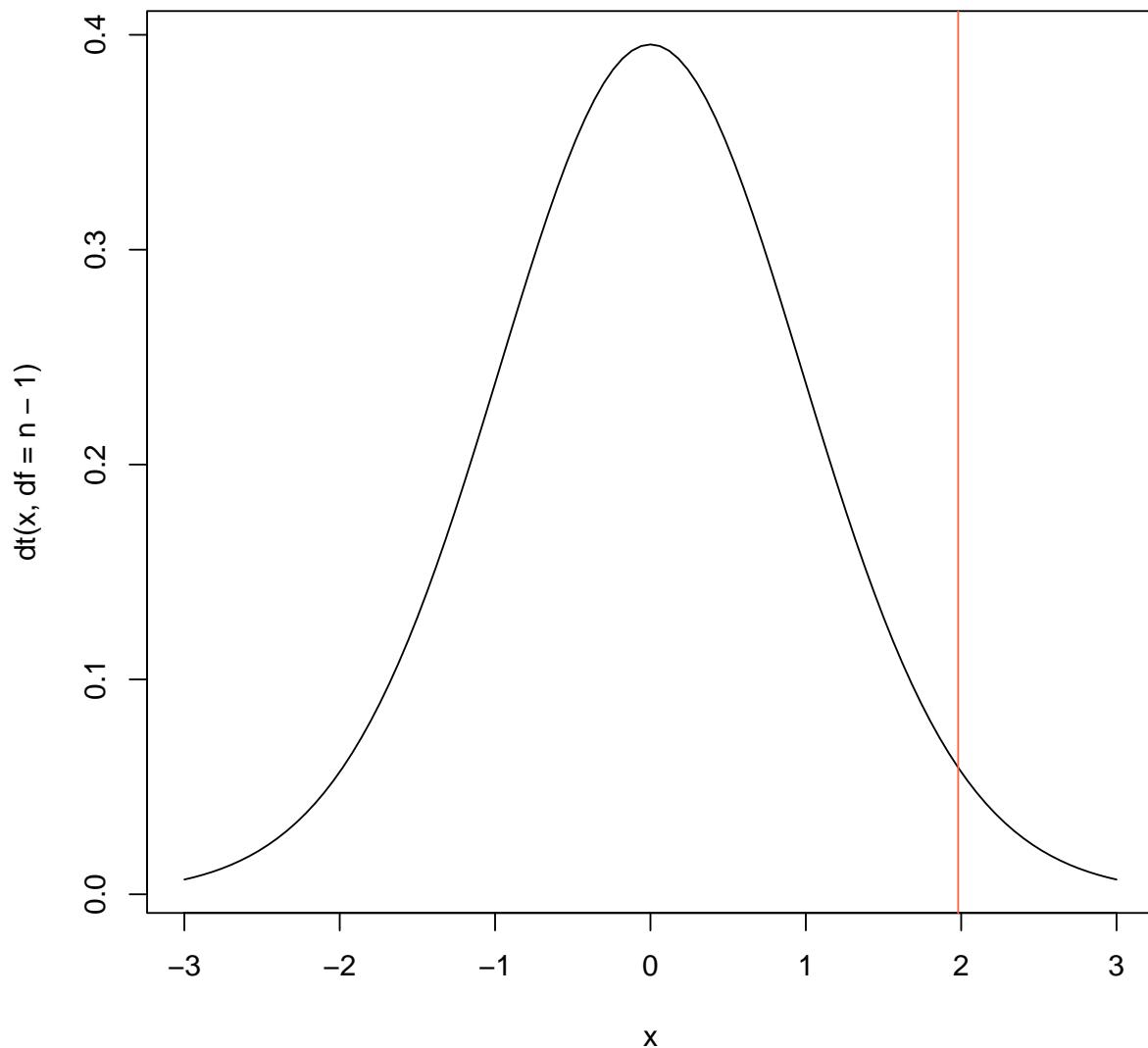
```
> mu0<- 0.4
> (s<- sd(oursampley))

[1] 0.27854

> (ourtstat<- (ourybar - mu0) / (s / sqrt(n)))

[1] 1.9807

> curve(dt(x,df=n-1), from=-3, to=3)
> abline(v=ourtstat, col="tomato")
```



```
> (pval<- 2 * (1 - pt(ourtstat, df=n-1)))
```

```
[1] 0.057182
```

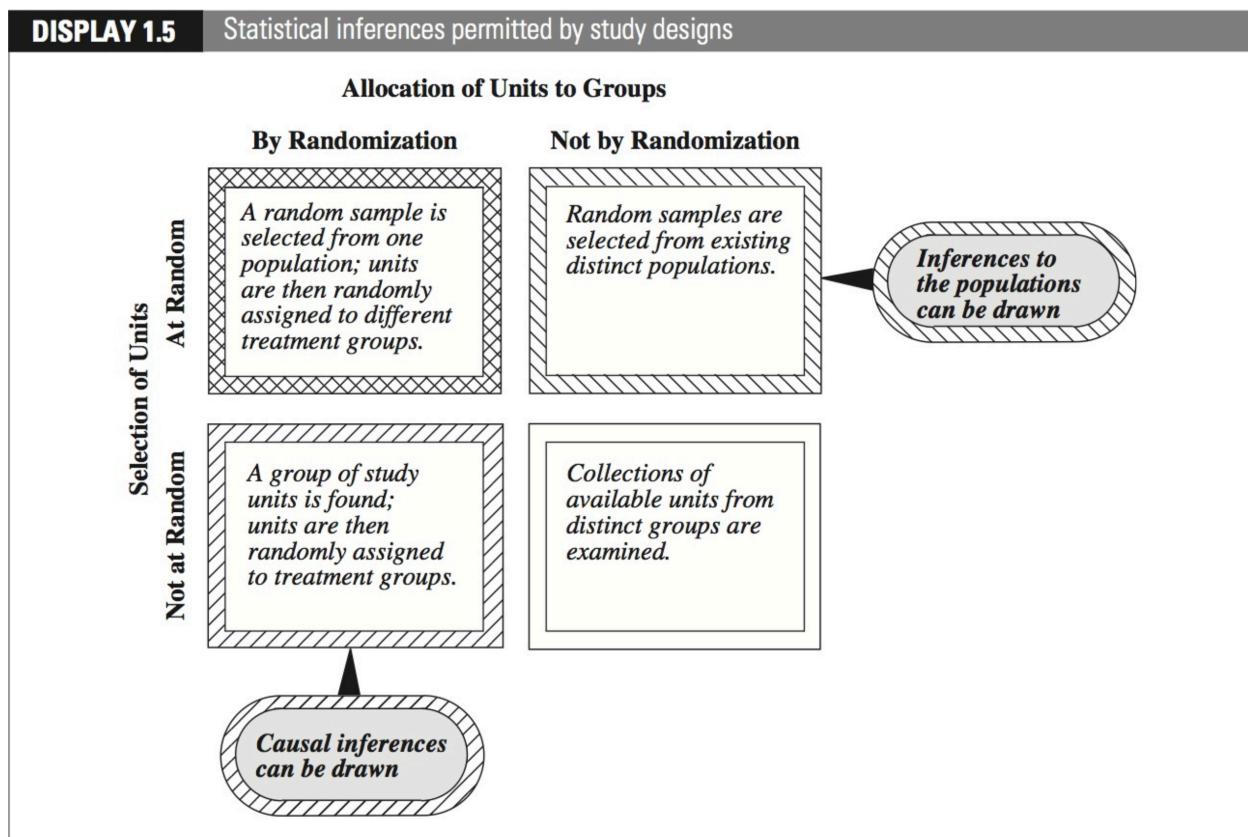


Figure 5.1: Scope of inference (Source: [RS13]).

Scope of Inference: Summary

Though Chapter 3 was entitled ‘Inference,’ we now, after having discussed randomization and random sampling, discuss the scope of inference obtained from a study. Figure 5.1 provides a concise summary.

- **Randomization Permits Causal Inference.** Randomization mixes units to balance units, on average, with respect to potential confounding variables so that (on average!) effects of the treatment of interest are not biased by confounders. Thus, our scope of inference is permitted to

extend from one of vague association to a causal effect of treatment on response.

- **Randomization Distribution Allows Calibration of Result via p-value Under Null.** Of course, we typically observe only one randomization and, by chance, we may observe a particularly unbalanced randomization that gives an extreme result. But, under the assumption of no treatment effect, the randomization distribution, via the random mechanism of randomly assigning units to treatment (or vice versa), allows us to compute the probability of observing such extreme results (again, assuming no treatment effects (null is true)), via the p-value (technically, the probability of observing a result at least as extreme as the one we computed for our particular data, assuming null true). We may choose significance level, α , and we may conclude that a p-value this small or smaller, while possible under the null, suggests that our null assumption is incorrect and that there exists a treatment effect.

Of course, our conclusion may be wrong, i.e., we may have committed a type I error, but the randomization distribution (or model approximation thereof (e.g., normal, χ^2 , t , F)) has allowed us to control that probability to be small, α .

Again, if we have not randomized (and we have not assumed some sort of causal model), then our scope of inference is limited to association, not causality.

- **Random Sampling Permits Inference to Population.** Similarly, if our units have been selected randomly from some larger population, then the random selection mechanism again has a mixing action or average balancing action so that, on average, the characteristics of the sample units reflect characteristics of the population units (e.g., mean, etc.). Thus, our scope of inference is permitted to extend from our sample units to the population of units.
- **Sampling Distribution Allows Calibration of Result via p-value Under Null.** Of course, again, we only have one random sample and,

by chance, we may obtain a sample with characteristics that are somehow extreme relative to those of the population of units. But, random sampling allows us to compute the probability of observing such extreme samples via the p-value (again, assuming a null hypothesis is true). And, again, we may compare the p-value to α to conclude some null hypothesis about the population is false, and, again, we may be wrong (type I error), perhaps because we obtained a particularly extreme sample by chance even though the null is true, but we have controlled the probability of this (type I) error to me small, α . If we do not have a random sample, then our scope of inference is limited to the sample units at hand, though we may be compelled to pursue further random sampling studies or randomized experiments.

- **Best of Both Worlds.** If we have randomly sampled units that have been randomly assigned to treatments, then we are permitted to infer a causal effect of treatment on response for units in the population.
- **See Nearby Diagram for Succinct Summary**

Lecture 6

Diagnostics

Contents

Why check your model?	229
Things to Check	230
6.1 Checking Error Assumptions	231
6.1.1 Constant Variance	232
Prototypical Plots	232
Savings Data Example	233
Typical Fan Pattern	235
Calibrate Your Eye	235
Tests of Non-constant Variance	241
Brown–Forsythe Test	241
Savings Data Example	242
Breusch–Pagan Test	244
Savings Data Example	245
Transformations of Outcomes (Y)	246
A Juggling Act	249
Box–Cox Procedure	249
Galapagos Data Example	250
Remedial Actions for Non-Constant Variance	252
6.1.2 Normality	253
Normal Probability Plot	253
Savings Data Example	255
Calibrate Your Eye	256
Correlation Tests of Normality	258

Test 1	258
Savings Data Example	259
Test 2: Shapiro-Wilk	260
Savings Data Example	261
Remedial Actions for Non-Normality	261
6.1.3 Correlated Errors	261
Global Warming Data Example	262
Durbin-Watson Test for Correlated Errors	264
Remedial Actions for Correlated Errors	265
6.2 Finding Unusual Observations (Outliers)	265
6.2.1 Leverage	266
Outlying X : Hat Matrix Leverage Values	266
Savings Data Example	268
6.2.2 Outliers	272
Outlying Y : Studentized Deleted Residuals	272
Side note: Bonferroni family-wise error rate	281
Remedial Actions for Outliers	284
6.2.3 Influential Observations	285
Influence of Case i on a Single Fitted Value: DFFITS	285
Cook's Distance	286
Influence of Case i on Regression Coefficients: DFBETAS	288
Savings Data Example	289
Default Diagnostic Plots	293
Influence on Inferences of Interest	295
6.3 Checking the Systematic Structure of the Model	295
Added-Variable Plots aka Partial Regression Plots	295
Savings Data Example	296
Partial Residual Plot	298
6.4 Discussion	303

Why check your model?

- **Good Goodness.** If our normal linear model likelihood is correct, then we get optimality in the form of BLUEs (§2.8; and BLUPs): correct and shortest confidence intervals, correct and most powerful tests, and correct p-values (but, no torturing our data).
- **Not So Bad Goodness.** We may still get good and accurate results when parts of our full likelihood fail to agree with our data, which seems more practical as assumptions of normality for observed data (or errors) seem unrealistic.
- **Goodness from Correct Mean Model.** Of particular importance is our **regression (mean) model**, $\mathbf{X}\beta$ (not the full normal linear model likelihood).
 - **Consistency.** Under broad conditions, if we merely get our **mean model correct**, then we get consistency: loosely speaking, our least-squares estimators of regression function parameters become arbitrarily close to their corresponding unknown regression function parameter targets, β , with arbitrarily high probability as sample size, n , gets large.
 - **(Prediction).** (While consistency seems fundamental to inferring about regression model parameters or functions thereof, some may argue that it's not crucial if prediction is the goal. INF 504 focuses more on prediction and less on modeling the mean and variance.)
 - **Variance Model.** Sometimes, consistency depends also on our variance model being correct, but not in INF 511 where our variance model is very simple (constant variance); more in INF 512.
- **Goodness from Correct (Co)variance Model.** In addition to consistency, if our (co)variance model is also correct (again not requiring that a full likelihood (e.g., normal) is correct), we get **asymptotically correct standard errors and asymptotic normality, which give**

asymptotically correct confidence intervals, test error rates, and p-values (but, no torturing our data). So, to reiterate, we need only to get the mean and (co)variance model to be correct for such goodness. (Sometimes, we need to get higher moments correct, beyond the mean and variance, to get goodness, but not in INF 511; again, more in INF 512).

- **Check Model to Justify Claims of Goodness.** Of course, in practice, we can typically never be certain that we are correct, but presenting a **good diagnostic analysis is necessary** to justify claims of correct, good or optimal inferences.

Things to Check

- **Errors.** $\epsilon \sim N(0, \sigma^2 I)$
 - **Constant Variance.** $\text{Var}(\epsilon_i) = \sigma^2$
 - **Normal**
 - **Uncorrelated (independent)**
- **Outliers.** Do we have **unusual observations**, Y , or X or both? That is, do we have outliers?
- **Influential Observations.** Are any observations somehow especially influential in determining some aspect of the fitted model?
- **Mean Model.** Is our **regression model** correct, i.e., is $E(Y | X) = X\beta$? Or, is there somehow **lack of fit**?

6.1 Checking Error Assumptions

- **Residuals (or some function thereof) are the primary tool for diagnosing model assumptions.**
- Recall the **residual vector** (§2.3 & 2.4)

$$\begin{aligned}
 \hat{\epsilon} &= y - \mathbf{X}\hat{\beta} \\
 &= y - \hat{y} \\
 &= y - \mathbf{H}y \\
 &= (\mathbf{I} - \mathbf{H})y \\
 &= (\mathbf{I} - \mathbf{H})\mathbf{X}\beta + (\mathbf{I} - \mathbf{H})\epsilon \\
 &= (\mathbf{I} - \mathbf{H})\epsilon
 \end{aligned}$$

- Thus (a bit of detail omitted),

$$\text{Var}(\hat{\epsilon}) = \sigma^2(\mathbf{I} - \mathbf{H}).$$

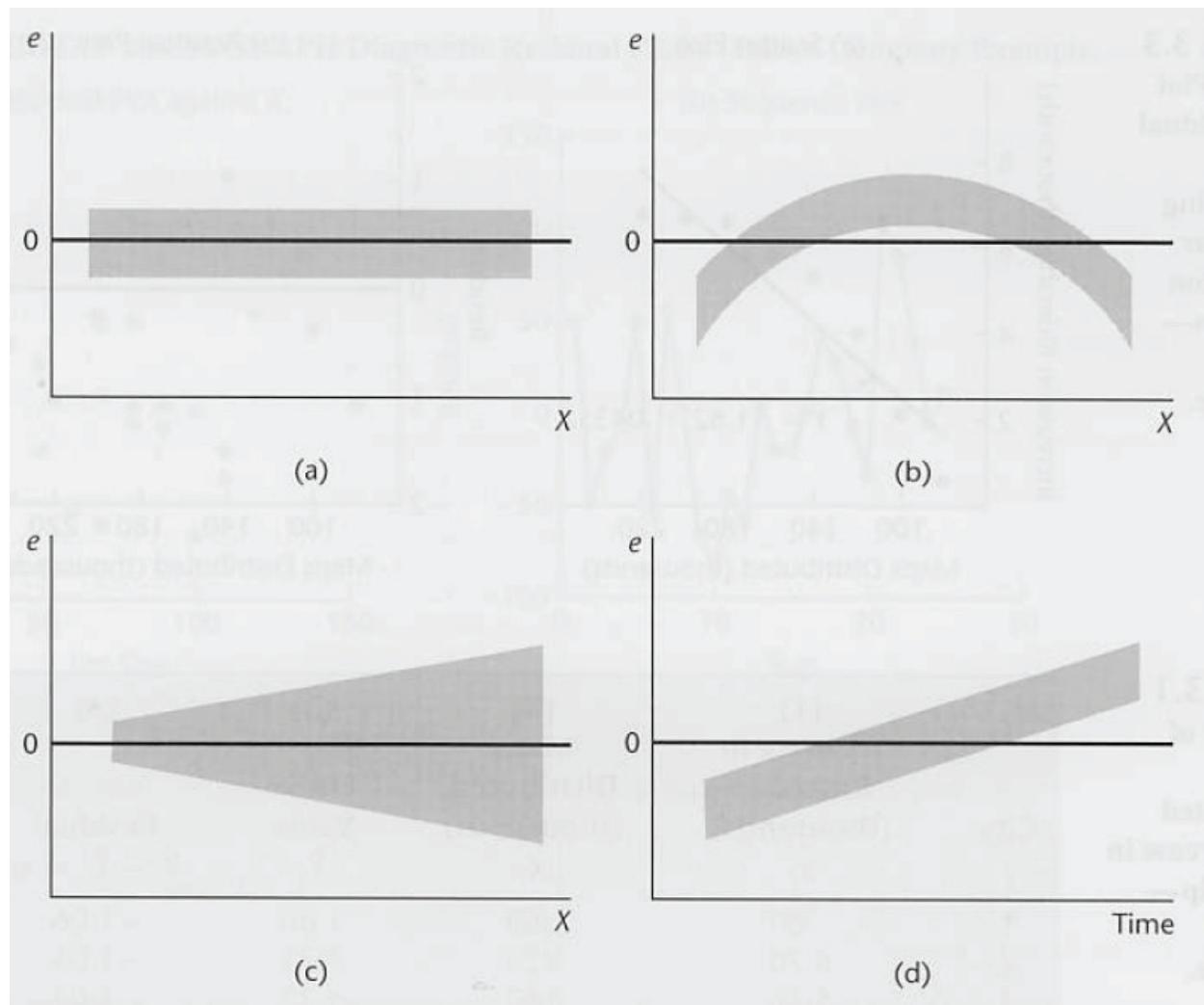
- **So, residuals do not have constant variance**, even if the errors do. Are residuals useful for assessing departures from constant variance?
- **And, residuals are not uncorrelated**, thus cannot be independent, even if the errors are. Are residuals useful for assessing departures from uncorrelation/independence?
- **Always sum to zero** (if we have an intercept), even if $E(\mathbf{Y} | \mathbf{X}) \neq \mathbf{X}\beta$. That is, their **average is zero**, even when the model is wrong. Are they helpful for assessing $E(\epsilon) = 0$, i.e., for assessing $E(\mathbf{Y} - \mathbf{X}\beta) = 0$, i.e., for assessing $E(\mathbf{Y} | \mathbf{X}) = \mathbf{X}\beta$?
- **(Non-)Normal.** Residuals are/are not normal if the errors are/are not, so it seems that residuals may be useful for assessing normality of errors.
- **Loosely, Observed Errors.** Despite these properties, we may think loosely of residuals as sorts of “observed errors” to help use diagnose departures from model assumptions.

6.1.1 Constant Variance

- **Homoskedasticity** means constant variance.
- **Heteroskedasticity** means non-constant variance.

Prototypical Plots

- Your textbook's author presents prototypical plots ([Far14, Fig. 6.1]) illustrating
 - (i) no problem,
 - (ii) heteroskedasticity and
 - (iii) lack of fit (regression model missing a non-linear component).
- Figure [KNNL05, Fig. 3.4 (c)], reproduced below, also illustrates a **prototypical fanning pattern** of residuals indicating non-constant error variance. The other plots in the figure illustrate the wider use of residuals for indicating (a) no departure from assumptions, (b) omitted variable X or missing curvilinear specification of existing X , (d) residual time trend (lack of fit of the regression model to the regression function due to omission of time as a covariate in the regression model).



Savings Data Example

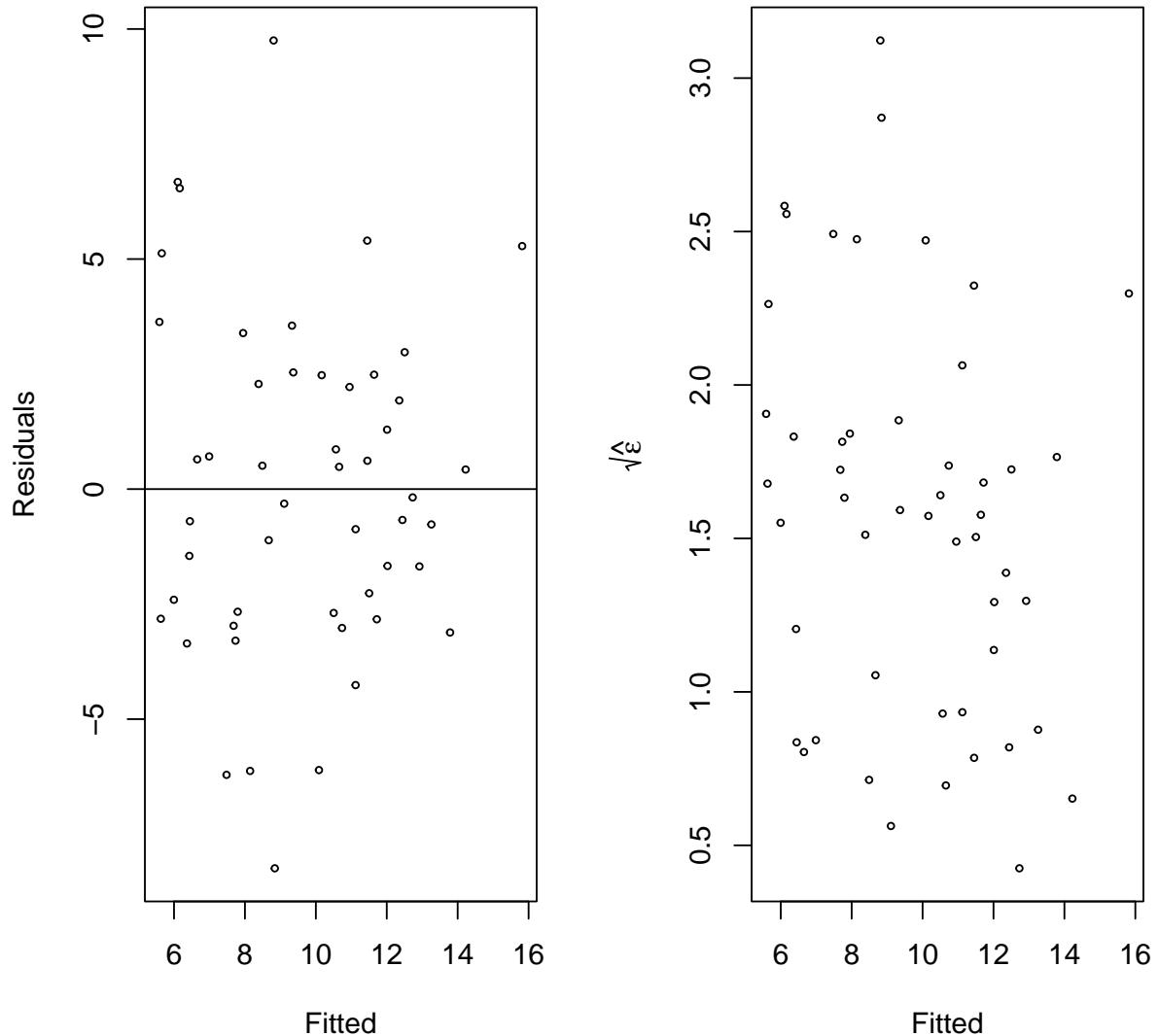
- We illustrate with some of your author's R code/output, below, which does not seem to indicate much of a problem ([Far14, Fig. 6.2]).

```
> library(faraway)
> data(savings, package="faraway")
> lmod <- lm(sr ~ pop15+pop75+dpi+ddpi, savings)
```

```

> par(mfrow=c(1,2))
> ## Classic
> plot(fitted(lmod),residuals(lmod),xlab="Fitted",
+       ylab="Residuals", cex=0.5)
> abline(h=0)
> ## Sqrt to reduced skewness, possibly clarifying the question constant
> ## variance.
> plot(fitted(lmod),sqrt(abs(residuals(lmod))), xlab="Fitted",
+       ylab=expression(sqrt(hat(epsilon))), cex=0.5)

```



```
> par(mfrow=c(1,1))
```

- What do you think?

Typical Fan Pattern

- The above figures use the fitted value, \hat{y}_i , on the horizontal axis.
- We often see an **increasing fanning pattern with increasing (mean) regression function** $E(Y | x)$ as estimated by the fitted values.
- \hat{y}_i could be from an ANOVA model (a linear model), too, and the figures still apply. We hope to get to ANOVA in INF 511.
- In the case of SLR (as opposed to MLR), it doesn't matter (qualitatively) whether we use fitted values or covariate values for assessing non-constant variance.

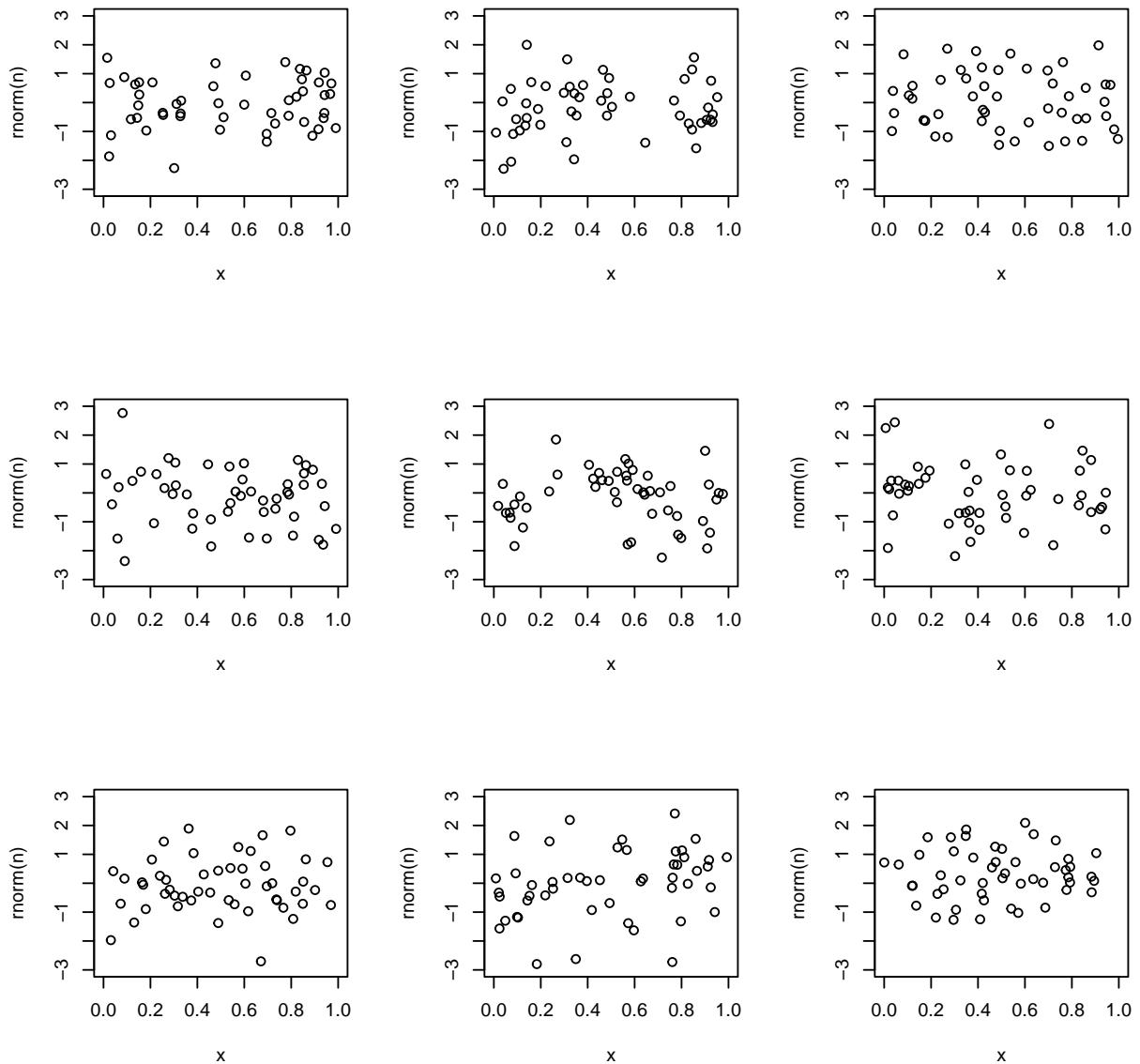
Calibrate Your Eye

- Are you unsure about what you're seeing? Is it non-constant variance? Are you seeing a pattern that does not exist? Are you an apopheniac?
- Your author provides some code/plots to help calibrate/train your eye; reproduced below.
 1. Constant variance
 2. Strong non-constant variance, $\text{Var}(\epsilon | x) = \sigma^2 x^2$ ($\sigma^2 = 1$)

3. Mild non-constant variance, $\text{Var}(\epsilon | x) = \sigma^2 x,$
4. Lack of fit (missing a function of x in our regression function or regression function is not linear). (The code, below, is taken from your author; at face value, I don't understand what he's trying to get across here (aside from the fact that he appears to intend to demonstrate lack of fit). I'll try to explain in class.)

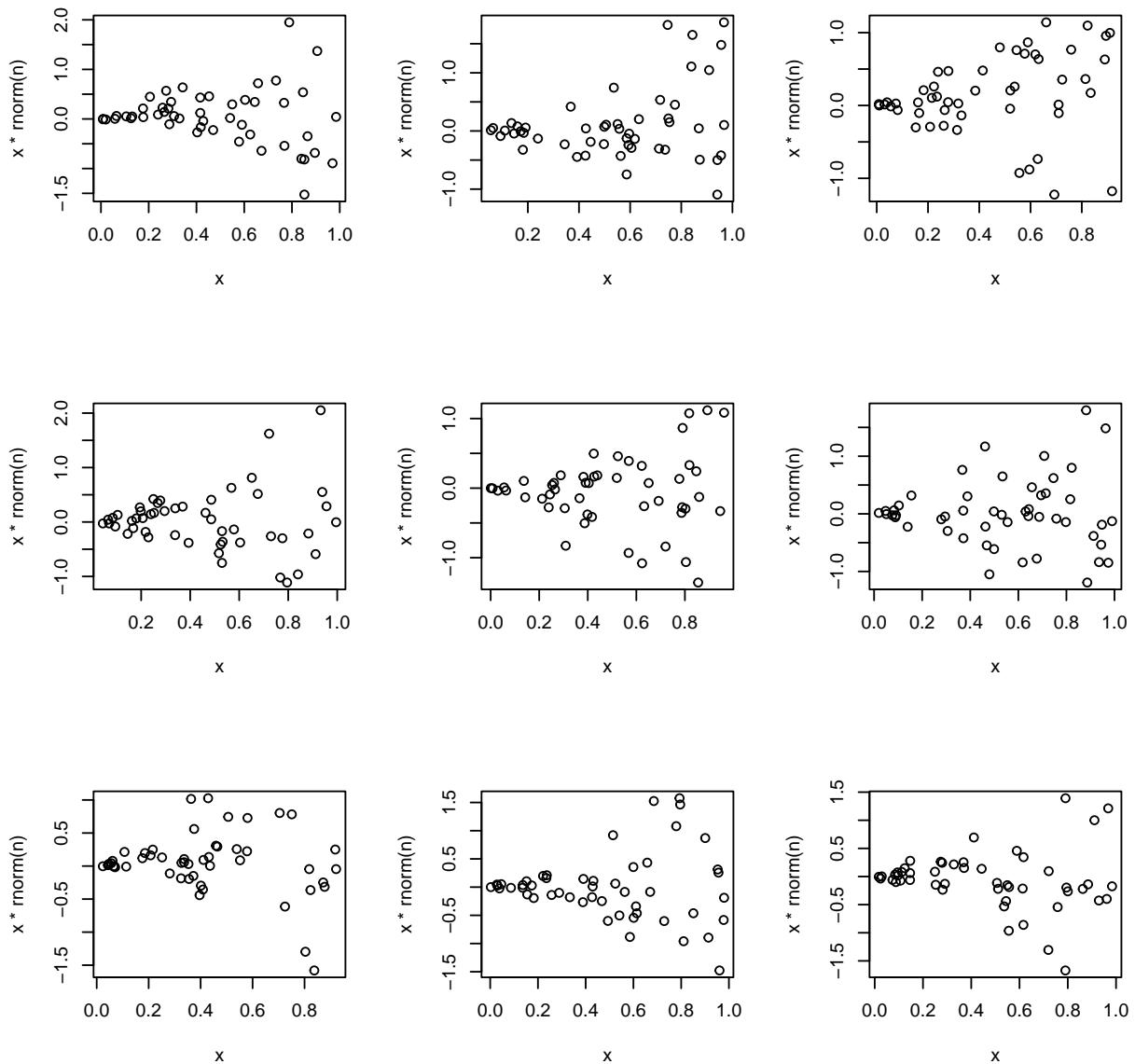
```
> ## Calibrate/train your eye:  
> par(mfrow=c(3,3))  
> n <- 50  
> for(i in 1:9) {x <- runif(n) ## (1)  
+   plot(x,rnorm(n),xlim=c(0,1),ylim=c(-3,3))}  
> mtext(text="(1) Constant variance",outer=TRUE, line=-2)
```

(1) Constant variance



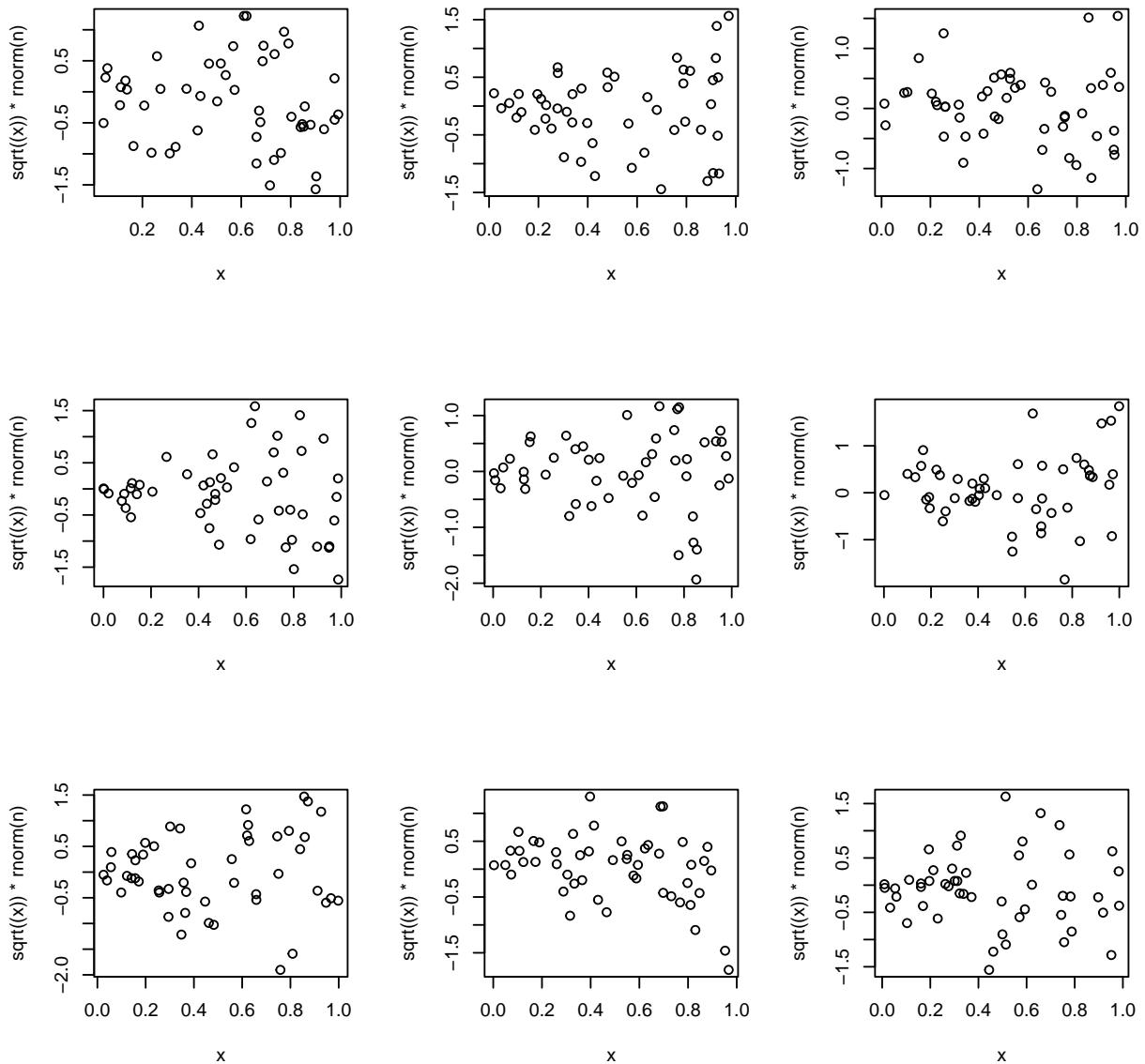
```
> for(i in 1:9) {x <- runif(n) ## (2)
+   plot(x,x*rnorm(n))}
> mtext(text="(2) Strong non-constant variance",outer=TRUE, line=-2)
```

(2) Strong non-constant variance

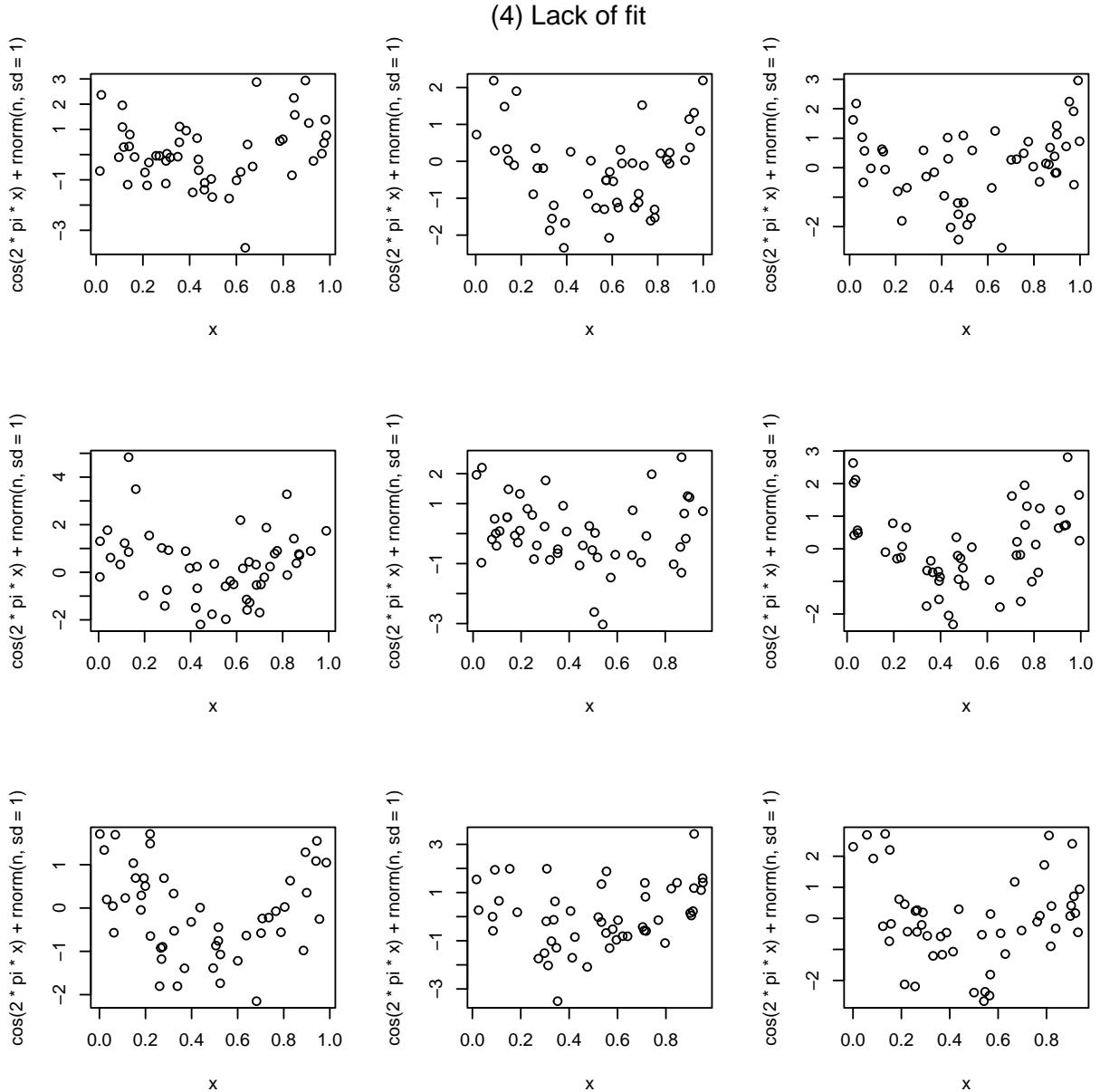


```
> for(i in 1:9) {x <- runif(n) ## (3)
+   plot(x,sqrt((x))*rnorm(n))}
> mtext(text="(3) Mild non-constant variance",outer=TRUE, line=-2)
```

(3) Mild non-constant variance



```
> for(i in 1:9) {x <- runif(n) ## (4)
+   plot(x, cos(2*pi*x)+rnorm(n, sd=1))} ## changed according to errata
> mtext(text="(4) Lack of fit", outer=TRUE, line=-2)
```



```
> par(mfrow=c(1, 1))
```

- In addition to the relatively subjective (but useful), graphical diagnostics, above, we have more formal tests of non-constant variance.

Tests of Non-constant Variance

- See [KNNL05, Sec. 3.6] for the Brown-Forsythe (BF) test and Breusch-Pagan (BP) test, which we cover only briefly, below.

Brown–Forsythe Test

- **Brown–Forsythe (BF) Test (Modified Levene Test):** Essentially, perform a one-way ANOVA overall F-test using “observations” consisting of absolute deviations of residuals from group medians for **some grouping** of residuals.

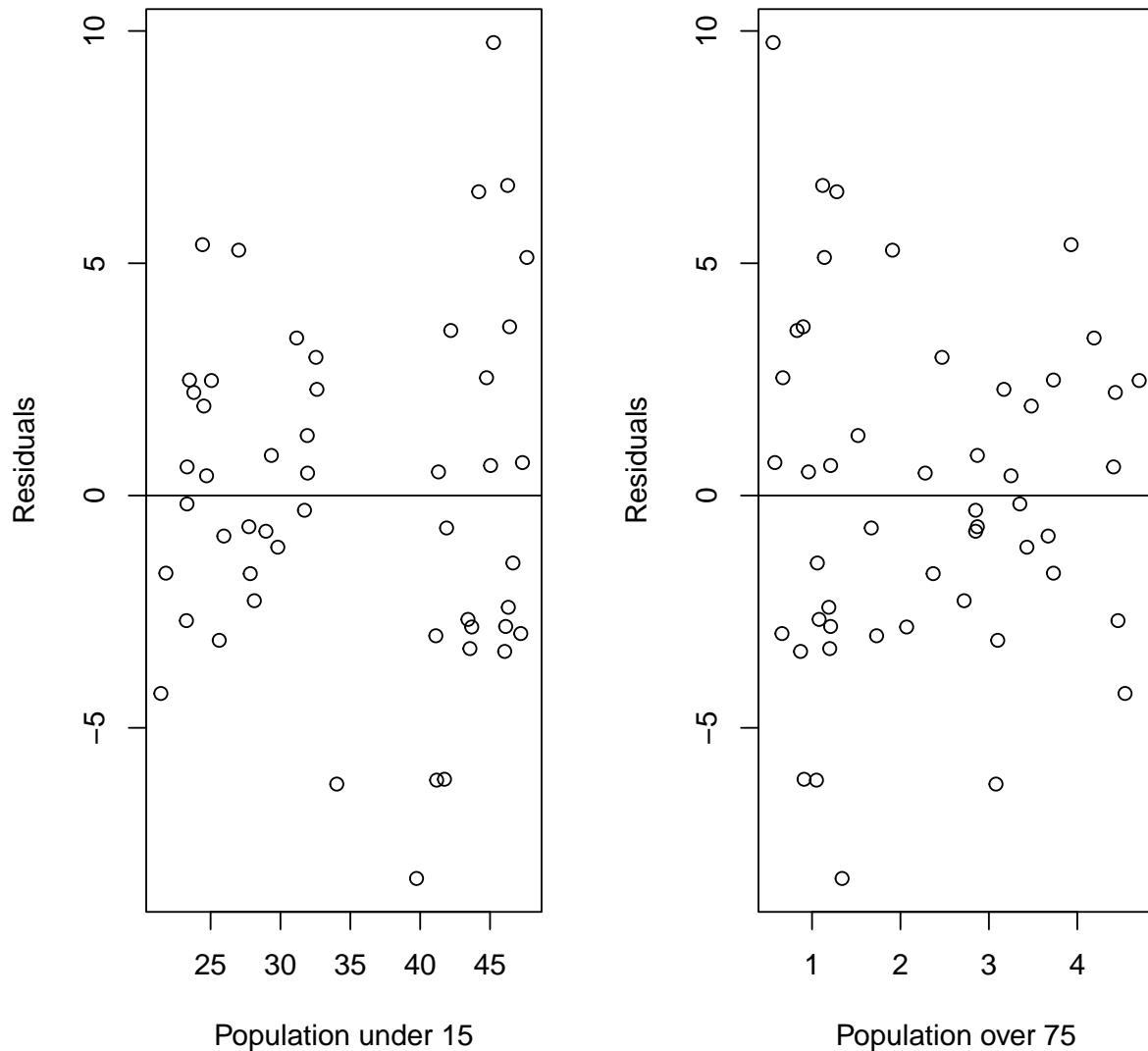
H_0 : variance of residuals ϵ_i is constant across i

H_a : variance of residuals ϵ_i is not constant across i

- We will get to one-way ANOVA near the end of the course. For now, just be aware of the BF test for non-constant variance.
 - The residual group sizes, n_j , $j = 1, \dots, J$ groups, should not be “too small”.
 - The BF test assumes independent absolute deviations $d_{ij} = |\hat{\epsilon}_{ij} - \tilde{\epsilon}_{ij}|$, which is not true, but is approximately true if n is not “too small.”
 - The BF test is **robust** against departures from normal ϵ_i .
- Your author performs a test similar in spirit to the BF test using `stats::var.test`, with two groups suggested by one of the predictor variables in the savings data example. It’s a bit different from the BF test. In particular, it’s not robust against departures from normality.

Savings Data Example

```
> ## Residuals v. a predictor
> par(mfrow=c(1,2))
> plot(savings$pop15,residuals(lmod),
+       xlab="Population under 15",ylab="Residuals")
> abline(h=0)
> plot(savings$pop75,residuals(lmod),
+       xlab="Population over 75",ylab="Residuals")
> abline(h=0)
```



```
> ## BF test is similar in spirit to F test on p. 77 LMwR2e:
> var.test(residuals(lmod)[savings$pop15 > 35],
+           residuals(lmod)[savings$pop15 < 35])
```

F test to compare two variances

```
data: residuals(lmod)[savings$pop15 > 35] and residuals(lmod)[savings$pop15 < 35]
```

```
F = 2.79, num df = 22, denom df = 26, p-value = 0.014
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.2410 6.4302
sample estimates:
ratio of variances
 2.7851
```

```
> ### Let's define our own Brown-Forsythe Test function.
> ### See also the levene.test() function in the car package.
> bf.test<- function(x,groups)
+ {
+   medians <- sapply(split(x,groups), median, na.rm = TRUE)
+   dij <- abs(x - rep(medians, table(groups)))
+   anova(lm(dij ~ groups))
+ }
> bf.test(residuals(lmod), groups=savings$pop15>35)
```

Analysis of Variance Table

Response: dij

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
groups	1	25.1	25.13	4.49	0.039 *
Residuals	48	268.5	5.59		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ## p-value comparable to that on p. 77 LMwR2e

Breusch–Pagan Test

- **Breusch-Pagan (BP) (aka Cook-Weisberg) test.** Essentially, regress the natural log of the squared residuals against one or more predictors and perform an overall F-test for linear association.
 - BP tests for a particular type of non-constant variance: e.g., $\ln(\sigma_i^2) = \gamma_0 + \gamma_1 X_i$, as might be suggested in the case of pop15

in the savings data example, hence our hypotheses are more specific (more structured) than BF: e.g., $H_0 : \gamma_1 = 0$ $H_a : \gamma_1 \neq 0$

- **lmtest::bptest** can model the variance more generally than presented here. For the running savings data example, by default, **lmtest::bptest** tests against $\ln(\sigma_i^2) = \gamma_0 + \gamma_1 X_{1i} + \gamma_2 X_{2i} + \gamma_3 X_{3i} + \gamma_4 X_{4i}$, where X_1 is pop15, X_2 is pop25, X_3 is dpi, X_4 is ddpi.
- No grouping required (BF requires groups), economizes on degrees of freedom, but requires specification of linear model of (log) of the variance.
- BP assumes “large” sample size n .
- BP assumes ϵ_i are independent and normally distributed.
- We’ll ignore the potential remedial measure that’s beating us over the head here: model the variance as suggested by the BP test. INF 512.
- See also **car::ncvTest**.

Savings Data Example

```
> ## The default value for argument varformula = RHS of
> formula(lmod)

sr ~ pop15 + pop75 + dpi + ddpi

> lmtest::bptest(lmod, studentize=TRUE)

studentized Breusch-Pagan test

data: lmod
BP = 4.99, df = 4, p-value = 0.29

> ## BP is similar in spirit to p. 75 LMwR2e (not studentized):
> summary(lm(sqrt(abs(residuals(lmod))) ~ fitted(lmod)))
```

```

Call:
lm(formula = sqrt(abs(residuals(lmod))) ~ fitted(lmod))

Residuals:
    Min      1Q  Median      3Q     Max 
-1.0404 -0.5584  0.0189  0.3213  1.5011 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.1622     0.3479   6.22  1.2e-07 ***
fitted(lmod) -0.0614     0.0348  -1.77   0.084 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.634 on 48 degrees of freedom
Multiple R-squared:  0.061, Adjusted R-squared:  0.0414 
F-statistic: 3.12 on 1 and 48 DF,  p-value: 0.0838

> ## How about something akin to variance function of pop15, like previous
> ## BF test...
> lmtest::bptest(lmod, varformula= ~ pop15, data=savings)

studentized Breusch-Pagan test

data: lmod
BP = 4.46, df = 1, p-value = 0.035

> ## ...gives similar p-value, too.
>
> ## See also the car::ncvTest.

```

Transformations of Outcomes (Y)

We may consider the transformation of Y to achieve approximate homoskedasticity to be a **remedial measure**; remodeling.

- A **variance stabilizing transformation** is a traditional remedial measure to address non-constant variance.

- We seek a transformation, $h(y)$, of the response, y , so that $\text{Var}(h(Y))$ is approximately constant.
- A **Taylor series expansion** of $h(y)$ about $E(Y)$ gives

$$\begin{aligned} h(y) &= h(E(Y)) + h'(E(Y))(y - (E(Y))) + \dots \\ \text{Var}(h(Y)) &= 0 + h'(E(Y))^2 \text{Var}(Y) + \dots . \end{aligned}$$

- Thus, for $\text{Var}(h(Y))$ to be approximately constant (up to first order Taylor approximation), we need

$$h'(E(Y)) \propto \text{Var}(Y)^{-1/2},$$

which suggests that we need h to be the antiderivative,

$$h(y) = \int \frac{1}{\sqrt{\text{Var}(y)}} dy.$$

- We should be aware of **common instances**.

1. If y_i are counts, then, often, the error **variance equals the mean** (e.g., Poisson) or is **proportional to the mean** (under- or over-dispersed Poisson), i.e.,

$$\text{Var}(Y) \propto E(Y).$$

This suggest $h(y) = \sqrt{y}$ (because, then, $h'(y) = dy^{1/2}/dy \propto y^{-1/2}$ so that $h'(E(Y)) = E(Y)^{-1/2}$ and $h'(E(Y))^2 = E(Y)^{-1}$, which cancels $\text{Var}(Y) \propto E(Y)$ to give approximately constant variance.)

- To illustrate, using an SLR, $E(Y|x) = \beta_0 + \beta_1 x$, then the variance behaves linearly in x , or the standard deviation, σ , behaves like \sqrt{x} , which is illustrated by case (3) of the above, eye-calibrating plots. If we use a common rule-of-thumb that the range (of residuals) is about 4σ , then the range (“pattern envelope” or “outline”) of the residual fan behaves like \sqrt{x} .

2. More severely, if the **variance behaves like the square of the mean**, i.e., if

$$\text{Var}(Y) \propto E(Y)^2,$$

then this suggests $h(y) = \log(y)$. In other words, we have a **constant coefficient of variation**

$$CV = E(Y)/\sqrt{\text{Var}(Y)} \propto c$$

(e.g., lognormal, gamma distributions).

- To illustrate, if we have an SLR, $E(Y | x) = \beta_0 + \beta_1 x$, then the variance behaves quadratically in x (i.e., like x^2), or the standard deviation, σ , behaves like x , which is illustrated by case (2) of the above, eye-calibrating plots. If we use a common rule-of-thumb that the range (of residuals) is about 4σ , then the range (“pattern envelope” or “outline”) of the residual fan behaves like x , like a line in x .

3. Even more severely, if the **variance behaves like the mean to the fourth power**,

$$\text{Var}(Y) \propto E(Y)^4,$$

then this suggests $h(y) = 1/y$.

- To illustrate, if we have an SLR, $E(Y | x) = \beta_0 + \beta_1 x$, then the variance behaves quartically in x (i.e., like x^4), or the standard deviation, σ , behaves like x^2 . If we use a common rule-of-thumb that the range (of residuals) is about 4σ , then the range (“pattern envelope” or “outline”) of the residual fan behaves like x^2 , quadratically in x (not illustrated!).

- Transforming y often goes hand-in-hand with transforming one or more covariates.
- Alternatively, we can model the variance of Y directly and keep the response on its original scale (which is suggested by the BP test, isn’t it?): INF 512.

A Juggling Act

- As we will see, in subsequent sections, we may also diagnose **non-linearity** of the regression function, which typically calls for transformations of an existing X and/or addition of regressors beyond those already in the model.
- When transforming Y to address non-constant variance, this may cause non-linearity in an X , and we may simultaneously have to consider transforming an X .
- Transformations to Y are also performed to address **non-normality** of Y (e.g., skewness). More generally, we should realize that diagnoses and transformations are often done together in some sense.
- In other words, some transformations for non-constant variance / non-normality may also fix (or break) non-linearity or non-normality, or vice versa. More on non-linearity and non-normality in subsequent sections.
- My **rule-of-thumb** is to try to get the regression model correct, then diagnose and, if necessary, remediate the variance, then return attention to the regression model to see if things are still okay.

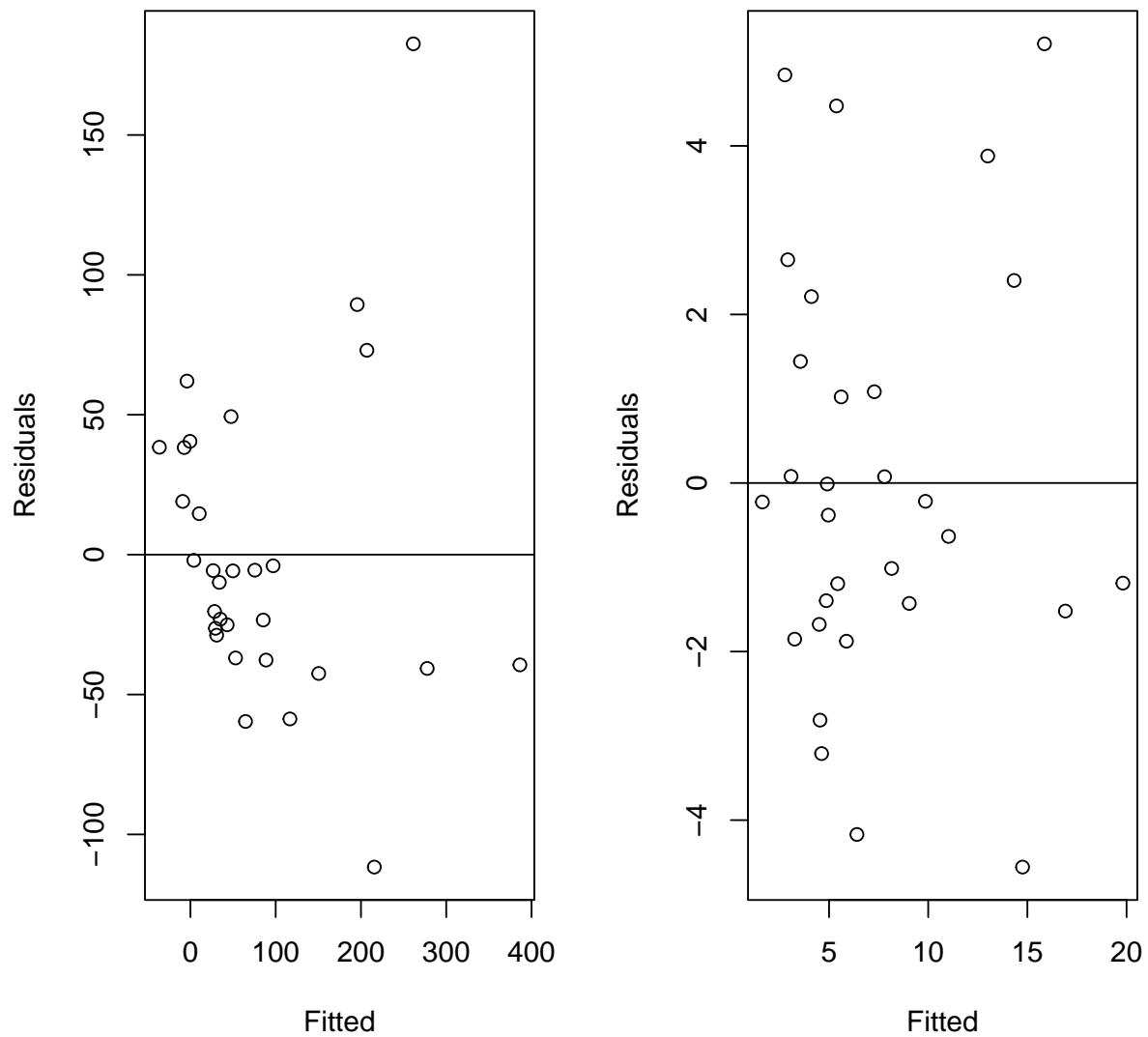
Box–Cox Procedure

- Instead of the Taylor series approximation approach, above, we can estimate λ in y_i^λ using a **normal maximum likelihood approach** to addressing non-constant variance (and normality).
- In other words, we choose λ , along with β and σ^2 in our normal linear regression model to make the transformed data most likely under the linear model of normality and constant variance.

- This is what the Box–Cox procedure does.
- The Box–Cox procedure is presented in [Far14, §9.1], but I prefer to discuss it (briefly) here

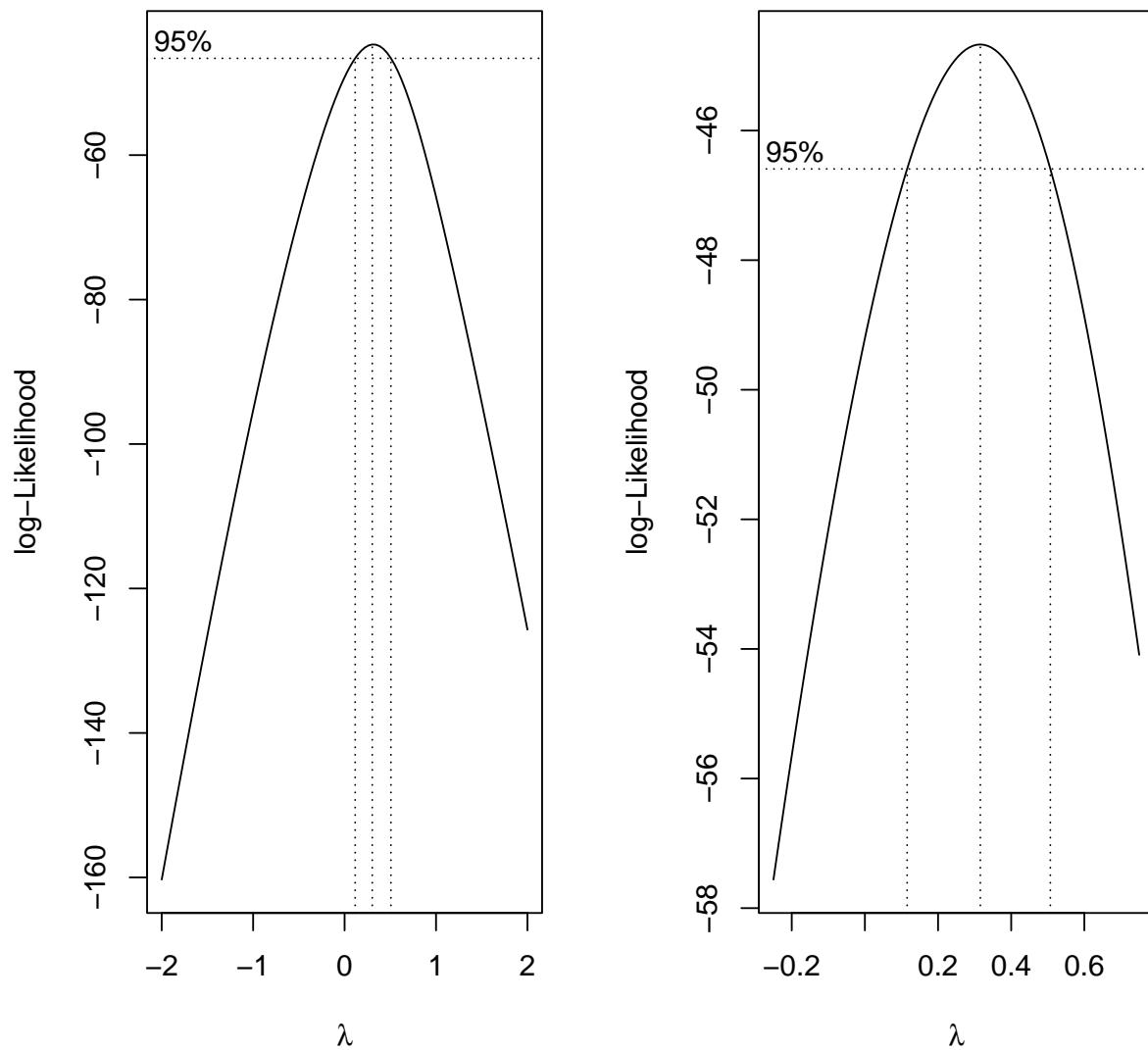
Galapagos Data Example

```
> ## Instead of ``guesstimating'' a sqrt transformation...
> data(gala, package="faraway")
> lmod <- lm(Species ~ Area + Elevation + Scruz + Nearest + Adjacent,
+             data=gala)
> par(mfrow=c(1,2))
> plot(fitted(lmod), residuals(lmod), xlab="Fitted", ylab="Residuals")
> abline(h=0)
> lmodsqrt <- lm(sqrt(Species) ~ Area + Elevation + Scruz + Nearest + Adjacent, gala)
> plot(fitted(lmodsqrt), residuals(lmodsqrt), xlab="Fitted", ylab="Residuals")
> abline(h=0)
```



```
> par(mfrow=c(1,1))
```

```
> ## ...we might instead try the Box-Cox procedure.  
> par(mfrow=c(1,2))  
> MASS::boxcox(lmod)  
> MASS::boxcox(lmod, lambda=seq(-0.25,0.75,by=0.05))
```



```
> par(mfrow=c(1, 1))
```

Remedial Actions for Non-Constant Variance

- As we've seen, transformations represent one remedial action to address non-constant variance.
- We will see weighted least squares (WLS) in [Far14, §8.2] and, more generally, generalized least squares (GLS) in [Far14, §8.1].
- INF 512 covers variance modeling even more generally.

6.1.2 Normality

- Your textbook's author appears to favor normal quantile-quantile plots (normal Q-Q plots) for diagnosing departures from normality; for one thing, we get to see all of the data points. He suggests the binning required for box-plots and histograms is problematic for checking normality. (???)
- More formal tests for departures from normality are motivated by the normal Q-Q plot. More on this, below.
- Perhaps should have discussed Box–Cox after this. (?)
- A stem and leaf plot may also be used (for smaller data sets).

Normal Probability Plot

- We present a slightly different derivation of the normal Q-Q plot than in [Far14, Chap. 6]. (The normal probability plot is just a special case of a Q-Q plot, a normal Q-Q plot (plot empirical quantiles (here, residual quantiles) vs. corresponding expected quantiles of a probability model (here, normal)).

- Let $\hat{\epsilon}_i$, $i = 1, \dots, n$ be residuals. Denote the ordered residuals as $\hat{\epsilon}_{(k)}$, $k = 1, \dots, n$.
- Plot $\hat{\epsilon}_{(k)}$ versus $\sqrt{MSE}z\left(\frac{k-0.375}{n+0.25}\right)$, $k = 1, \dots, n$.
- The procedure comes from theory that says

$$E(\hat{\epsilon}_{(k)}) \approx \sqrt{MSE}z\left(\frac{k-0.375}{n+0.25}\right),$$

if $\epsilon \sim N(0, \sigma^2)$.

- Thus, the sample quantiles (of the residuals) should roughly match quantiles from a normal distribution, and the plot should, in some average sense, approximate a scatter about the 1–1 line (45 degree line) *IF* we scale the standard normal quantiles by \sqrt{MSE} so that they are comparable in scale to the residuals (or if we scale the residuals by $1/\sqrt{MSE}$ and compare to unscaled normal quantiles). (Otherwise, we may see the slope of the line to be (about) \sqrt{MSE} , as in the plots, below.)
- Based on what we have learned (will learn) about residuals, $\sqrt{MSE}(1 - h_{ii})$ may be a better scale factor! Why? (This is the scale factor used by `plot.lm(x, which=2)`.)
- If the ϵ_i are not normally distributed, we expect to see a systematic departure from the 1–1 line pattern.
- Recall $z(p)$ is the standard normal value with left-tail probability p . Thus,

$$z\left(\frac{k-0.375}{n+0.25}\right)$$

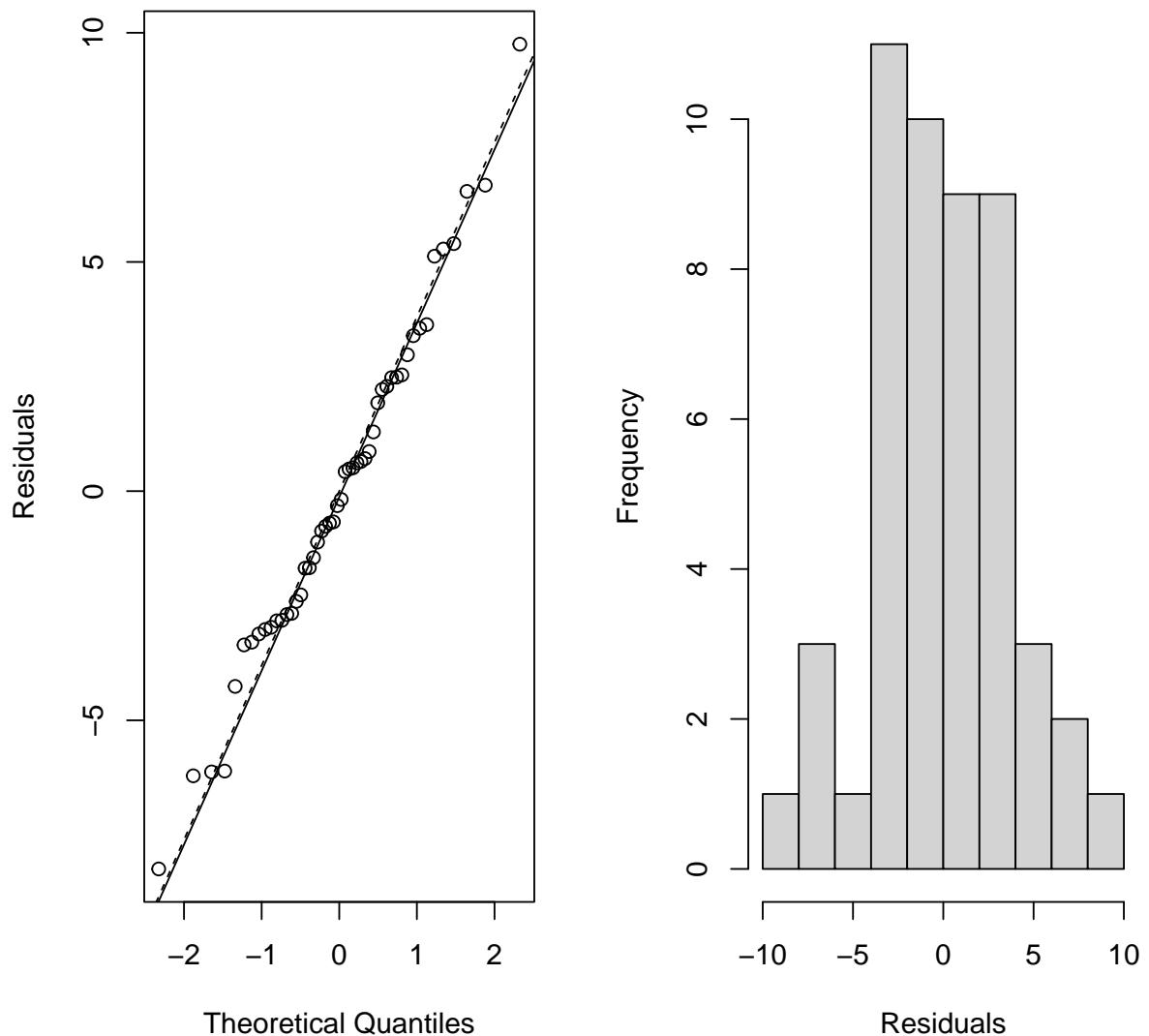
is the standard normal value with left tail probability

$$\frac{k-0.375}{n+0.25}.$$

- **NOTE:** `qqnorm` and `qqline` do NOT appear to scale the residuals or standard normal quantiles, so we do not see a 1-1 line, but a line with slope about \sqrt{MSE} . In any case, departures from a line indicate non-normality.

Savings Data Example

```
> par(mfrow=c(1,2))
> lmod <- lm(sr ~ pop15+pop75+dpi+ddpi, data=savings)
> qqnorm(residuals(lmod), ylab="Residuals", main="")
> qqline(residuals(lmod))
> abline(c(0,summary(lmod)$sigma), lty=2) ## sqrt(MSE) slope
> hist(residuals(lmod), xlab="Residuals", main="")
```

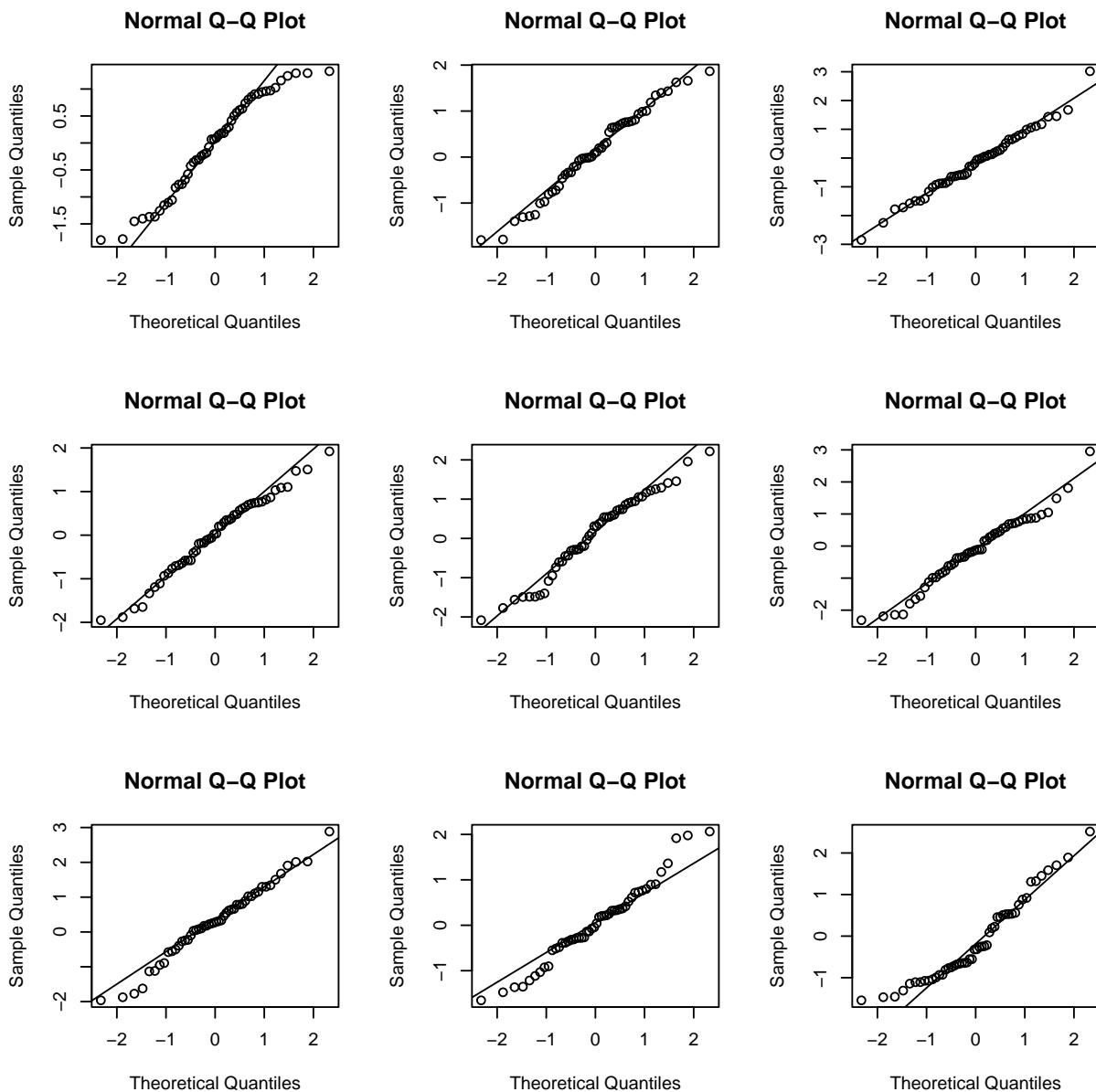


```
> par(mfrow=c(1,1))
```

Calibrate Your Eye

- Calibrate/train your eye with the code/plots that follow. (Try the commented code yourself.)

```
> n <- 50
> par(mfrow=c(3,3))
> for(i in 1:9) {x <- rnorm(n); qqnorm(x); qqline(x)}
```



```
> ##for(i in 1:9) {x <- exp(rnorm(n)); qqnorm(x); qqline(x)}
> ##for(i in 1:9) {x <- rcauchy(n); qqnorm(x); qqline(x)}
> ##for(i in 1:9) {x <- runif(n); qqnorm(x); qqline(x)}
> par(mfrow=c(1,1))
```

Correlation Tests of Normality

- The normal probability plot can be used to motivate more formal tests of the normality assumption.
- We discuss 2 similar tests.
- The idea behind the tests is to estimate the correlation between the ordered residuals and their expectations, as depicted in the normal probability plot.
- If the ordered residuals and their expected values fall near the 1–1 line (if scaled so), correlation is high (near 1), suggesting no departure from normality, otherwise, correlation is low, suggesting errors are not well-modeled as normal.

Test 1

Test 1 Let $\hat{\epsilon}_{(k)}$, $k = 1, \dots, n$, be the ordered residuals, lowest to highest. We know

$$\sqrt{MSE} z \left(\frac{k - 0.375}{n + 0.25} \right),$$

which we write as $z(k)$ for convenience in what follows, is the approximate expected value of $\hat{\epsilon}_{(k)}$.

–**Hypotheses** H_0 : Errors are normal. H_a : Errors are not normal. (Choose α).

–**Test Statistic**

$$r^* = \frac{\sum_{k=1}^n (\hat{\epsilon}_{(k)} - \overbrace{\bar{\epsilon}}^0)(z(k) - \overbrace{\bar{z}}^0)}{s_{\hat{\epsilon}} s_z},$$

where $s_{\hat{\epsilon}}$ and s_z are the (sample) standard deviations of the $\hat{\epsilon}_{(k)}$ (or $\hat{\epsilon}_i$) and of the $z(k)$, respectively.

–**Critical Value** r_{crit} is obtained from [KNNL05, Table B.6].

–**Decision Rule and Conclusion:** Reject null of normal errors if $r^* < r_{\text{crit}}$.

State your conclusion appropriately as in the many other testing situations we've discussed.

–**Remediation** Take remedial action if you reject the null. This typically means, for us, transforming the response Y . We consider non-normal models for Y in INF 512.

Savings Data Example

- The following chunk illustrates computations for Test 1 applied to the running savings data example.

```
> ## We use the savings data example lmod object, from previous chunks,
> ## to illustrate "Test 1" in our notes. This is the test discussed
> ## in Section 3.5 of Kutner et al:
>
> lmod.res<- residuals(lmod)
> (n <- length(lmod.res))

[1] 50

> ## Approximate expected values of ordered residuals assuming errors
> ## are normal:
> lmod.zk<- qnorm((1:n - 0.375)/ (n + 0.25))
>
> mean(lmod.zk) ## numerically zero as expected (just lookin')

[1] -1.3088e-17
```

```

> ## Correlation between ordered residuals and their expected values:
> (r<- cor(sort(lmod.res), lmod.zk))

[1] 0.99252

> ## Alternatively:
> (R2<-summary(lm(sort(lmod.res) ~ lmod.zk))[[ "r.squared"]])

[1] 0.98509

> (r<- sqrt(R2))

[1] 0.99252

> ## Consulting Kutner et al Table B.6 with n=50 and, say, alpha=0.05
> ## we find the critical value, rcrit = 0.977. Since r > rcrit, we
> ## do NOT reject the null of normally distributed errors.

```

Test 2: Shapiro-Wilk

Test 2: Shapiro–Wilk Test Similar to Test 1, but not as simple. We will use R exclusively to compute this test; details omitted.

- Hypotheses** H_0 : Errors are normal. H_a : Errors are not normal. (Choose α).
- Test Statistic** W (see R)
- Critical Value** Instead of comparing W to, say, W_{crit} , which we do not have, we compare the p-value to α .
- Decision Rule and Conclusion:** Reject if p-value $< \alpha$. State your conclusion appropriately as in the many other testing situations we've discussed.
- Remediation** Take remedial action if you reject the null (e.g., transform Y ; use generalized linear model (GLM) (INF 512).)

Savings Data Example

```
> ### Now Test 2 in our notes. This is the Shapiro-Wilk test.  
> ### We use R for this test. Same null as before. We look at  
> ### the p-value and compare to our favorite alpha value just  
> ### like any test we've done. See LMwR2e p. 81.  
> shapiro.test(residuals(lmod))
```

```
Shapiro-Wilk normality test
```

```
data: residuals(lmod)  
W = 0.987, p-value = 0.85
```

NOTE: In the current example, the two tests agree, but this may not always be the case.

Remedial Actions for Non-Normality

- We've seen the Box-Cox procedure.
- If we have a large sample size (and we have confidence in our regression model and our (co)variance model, but not normality), then we might do nothing but simply rely on the CLT for asymptotically correct (if not optimal) inference.
- INF 512 deals directly with modeling non-normal responses using other distributions.

6.1.3 Correlated Errors

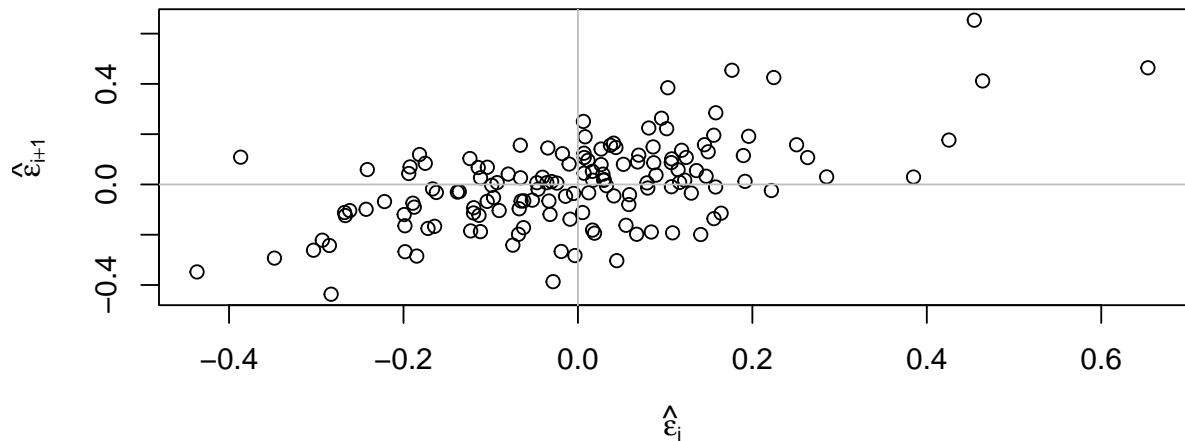
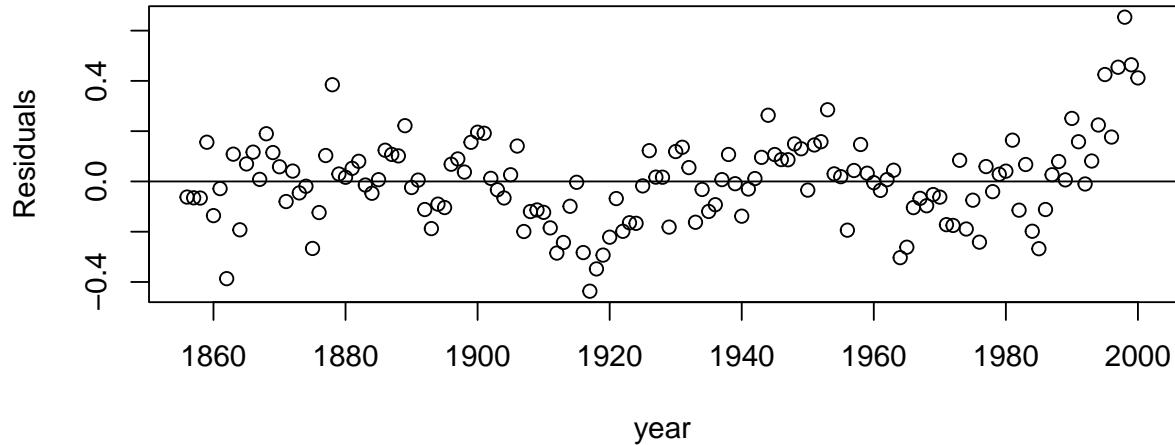
- Correlated errors often occur with **temporal or spatial data**, wherein nearby errors tend to be more alike than errors further apart—this positive correlation is most typical. (Potentially in INF 512.)

- Often, there may be an **unobserved covariate**, omitted from our regression model, that varies over space, time or over some other biological or physical scale. Thus, when such a covariate is omitted from a model, the variability in the response which would otherwise be attributed to the covariate, remains in the residuals, which then may reflect the nature of the scale of the omitted covariate, typically including positively correlated residuals (over space, time or other scale).
- Or, **repeated measurements** on the same unit/subject (**grouped data**) often result in errors that are more alike within the same unit/subject than errors across subjects.
- We will look at generalized least squares (GLS) in [Far14, §8.1] as a way to accommodate correlated errors. INF 512 covers modeling of correlated data more generally.
- We illustrate by following the **global warming example** in your textbook ([Far14, §6.1.3]) wherein we have average Northern Hemisphere Temperature from 1856 to 2000, as a response, and eight climate proxies and year as potential predictor variables (tree ring, ice core and sea shell climate proxies).

Global Warming Data Example

```
> data(globwarm, package="faraway")
> lmod <- lm(nhtemp ~ wusa + jasper + westgreen + chesapeake +
+               tornetrask + urals + mongolia + tasman,
+               data=globwarm)
> par(mfrow=c(2,1))
> ## First plot
> plot(residuals(lmod) ~ year, na.omit(globwarm), ylab="Residuals")
> abline(h=0)
> n <- length(residuals(lmod))
> ## Second plot (residual lag plot)
> plot(tail(residuals(lmod), n-1) ~ head(residuals(lmod), n-1),
+       xlab= expression(hat(epsilon)[i]),
```

```
+     ylab=expression(hat(epsilon)[i+1]))
> abline(h=0, v=0, col=grey(0.75))
```



```
> par(mfrow=c(1,1))
```

```
> ## Auto-regression suggested by above lag plot
> summary(lm(tail(residuals(lmod), n-1) ~ head(residuals(lmod), n-1) - 1))
```

```

Call:
lm(formula = tail(residuals(lmod), n - 1) ~ head(residuals(lmod),
n - 1) - 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.3697 -0.0739  0.0073  0.0767  0.3831 

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)    
head(residuals(lmod), n - 1)  0.5951     0.0693   8.59  1.4e-14 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 0.139 on 143 degrees of freedom
Multiple R-squared:  0.34, Adjusted R-squared:  0.336 
F-statistic: 73.7 on 1 and 143 DF,  p-value: 1.39e-14

```

- What do you think?

Durbin-Watson Test for Correlated Errors

You might hear of the Durbin-Watson test for auto-correlated errors (auto = self). We omit details and go to code/results in your textbook.

```

> ## require(lmtest)
> ## Default alternative hypothesis is positive auto-correlation (common)
> lmtest::dwtest(nhtemp ~ wusa + jasper + westgreen + chesapeake +
+                  tornetrask + urals + mongolia + tasman,
+                  alternative="greater", ## default
+                  data=globwarm)

Durbin-Watson test

data: nhtemp ~ wusa + jasper + westgreen + chesapeake + tornetrask + urals + mongol
DW = 0.817, p-value = 1.4e-15
alternative hypothesis: true autocorrelation is greater than 0

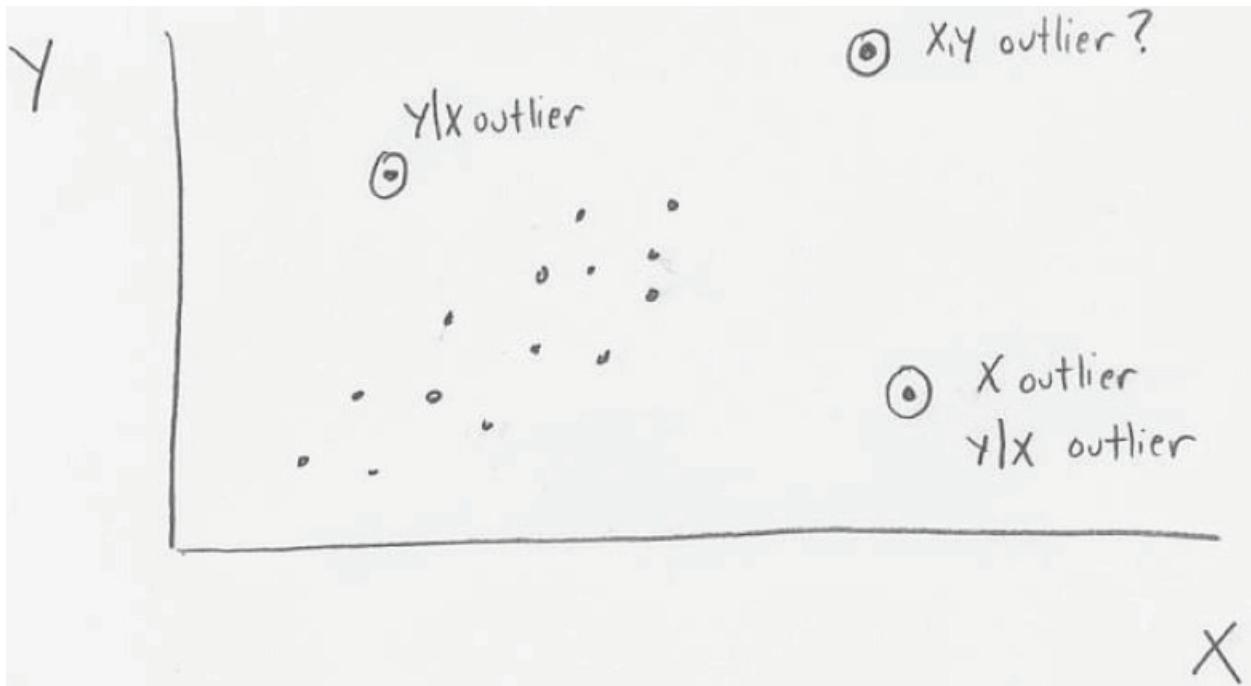
```

Remedial Actions for Correlated Errors

- Use covariates, time series methods, spatial statistics, GLS, mixed models, generalized estimating equations, INF 512.
- Assuming our mean model is correct but our covariance model is not, then, as mentioned, our standard errors are not correct, and our p-values and CI confidence levels are not correct. And, more complicated covariance remodeling comes with the risk of losing consistency of mean model parameters, even if our mean model is correct, depending on how fancy a covariance model we consider. Again, more in INF 512.

6.2 Finding Unusual Observations (Outliers)

- An **outlier** may be (somewhat loosely) defined as a data point (\mathbf{x}_i, y_i) that does not follow the general pattern of the data. The point may be outlying in terms of its y coordinate, its \mathbf{x} coordinate, or both.
- An observation (y_i, \mathbf{x}_i) is said to be **influential** if its omission from the data results in “large” changes to the fitted regression function.



6.2.1 Leverage

Outlying X : Hat Matrix Leverage Values

- The **hat matrix** (note §2.4) is fundamental in assessing influential and outlying observations.
- To gain some insight into the use of \mathbf{H} for identifying potentially outlying X values, consider what we already know:

$$\hat{\mathbf{Y}} = \mathbf{HY} \quad \text{or}$$

$$\hat{Y}_i = h_{i1}Y_1 + h_{i2}Y_2 + \cdots + h_{ii}Y_i + \cdots + h_{in}Y_n,$$

where h_{ij} is the ij th entry of $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$.

- We see h_{ii} as the **weight** of observation Y_i in determining the corresponding prediction (or estimate) \hat{Y}_i .
- h_{ij} is determined solely by \mathbf{X} (it is not affected by the response data).

- In particular, $h_{ii} = \mathbf{x}'_i(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_i$ where \mathbf{x}'_i is the i th row of \mathbf{X} and is called the **leverage** of the i th observation (terminology explained shortly).
- The leverage, h_{ii} , can be interpreted as the **squared length of the i th (scaled) covariate** \mathbf{x}_i (i th row vector of \mathbf{X} , not column; p dimensions, not n). (Consider the scaled vector $\mathbf{z}_i = \mathbf{L}^{-1}\mathbf{x}_i$ where \mathbf{L} is a matrix such that $\mathbf{LL}' = \mathbf{X}'\mathbf{X}$ (e.g., Cholesky factor)...).
- $0 \leq h_{ii} \leq 1$ (details omitted).
- $\sum_{i=1}^n h_{ii} = p$ (i.e., $\text{trace}(\mathbf{H}) = p$; details omitted). (There is a connection here to what is known as the **effective degrees of freedom** of a smoother matrix; INF 504.)
- Thus, $\bar{h} \equiv \sum_{i=1}^n h_{ii}/n = p/n$
- Thus, roughly speaking, if h_{ii} is “large”, i.e., “much above” $\bar{h} = p/n$, then \mathbf{x}_i is relatively “longer” than average and, thus, is “far” from the average of the observed \mathbf{x}_i ; hence **outlying** in this sense.
- And, we see that such “long” \mathbf{x}_i give Y_i a large weight in determining the corresponding fitted value \hat{Y}_i .
- What happens as h_{ii} approaches 1? (From §6.1, we have $\text{Var}(Y_i - \hat{Y}_i) = \sigma^2(1 - h_{ii})$.)

- To summarize/discuss:
 - Thus, outlying (long, scaled) \mathbf{x}_i covariates can pull fitted values \hat{Y}_i toward corresponding observed values Y_i .
 - For this reason, we refer to h_{ii} as the **leverage** of the i th case.
 - A rough **rule-of-thumb** is that \mathbf{x}_i is a *potential* outlier if $h_{ii} > 2p/n$, twice the average, \bar{h} .

- Generally, an influential point may/may not be an outlier or may/may not have high leverage, but, often, an influential point is an outlier and/or has high leverage; thus, we look at outliers and leverage.
- Here, we just use h_{ii} to identify *potential* outlying \mathbf{x}_i , which may or may not belong to an influential case.
- **Influential observations** are the subject of a subsequent section.
- We use the savings data example to illustrate, as in [Far14, §6.2.1].

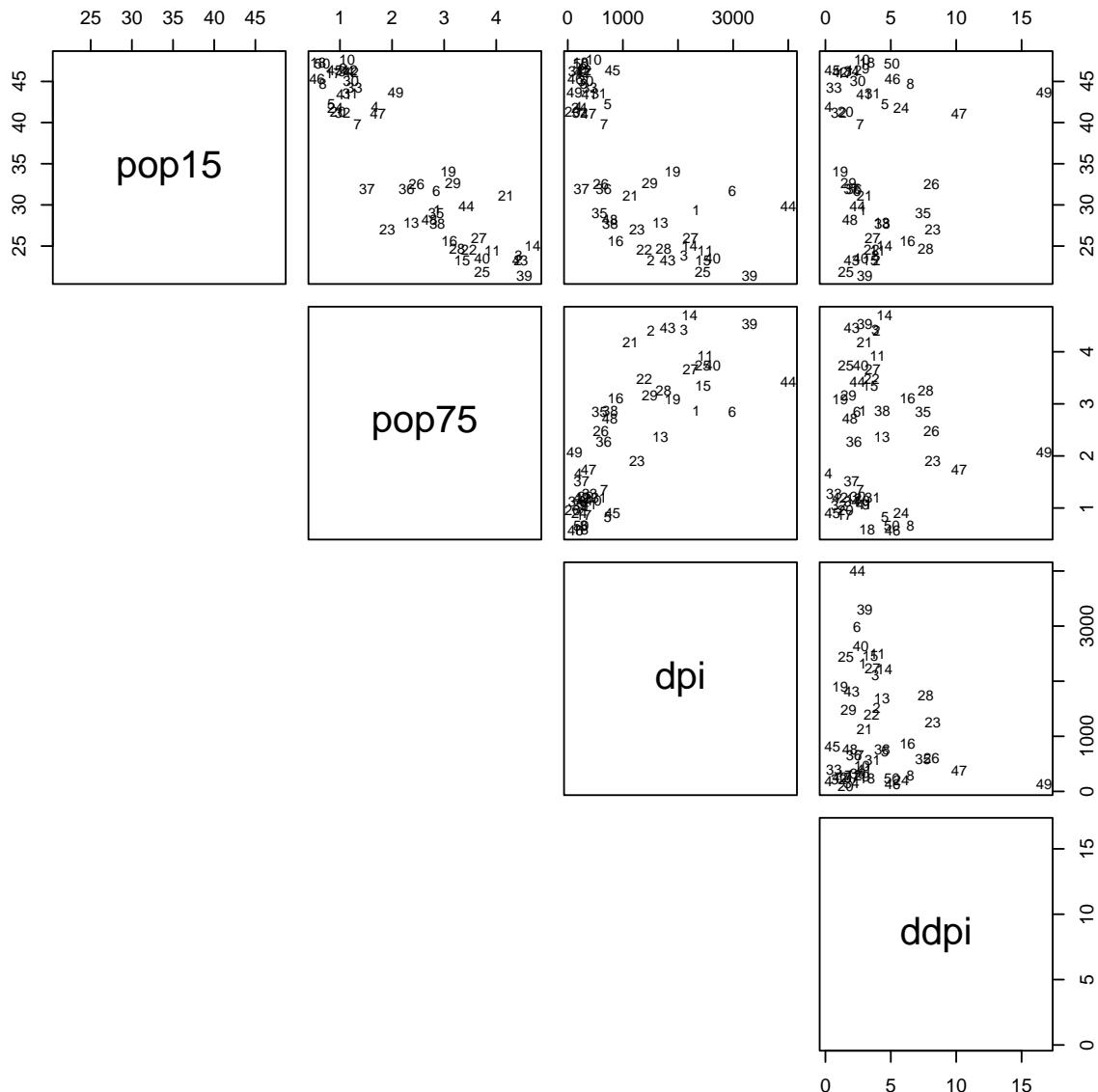
Savings Data Example

```
> lmod <- lm(sr ~ pop15 + pop75 + dpi + ddpi,
+             data=savings)
> hatv <- hatvalues(lmod)
> head(hatv)

Australia    Austria    Belgium    Bolivia    Brazil    Canada
0.067713  0.120384  0.087482  0.089471  0.069559  0.158402

> sum(hatv) ## trace(H)=p
[1] 5
```

```
> pairs(~ pop15 + pop75 + dpi + ddpi,
+       data=savings,
+       panel=function(x,y,...)
+         text(x=x,y=y,labels=as.character(1:dim(savings)[1]),...),
+       lower.panel=NULL,
+       cex=0.8)
```



```

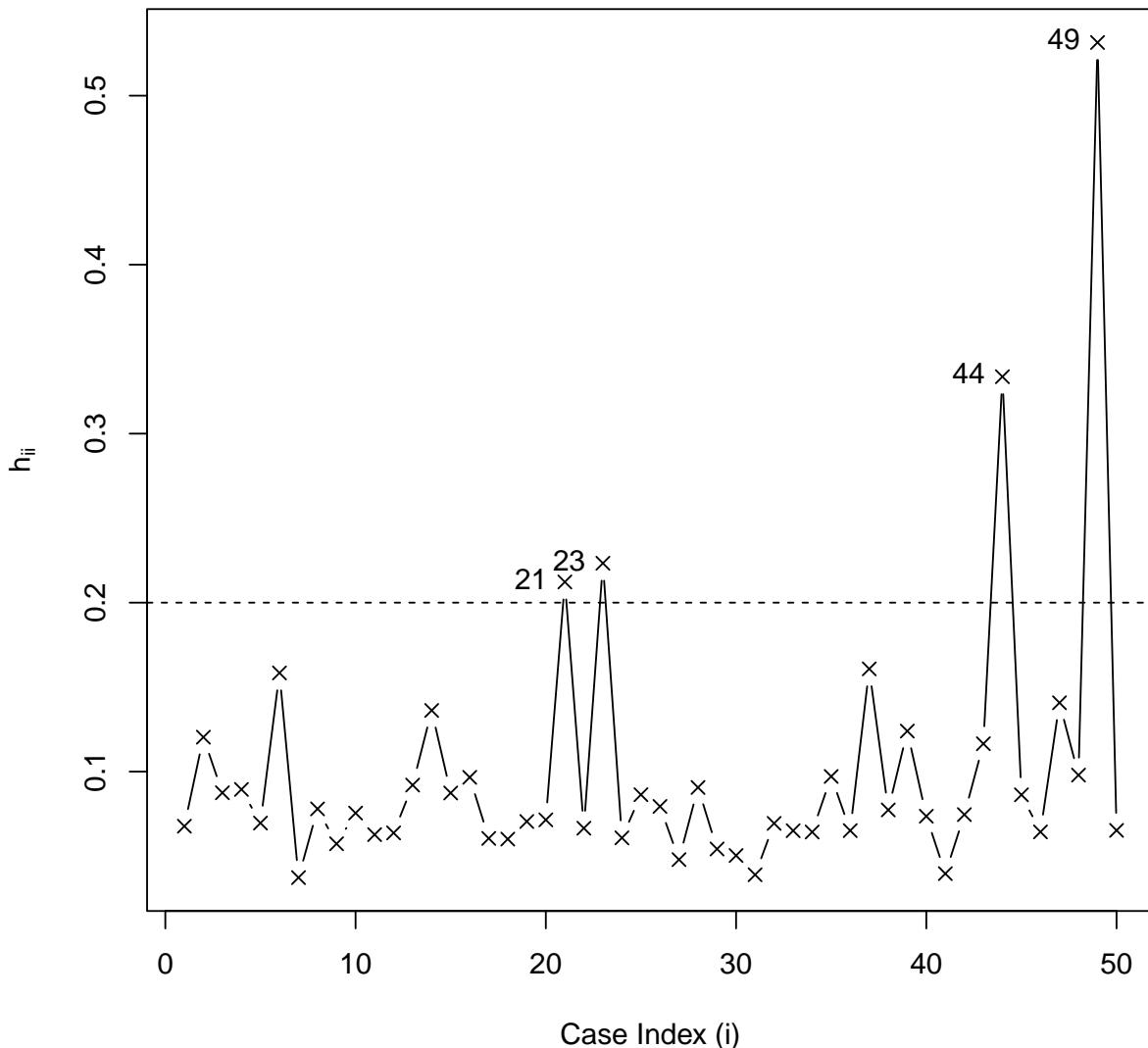
> ## Let's look at some hat (hii) values:
> hii<- hatvalues(lmod)
> plot(hii,type="b",pch=4,xlab="Case Index (i)",ylab=expression(h[i][i]))
>
> ## Add ``hcrit'' (2 p/n) reference line:
> (n<- length(hii)); p<- 5
[1] 50

```

```
> hcrit<- 2*p/n
> abline(h=hcrit, lty=2)
>
> ## Identify potential outlying X values on the plot:
> (outi<- which(hii > hcrit)) ## same as on pg 412 of textbook
```

Ireland	Japan	United States	Libya
21	23	44	49

```
> text(x=outi, y=hii[outi], labels=outi, pos=2)
```



```
> ## Confirm hatvalues function gives same as diag(H),
> ## H=X(X^tX)^(-1)X^t
> Xmat<- model.matrix(lmod)
> Hmat<- Xmat%*%solve(crossprod(Xmat))%*%t(Xmat)
> head(diag(Hmat))
```

	Australia	Austria	Belgium	Bolivia	Brazil	Canada
	0.067713	0.120384	0.087482	0.089471	0.069559	0.158402

```
> which(diag(Hmat) > hcrit)
```

Ireland	Japan	United States	Libya
21	23	44	49

```

> ## Or, with Cholesky,  $H = X(X^t X)^{-1} X^t = (L^{-1} X)^t (L^{-1} X^t)$ 
> cholL<- t(chol(crossprod(Xmat)))
> LinvXt<- forwardsolve(cholL, t(Xmat))
> head(diag(Hmat<- crossprod(LinvXt)))

[1] 0.067713 0.120384 0.087482 0.089471 0.069559 0.158402

> which(diag(Hmat) > hcrit)

[1] 21 23 44 49

> ## Or, with QR decomposition,  $H = (R^{-t} X^t)^t (R^{-t} X^t)$ 
> Rmat<- qr.R(qr(Xmat))
> RtinvXt<- backsolve(Rmat, t(Xmat), transpose=TRUE)
> ##RtinvXt<- forwardsolve(t(Rmat), t(Xmat))
> head(diag(Hmat<- crossprod(RtinvXt)))

[1] 0.067713 0.120384 0.087482 0.089471 0.069559 0.158402

> which(diag(Hmat) > hcrit)

[1] 21 23 44 49

```

6.2.2 Outliers

Outlying Y : Studentized Deleted Residuals

- The **rough idea underlying the detection of outlying Y :**

- If Y_i is “extreme” then

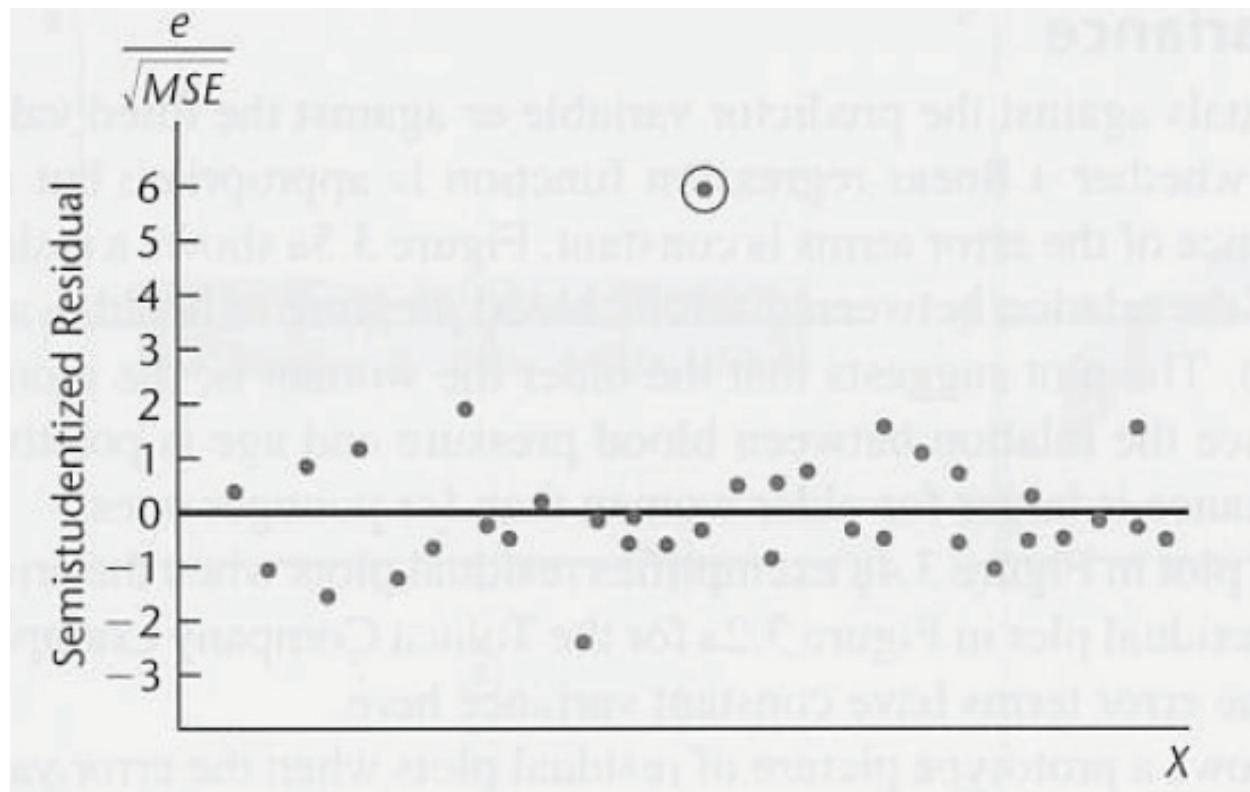
$$\hat{\varepsilon}_i = Y_i - \hat{Y}_i$$

will be extreme, and we should be able to use these residuals to detect such extreme cases.

- But, as alluded to in our previous discussion of leverage, this idea is a bit too simplistic.
- From §6.1 and basic properties of the hat matrix, we have $Var(\hat{\varepsilon}_i) = \sigma^2(1 - h_{ii})$.
- In other words, high leverage points will cause $Var(Y_i - \hat{Y}_i)$ to be relatively small, i.e., the fitted value \hat{Y}_i will be “pulled” relatively close to the observed value, Y_i , thus potentially masking an outlying Y_i . (You might say that the residual may be influenced against identifying outlying Y_i .)
- Conversely, large (in absolute value) $\hat{\varepsilon}_i$ may be due to an outlying observation Y_i or to $\hat{\varepsilon}_i$ having a relatively large variance (associated with low leverage h_{ii}) compared to other $\hat{\varepsilon}_j$, $j \neq i$.
- We follow these ideas in the following sequence of residual types that you may encounter in practice.

Semistudentized Residual

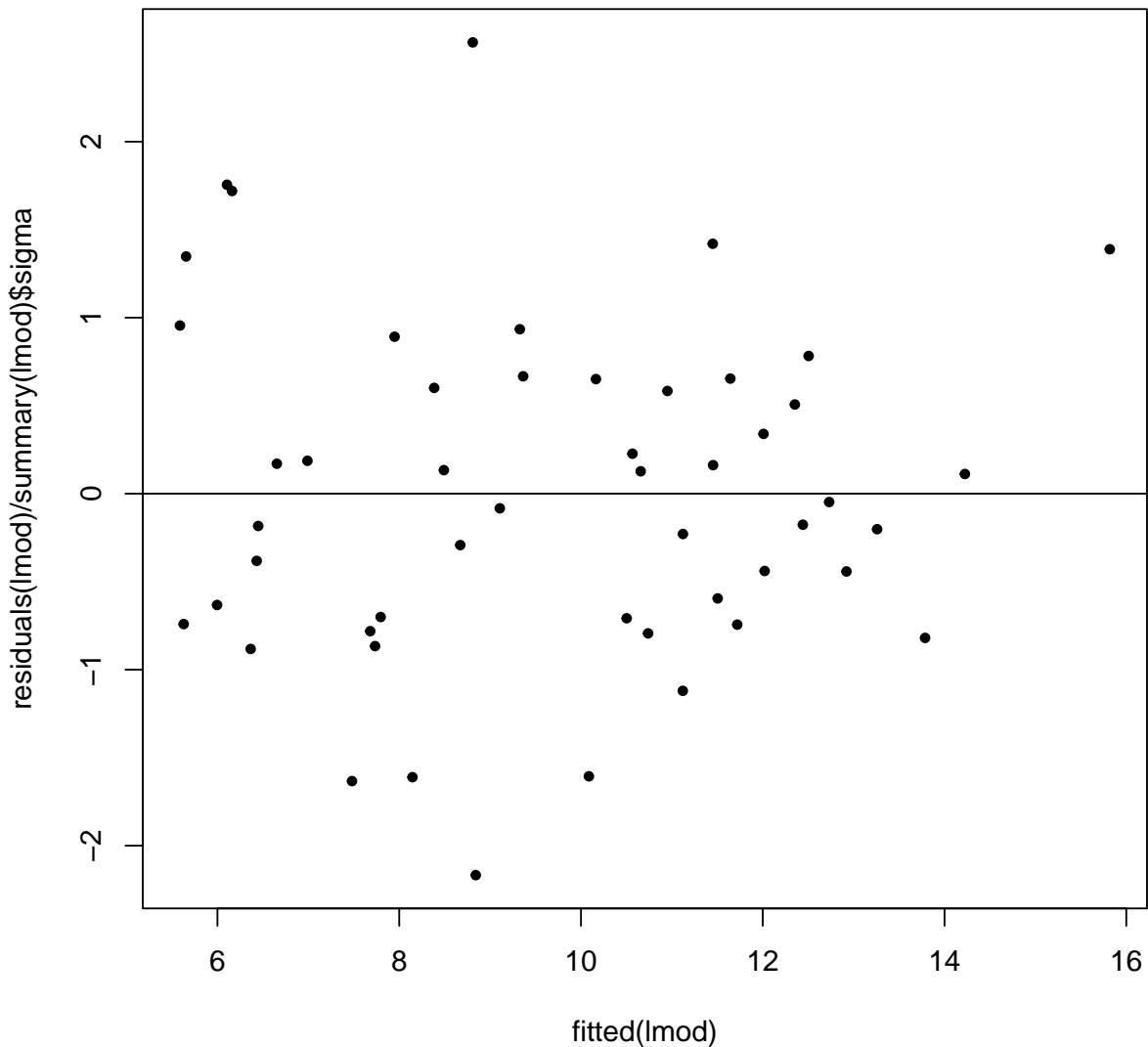
$$r_i^* = \frac{\hat{\varepsilon}_i - 0}{\sqrt{MSE}} = \frac{\hat{\varepsilon}_i}{\sqrt{MSE}} \quad i = 1, \dots, n.$$



- From our knowledge of residuals, **semistudentized** residuals are NOT correctly standardized, or “studentized”, to get an exact (Student’s) t–ratio (if assumptions hold) (hence “semi”).

```
> ## Savings Data Example again
> ## Semistudentized residuals
> plot(residuals(lmod)/summary(lmod)$sigma ~ fitted(lmod),
+       pch=20, main="Semistudentized Residuals")
> abline(h=0)
```

Semistudentized Residuals



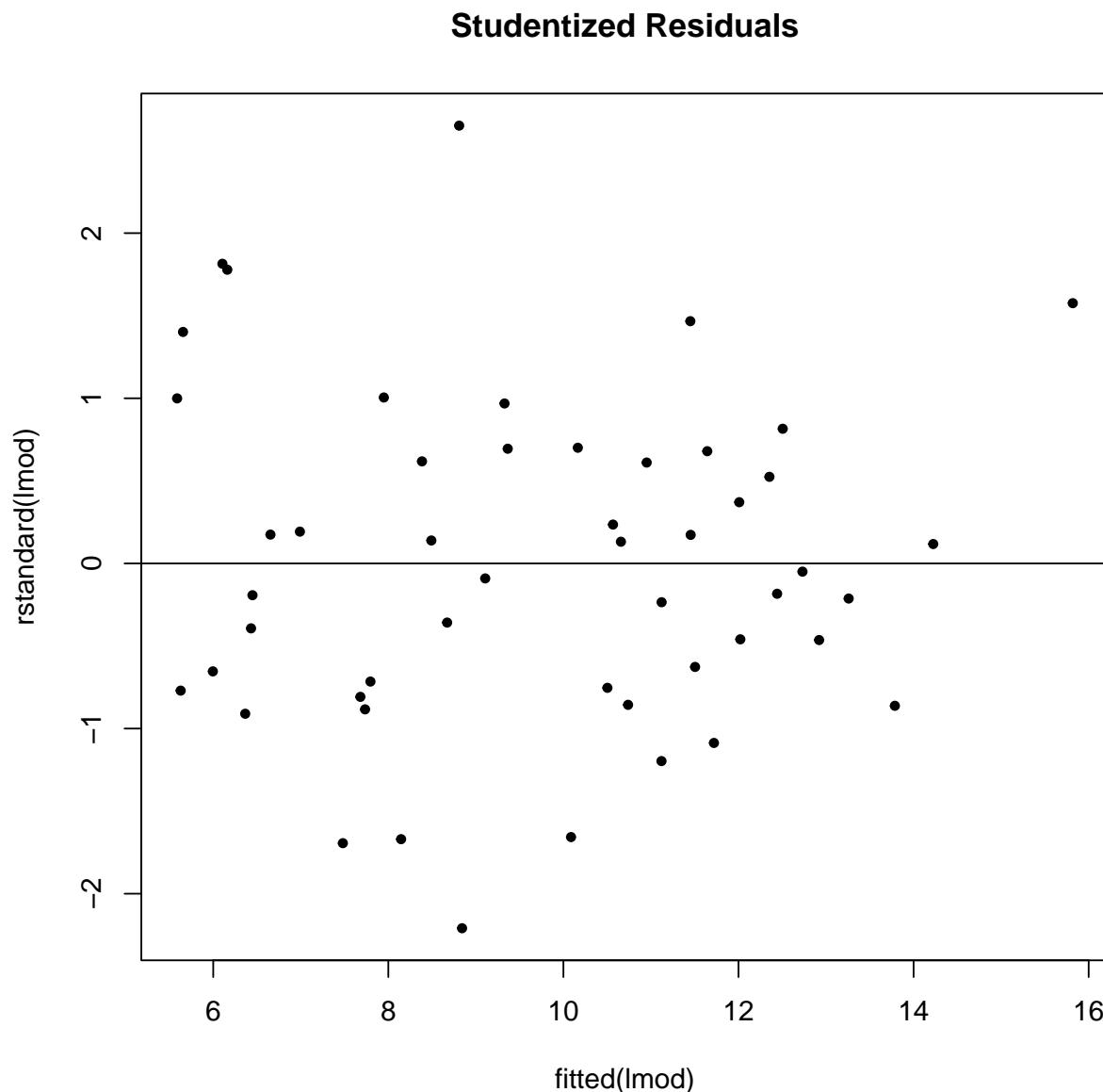
Studentized Residual

$$r_i = \frac{\hat{\varepsilon}_i}{\sqrt{MSE(1 - h_{ii})}} \quad i = 1, \dots, n,$$

([Far14, p. 84]) which follow a t distribution exactly (if assumptions hold).

These are also called **internally studentized residuals** because Y_i is used inside MSE ; shortly, we will see externally studentized residuals, which use an MSE that omits Y_i .

```
> ## Savings Data Example again
> ## (internally) Studentized residuals (automatically)
> plot(rstandard(lmod) ~ fitted(lmod), pch=20,
+       main="Studentized Residuals")
> abline(h=0)
```



```
> ### ...the identify function will wait for you to click the plot n=1
> ### times...
>
> ##identify(y=rstandard(lmod), x=fitted(lmod), n=1)
```

- The **deleted residuals**

$$d_i = Y_i - \hat{Y}_{i(i)} \quad i = 1, \dots, n,$$

where

$$\hat{Y}_{i(i)}$$

is the **deleted prediction**.

- (Incidentally, $Y_i - \hat{Y}_{i(i)}$ is also known as **PRESS prediction error for the i th observation**; [KNNL05, Chap. 9] discuss deleted predictions and PRESS (prediction (or predictive) sum of squares) in the context of model selection; these may be obtained for linear models using `rstandard(*, type = 'predictive')`, and summing these then dividing by n gives the leave-one-out cross-validation LOOCV (i.e., n -fold CV or `CV(n)`) estimate of prediction (or generalization) error. More on CV in INF 504.
- Of course, (conceptually) “deleted” comes from removing the potential outlier, Y_i , from the data, refitting the regression, without Y_i , then predicting Y_i , hence the notation $\hat{Y}_{i(i)}$ for the (deleted) prediction.
- Now, Y_i will not influence the fitted value $\hat{Y}_{i(i)}$, which we mentioned may be a problem for assessing Y_i to be an outlier.
- Thus, if we somehow Studentize deleted residuals, then we have a t (if assumptions hold) for assessing outlying Y_i that is not influenced by Y_i itself.
- Note that we are predicting Y_i using $\hat{Y}_{i(i)}$ (from the refit), thus, from §4.1, we have the estimated standard error of prediction

$$\widehat{s.e.}(d_i) = \sqrt{MSE_{(i)}(1 + \mathbf{x}'_i(\mathbf{X}'_{(i)}\mathbf{X}_{(i)})^{-1}\mathbf{x}_i)}.$$

- Note, \mathbf{x}_i is the i covariate vector associated with response Y_i , $\mathbf{X}_{(i)}$ is the matrix consisting of rows of $\mathbf{x}_{i'}$, $i' \neq i$ and $MSE_{(i)}$ is simply the usual estimate of σ^2 from the LS fit without case i .

- That is, we are merely computing the fitted value or “ $\mathbf{C}\hat{\boldsymbol{\beta}}$ ” for $\mathbf{C} = \mathbf{x}'_i$ and $\hat{\boldsymbol{\beta}}$ from the LS fit that omits the i th case ($\hat{\boldsymbol{\beta}}_{(i)}$??). Simple; and our previous results for variance and standard error hold unchanged up to notation and bookkeeping!

- And, as in §4.1, we standardize the deleted residuals by dividing by $\widehat{s.e.}(d_i)$ to get **studentized deleted residuals** ([Far14, p. 87])

$$t_i = \frac{y_i - \hat{y}_{i(i)}}{\sqrt{MSE_{(i)}(1 + \mathbf{x}'_{(i)}\mathbf{X}'_{(i)}\mathbf{X}_{(i)})^{-1}\mathbf{x}_i)}} \quad \text{Conceptual Formula,}$$

and, if assumptions hold,

$$t_i \sim t(n - 1 - p).$$

- Thus, we have a t (if assumptions hold) for assessing outlying Y_i that is not influenced by Y_i itself.
- Note that the degrees of freedom for this t distribution comes from $MSE_{(i)}$ and is $n - 1 - p$, not $n - p$, which you should know confidently if you've been following along well up to now.
- These are also called **externally studentized residuals**; “external” because the potentially offending Y_i is not inside $MSE_{(i)}$ or is “external to” $MSE_{(i)}$, and it is not used to predict $\hat{Y}_{i(i)}$.

- Note also that the original fit leverage h_{ii} can be used to simplify computations a bit (details omitted)

$$d_i = Y_i - \hat{Y}_{i(i)} = \frac{Y_i - \hat{Y}_i}{1 - h_{ii}} = \frac{\hat{\epsilon}_i}{1 - h_{ii}}$$

$$(1 - h_{ii})^{-1} = 1 + \mathbf{x}'_i (\mathbf{X}'_{(i)} \mathbf{X}_{(i)})^{-1} \mathbf{x}_i,$$

thus giving

$$t_i = \frac{y_i - \hat{y}_{i(i)}}{\sqrt{MSE_{(i)}(1 - h_{ii})}}$$

- Still, it appears that we must recompute the deleted fit to get $MSE_{(i)}$ for each i ! Fortunately, not!
- It turns out that (details omitted)

$$(n - p)MSE = (n - p - 1)MSE_{(i)} + \frac{\hat{\varepsilon}_i^2}{1 - h_{ii}}$$

to arrive at (after some algebra)

$$t_i = \hat{\varepsilon}_i \left[\frac{n - p - 1}{SSE(1 - h_{ii}) - \hat{\varepsilon}_i^2} \right]^{1/2} \quad \text{Computational Formula.}$$

- Nice! No refitting! All quantities are obtainable from the original fit, without omitting y_i ! Magic!
- **NOTE:** the last expression for t_i on [Far14, p. 87] still contains $MSE_{(i)}$ ($\hat{\sigma}_{(i)}$), and your textbook's author mentions that it avoids the refitting of n regressions, but this is not obvious. (Granted, the algebra we skipped to get our last expression may not be obvious, either, but, if accepted, then it's obvious our last expression for t_i avoids refitting.)
- Note, while I have labeled the last expression as “computational formula”, I do not actually know if this is computationally efficient, aside from avoiding refitting.
- Now, we can use the t_i to more formally assess outliers: if t_i is “large,” then we reject the null hypothesis that y_i is not an outlier in favor of the alternative hypothesis that it is an outlier.

- But, e.g., for $n = 100$, we would expect to reject the null (no outliers) 5 times assuming a significance level $\alpha^* = 0.05$ for each **individual** test. Thus, we may want to adjust the levels of the individual tests to avoid declaring an excess of outliers.
- In short, if we consider the Type I error for the **family** of n tests to be the false rejection of *one or more* of the n tests, then, to achieve a Type I error rate at least as small as a specified α for the **family** of tests, then The Bonferroni Type I error correction procedure tells us to conduct the **individual** tests at significance levels $\alpha^* = \alpha/n$. (We're just making it harder to reject the null of no outliers for each individual test.)
- Your textbook's author discusses the Bonferroni correction ([Far14, p. 87]), and I give more detail, below.

Side note: Bonferroni family-wise error rate

Consider 2 tests, and let A_1 and A_2 denote the events of falsely rejecting (Type I errors) for test i , $i = 1, 2$, respectively, and let α^* be the “individual-wise” or “statement-wise” error rate, i.e., $Pr(A_i) = \alpha^*$, $i = 1, 2$, i.e., the probability of a Type I error is α^* for each test, individually. Now, consider the event of falsely rejecting at least one of these 2 tests: $A_1 \cup A_2$, i.e., the (compound) event of rejecting test 1 or 2 (one or both). We may want to control the error rate of this “family” of events, not just the rate of false rejection events for individual tests, i.e., we may want to control the “family-wise” Type I error rate, α , of falsely rejecting at least one of these tests, i.e., control $\alpha = Pr(A_1 \cup A_2) = Pr(A_1) + Pr(A_2) - Pr(A_1 \cap A_2)$, i.e., the probability of making one or more false rejections out of the entire family of (2) tests.

Without knowing how the tests are related, we cannot say what $Pr(A_1 \cap A_2)$ is, except that $0 \leq Pr(A_1 \cap A_2) \leq 1$, so we can only say $\alpha = Pr(A_1 \cup A_2) = Pr(A_1) + Pr(A_2) - Pr(A_1 \cap A_2) \leq Pr(A_1) + Pr(A_2) = 2\alpha^*$. That

is, we can only say that $\alpha \leq 2\alpha^*$. So, if we conduct each individual test at, say $\alpha^* = 0.05$, we are only able to say that the probability of (at least one) false rejection is less than 0.1. We may want to give an error rate that applies to the entire family (just 2 for the moment) of tests that is smaller than this.

Thus, we might instead choose a nominal family-wise error rate α , then conduct each individual test at a rate of $\alpha^* = \alpha/2$. Thus, by conducting each individual test at $\alpha/2$, we insure that the overall, family-wise error rate is at least as small as α ; i.e., we ensure that the probability of false rejection for the entire family of tests, is at least as small as we say it is (α).

In general, if we have a family of n tests, conducting individual tests at α/n will ensure that our actual (unknown) overall, family-wise error rate is at least as small as the nominally specified α . Note that this is conservative, i.e., the actual family-wise error rate is at least as small as we say it is.

Take home message: a crude, but generally applicable, way to control the family-wise error rate at (or below) α is to conduct the n individual tests at $\alpha^* = \alpha/n$. It's too crude if n is large.

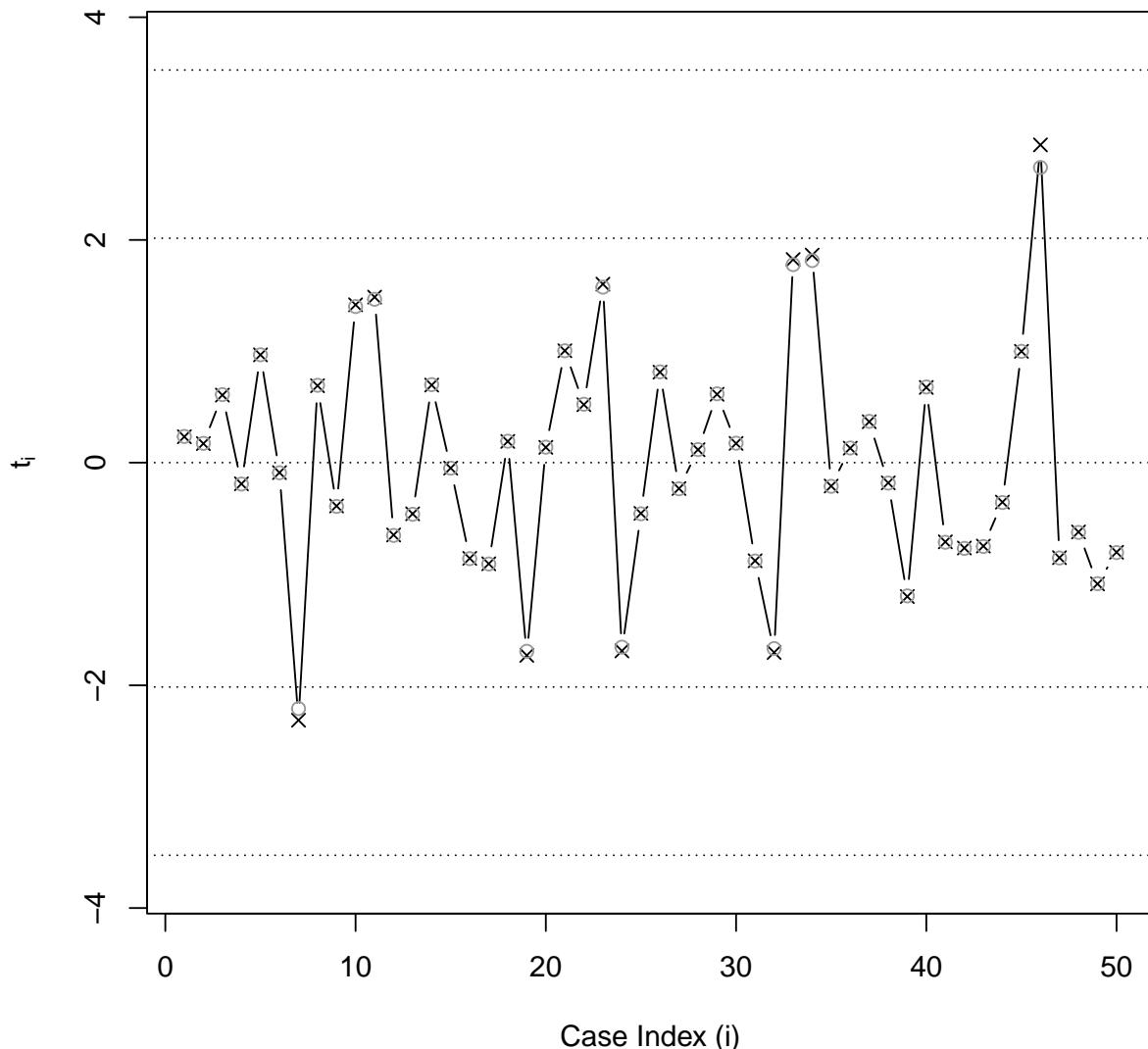
- We continue to illustrate the Bonferroni procedure for outliers using the savings data example. Our code is different than your author's, but agrees when it must.

```
> ## Savings Data Example again
> ## Studentized deleted residuals and Studentized (not deleted)
> ## residuals, shown for comparison.
> plot(ti<- rstudent(lmod), pch=4, type="b", ylim=c(-3.75,3.75),
+       xlab="Case Index (i)", ylab=expression(t[i]))
> points(rstandard(lmod), pch=1, col=grey(0.6))
> abline(h=0, lty=3)
> ## Bonferroni corrected t "critical value" for studentized
> ## deleted residuals (what are n and p?):
> (tcrit<- qt(1-0.05/(2*length(ti)), n - 1 - p))
[1] 3.5258
```

```
> abline(h=c(-tcrit, tcrit), lty=3)
> ## Uncorrected, for comparison:
> (tcrit<- qt(1-0.05/2, length(ti) - 1 - p))

[1] 2.0154

> abline(h=c(-tcrit, tcrit), lty=3)
```



```
> ## As on p. 87 of LMwR2e  
> ti[which.max(abs(ti))]
```

Zambia
2.8536

- In any case, however we have “standardized,” I am not particularly concerned about outlying Y (or X) observations in the savings data (assuming our model is correct).

Remedial Actions for Outliers

- See [Far14, p. 88] for more **points to consider about outliers**, and **what should be done about outliers** (remediation).
- In particular, we may consider **robust methods** in [Far14, §8.4].
- Note that outliers may be the most interesting and important aspect of your data! See the CFC/ozone hole example on [Far14, p. 89].
- Your textbook’s author gives an example of a dataset with multiple outliers ([Far14, p. 88 & Fig. 6.10]).
- He uses fake data to illustrate
 - (i) an outlier that does not have large leverage and does not influence the fit (much);
 - (ii) a point with large leverage but is not influential and not an outlier (by our residual diagnostic measures anyway).;
 - (iii) a point that is an outlier, with large leverage, and influential.

6.2.3 Influential Observations

- As we said, an observation (Y_i, \mathbf{X}_i) is said to be **influential** if its omission from the data set results in “large” changes in some aspect of the fitted regression function.
- Note that, if Y_i or \mathbf{X}_i is determined to be an outlier by previous diagnostics, case i may or may not be influential.
- If it’s not influential, then we might feel more confident that our model is adequate, else we might pursue remedial measures, as we’ve mentioned briefly, above.
- The procedures covered here are based on the idea of fitting the model to all observations, $i = 1, \dots, n$, then assessing the change in some “interesting” aspect of the fitted regression when omitting the i th observation from a refit.
- We cover three common diagnostic quantities to measure such changes to “interesting” quantities: DFFITS, Cook’s distance and DFBETAS.

Influence of Case i on a Single Fitted Value: DFFITS

If you don’t have some particular aspect of the regression model fit for which you want to assess influential cases, then DFFITS may be a default choice. This is not in [Far14].

- How big is the difference (“DF”) between the fitted value, \widehat{Y}_i , *with* case i , from that of the “fitted” value, $\widehat{Y}_{i(i)}$, *without* case i (“FITS”)?
- (The latter is what we called the **deleted prediction** when discussing externally Studentized (deleted) residuals.)
- That is, how much does the i “fit” change by including/excluding the i th case?

-

$$(DFFITS)_i = \frac{\hat{Y}_i - \hat{Y}_{i(i)}}{\sqrt{MSE_{(i)} h_{ii}}},$$

where, $MSE_{(i)}$ is from the fit that excludes the i th observation and h_{ii} is the i th diagonal element of the hat matrix of the \mathbf{X} matrix with all observations, as in our discussion of externally Studentized (deleted) residuals.

- Note, we standardize the difference by (almost) value of $Var(\hat{Y}_i) = \sigma^2 h_{ii}$, estimated using $MSE_{(i)}$ for σ^2 instead of using all of the data to compute MSE , which may be contaminated (influenced by) by an offending Y_i . (The hat matrix and leverage values, h_{ii} are not affected by responses, remember?)
- Once again, it appears that we must actually refit, without case i . Once again, no. See [KNNL05, Eqn. 10.10a]. No refitting necessary!
- Rule of thumb (details omitted): compare to 1 (not large n) or to $2\sqrt{p/n}$ (large n).

Cook's Distance

- Instead of just measuring the effect of case i on \hat{Y}_i , why not measure the effect of case i on all fitted values?
- **Cook's distance**, D_i does just that.

$$\begin{aligned} D_i &= \frac{\sum_{j=1}^n (\hat{Y}_j - \hat{Y}_{j(i)})^2}{pMSE} \\ &= \frac{1}{p} r_i^2 \frac{h_{ii}}{1 - h_{ii}}, \end{aligned}$$

where, evidently, r_i is the (internally) Studentized residual ([Far14, p. 84]).

- We see that it combines the effect of residuals and leverage, and that it appears similar to *DFFITS*, but measures the aggregated effect of case i on all fitted values: how do the fits, on average, change when case i is included?
- It appears as a sort of strange average, dividing by p , until we look at the motivation behind Cook's distance.
- By Result B.1,

$$\hat{\beta} \sim N(\beta, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}).$$

- Standardizing, by Result B.4 with $\mathbf{C} = \mathbf{I}$, we get a $\chi^2(p)$ random variable (assuming p parameters),

$$(\hat{\beta} - \beta)'(1/\sigma^2\mathbf{X}'\mathbf{X})(\hat{\beta} - \beta).$$

- If we divide by p and by MSE/σ^2 (or, we might think, if we estimate σ^2 by MSE and scale by $1/p$, then, by Result B.9 we have an F random variable:

$$\frac{(\hat{\beta} - \beta)'(\mathbf{X}'\mathbf{X})(\hat{\beta} - \beta)}{pMSE} \sim F(p, n - p).$$

- You may recall that level curves of a (positive definite) quadratic form give an ellipsoid. Thus,

$$\frac{(\hat{\beta} - \beta)'(\mathbf{X}'\mathbf{X})(\hat{\beta} - \beta)}{pMSE} = F(1 - \alpha, p, n - p),$$

gives an ellipsoid boundary of β values, centered at $\hat{\beta}$, whose axes are rotated and scaled by the (estimated) covariance matrix of $\hat{\beta}$, i.e., we get a $100(1 - \alpha)\%$ confidence ellipsoid for the p -dimensional β vector!

- See, e.g., the confidence region on [Far14, pp. 44-5].
- Anyway, Cook apparently used this last expression to motivate Cook's distance:

$$\frac{(\hat{\beta} - \hat{\beta}_{(i)})'(\mathbf{X}'\mathbf{X})(\hat{\beta} - \hat{\beta}_{(i)})}{pMSE},$$

which is an algebraically equivalent form of Cook's distance, D_i .

- Do you see the equivalence?
- Moreover, we now have **two different interpretations of Cook's distance**:
 1. the aggregated effect of case i on the fitted values, or
 2. the aggregated effect of case i on the fitted coefficients.
- The above development has led to the comparison of D_i to $F(p, n - p)$.
- BUT, we should not expect too severe a (scaled) difference between $\hat{\beta}$ and $\hat{\beta}_{(i)}$ to use $F(1 - 0.05, p, n - p)$ as a “critical value” (that would imply a big change in the location of the ellipse for the omission of just one observation!).
- Because $F(p, n - p)$ is not the actual reference distribution for Cook's distance (F-looking ratio), a **rule of thumb** is, instead, to view the difference as **not influential** if D_i is no bigger than about $F(0.10, p, n - p)$ or $F(0.20, p, n - p)$, but if it's bigger than $F(0.50, p, n - p)$, then we might say the i th observation is **very influential**.
- This rule of thumb is very rough: Note that we would not expect much change at all when omitting a single i from a large data set, yet the “critical” $F(p, n - p)$ doesn't change much as n gets big. Look at, e.g., `qf(p=0.15, 5, seq(10, 100, 10)-5)` in R.

Influence of Case i on Regression Coefficients: DFBETAS

- Just as DFFITS was limited to the effect of case i on the i th fitted value, we might also measure the effect of case i on each of the fitted coefficients, $\hat{\beta}_j$, individually, $j = 0, \dots, p - 1$:

$$(DFBETAS)_{j(i)} = \frac{\hat{\beta}_j - \hat{\beta}_{j(i)}}{\sqrt{MSE_{(i)}(\mathbf{X}'\mathbf{X})_{jj}^{-1}}},$$

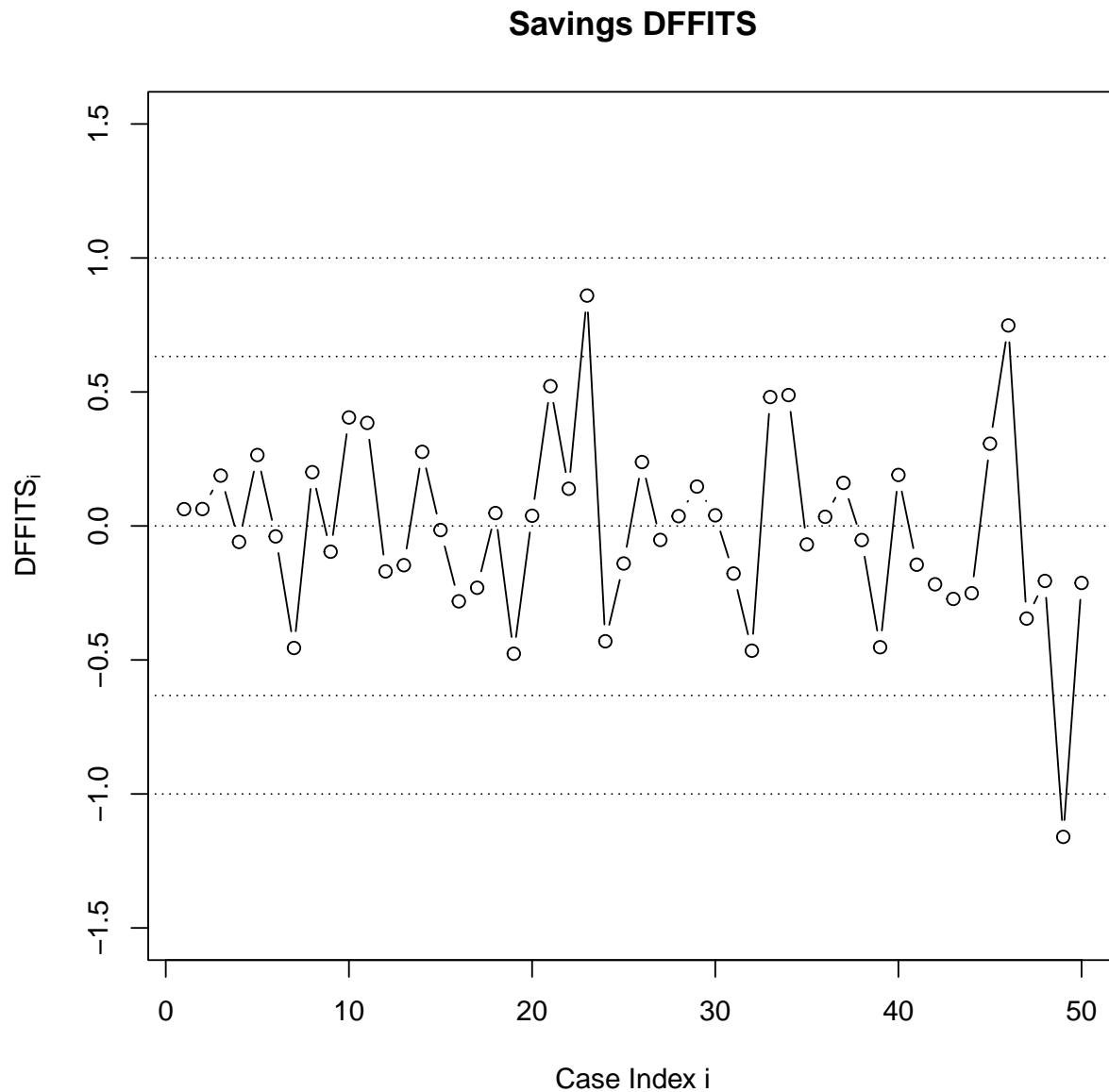
where $(\mathbf{X}'\mathbf{X})_{jj}^{-1}$ is the j th diagonal entry of $(\mathbf{X}'\mathbf{X})^{-1}$.

- This looks like a sort of univariate version of Cook's distance, which is easier to see with the latter form of Cook's distance presented above.
- **Rule of thumb** (details omitted): compare to 1 (for not large n) or to $2/\sqrt{n}$ (for large n).

Savings Data Example

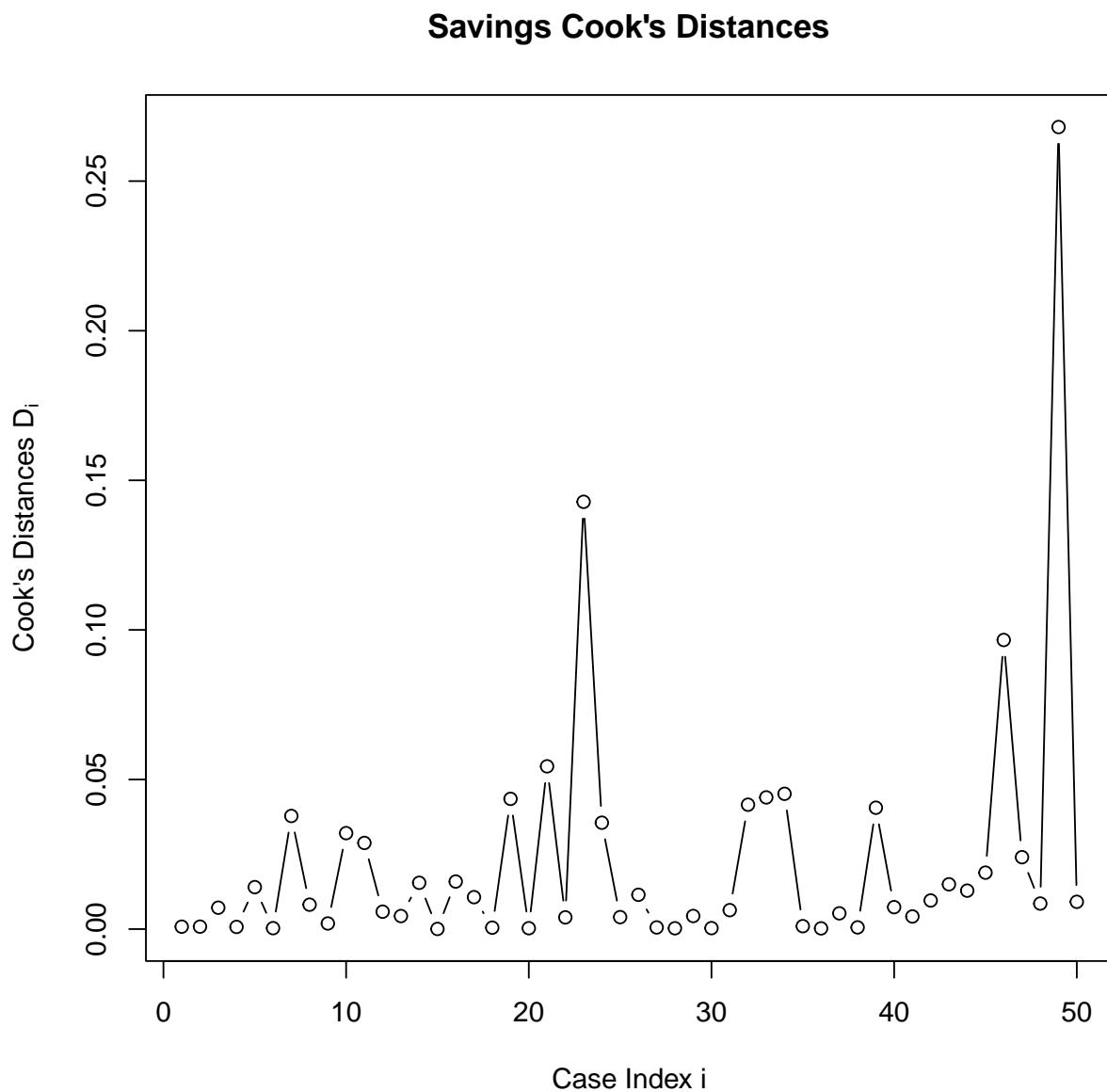
Again, DFFITS_i is influence of the i th case on the i th fitted value, so we expect to see n values of $DFFITS_i$, $i = 1, \dots, n$ against i ; I add the rule-of-thumb “influence” thresholds. (Does it make sense to connect the dots or use lines here?...)

```
> ### Savings data example: influential cases (Y,X)?: DFFITS:
> ### Assume existence of objects created in previous chunks.
> savdffits<- dffits(lmod)
> plot(savdffits, type="b", ylim=c(-1.5,1.5),
+       ylab=expression(DFFITS[i]),
+       xlab="Case Index i", main="Savings DFFITS")
> abline(h=c(0, -1, 1, -2*sqrt(p/n), 2*sqrt(p/n)), lty=3)
```



Again, **Cook's distance**, D_i is the aggregated influence of the i th case on all fitted values, and we expect to see n values of D_i , $i = 1, \dots, n$ against i ; I add the rule-of-thumb “influence” thresholds. (Does it make sense to connect the dots or use lines here?...)

```
> ### Savings data example: influential cases ( $Y, X$ )?: Cook's distances,  $D_i$ :  
> ### Assume existence of objects created in previous chunks.  
> savD<- cooks.distance(lmod)  
> plot(savD, type="b",  
+       ylab=expression(paste("Cook's Distances ", D[i], sep="")),  
+       xlab="Case Index i",  
+       main="Savings Cook's Distances")  
> abline(h=infcuts<-qf(p=c(0.1,0.2), df1=p, df2=n-p), lty=3)
```



```
> infcuts

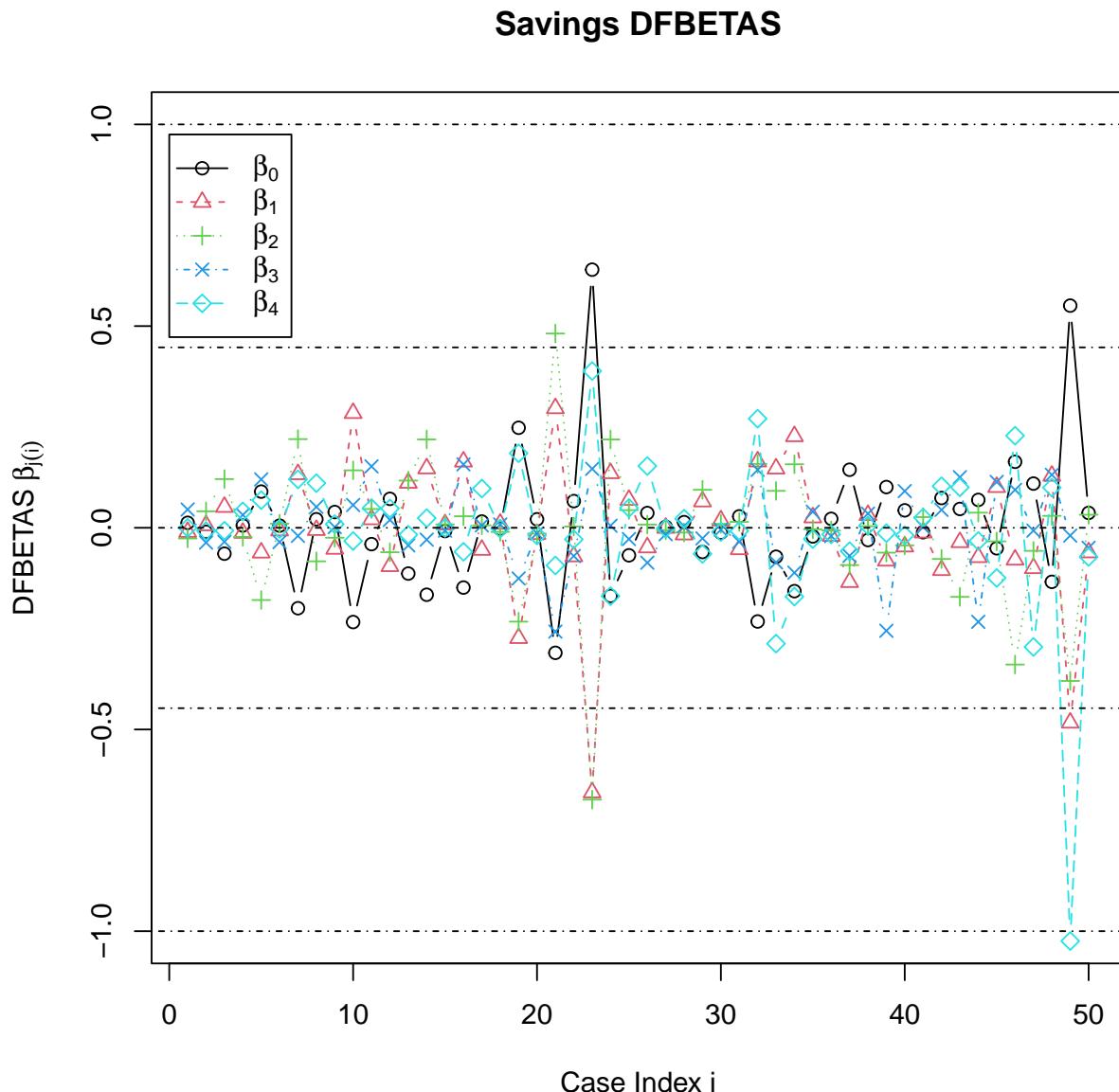
[1] 0.31729 0.46527

> tail(sort(savD), n=3)

Zambia      Japan      Libya
0.096633 0.142816 0.268070
```

Again, $DFBETAS_{j(i)}$ is the aggregated influence of the i th case on fitted coefficient, β_j , and we expect to see n values of $DFBETAS_{j(i)}$, $i = 1, \dots, n$ against i for each $j = 0, \dots, p$ (np total values); I add the rule-of-thumb “influence” thresholds. (Does it make sense to connect the dots or use lines here?...) Your textbook’s author shows $DFBETAS_{j(i)}$ for j corresponding to pop15 ([Far14, Fig 6.11]); Japan is most influential for this coefficient.

```
> ### Savings data example: influential cases (Y,X)?: DFBETAS:
> ### Assume existence of objects created in previous chunks.
> savdfbetas<- dfbetas(lmod)
> plot(savdfbetas[,1], type="b", pch=1,
+       ylab=expression(paste("DFBETAS ", beta[j][i], sep="")),
+       xlab="Case Index i",
+       main="Savings DFBETAS",
+       ylim=c(-1,1))
> lines(savdfbetas[,2], type="b", lty=2, pch=2, col=2)
> lines(savdfbetas[,3], type="b", lty=3, pch=3, col=3)
> lines(savdfbetas[,4], type="b", lty=4, pch=4, col=4)
> lines(savdfbetas[,5], type="b", lty=5, pch=5, col=5)
> abline(h=c(0,-1,1,-2/sqrt(20),2/sqrt(20)), lty=4)
> legend(0,0.975,
+         legend=expression(beta[0],beta[1],beta[2],beta[3],beta[4]),
+         pch=1:5, lty=1:5, col=1:5)
```

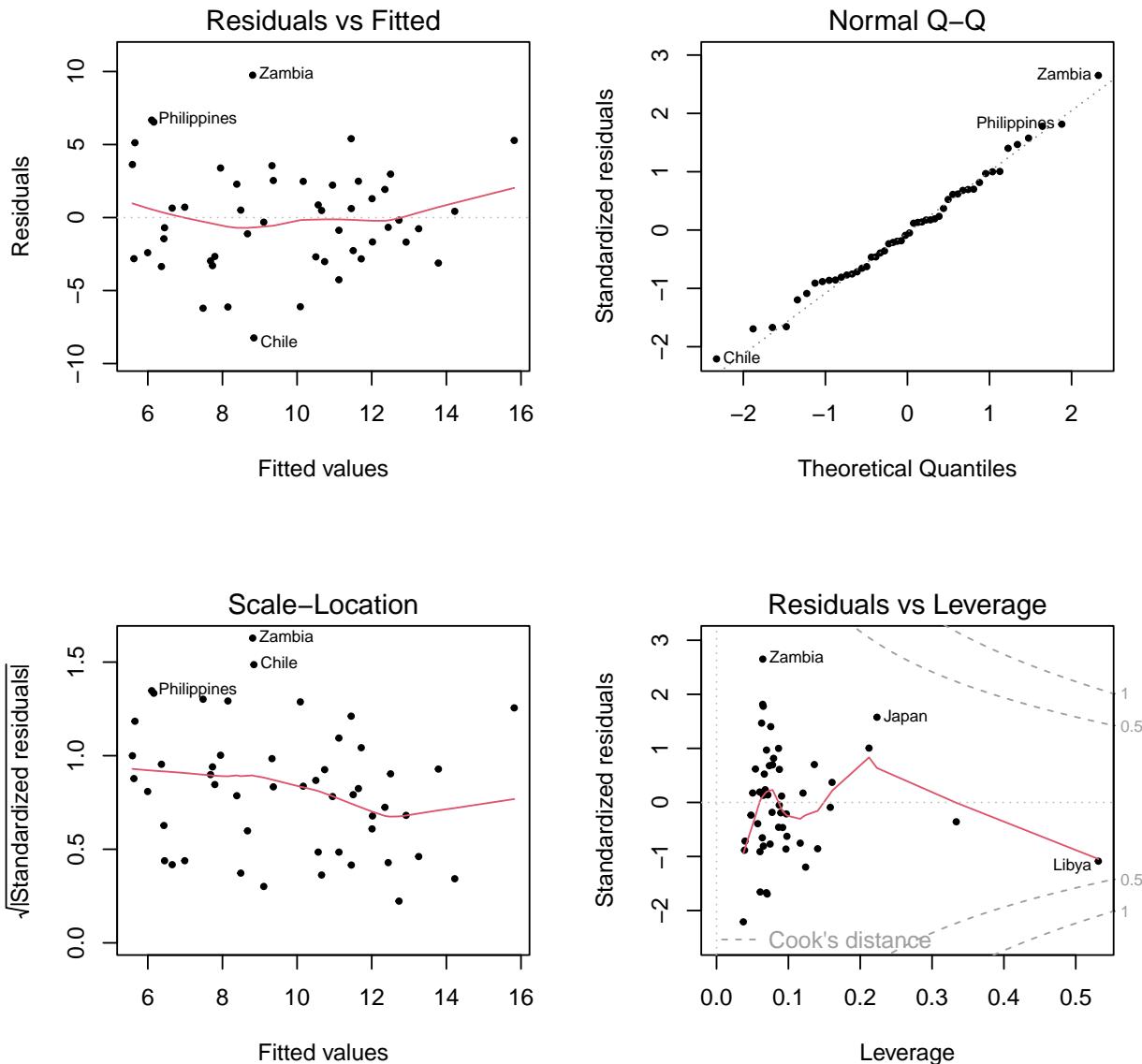


Default Diagnostic Plots

You may prefer a bit more convenience than provided by the above plots; R provides default diagnostic plots via `plot.lm`. Compare to [Far14, Fig. 6.12].

```
> ## Savings data example default diagnostic plots
> par(mfrow=c(2,2))
```

```
> plot(lmod, ask=FALSE, pch=20, cex=0.8)
```



```
> par(mfrow=c(1,1))
```

What are the main conclusions of the outlier/influential case diagnostics for the savings example? See [Far14, §6.2.3] for a bit more discussion along these lines.

Influence on Inferences of Interest

Of course, you might have particular inferences mind, and, as such, you would be interested in how *your* interesting inferences change when you omit a potentially influential outlier. See [KNNL05, p. 405–6].

6.3 Checking the Systematic Structure of the Model

Added–Variable Plots aka Partial Regression Plots

- Have we left out a significant regressor(s)?

- The **added variable plot** (aka **partial regression plot**) is a bit more sophisticated than the omitted–variable plot:
 - Plot the model residuals (this is the same as in the omitted–variable plot) against the residuals from the regression of the potential added variable on the same regressors (this is not the same as in the omitted–variable plot, which plots against the omitted variable itself).
- In other words, plot, e.g., $\widehat{\varepsilon}_i(Y|X_2, X_3, X_4) = Y_i - \widehat{Y}_i(X_2, X_3, X_4)$, the residuals from the regression of response Y_i on regressors X_2 , X_3 , and X_4 ,
- against $\widehat{\varepsilon}_i(X_1|X_2, X_3, X_4) = X_{i1} - \widehat{X}_{i1}(X_2, X_3, X_4)$, the residuals from the regression of potential added variable X_1 on X_2 , X_3 , and X_4 .
- Why do this? Why not just use the omitted–variable approach?
- We want to assess if there is there any *additional* variability in Y that a potential regression may explain **after accounting** for the variability already explained by the regressors in the model.

- In other words, we are assessing any *marginal*, or **adjusted**, importance of a potential, added variable to reduce residual variability in Y .
- And, the plots may indicate what sort (linear, quadratic, etc.) of (marginal, or “adjusted”) relationship may exist between Y and the potential added variable.
- This suggests **caution**: if you have misspecified the relationship between Y and existing regressor(s), and the existing regressors are related to the potential, added regressor, then the form of marginal relationship revealed by an added variable plot may not be appropriate.
- Notice, incidentally, that the spirit of the added variable plot is akin to the concept of the **effect** of a regressor in note chapter 5 after accounting for other regressors.

Savings Data Example

```

> ## sr regression
> sr.x2x3x4.res <- residuals(sr.x2x3x4.lm<-
+                               lm(sr ~ pop75 + dpi + ddpi,
+                               savings))
> ## pop15 regression
> x1.x2x3x4.res <- residuals(x1.x2x3x4.lm<-
+                               lm(pop15 ~ pop75 + dpi + ddpi,
+                               savings))
> par(mfrow=c(1,2))
> plot(x1.x2x3x4.res, sr.x2x3x4.res,
+       xlab="pop15 residuals", ylab="sr residuals",
+       main="Added Variable Plot for\npop15 after pop75, dpi, ddpi",
+       pch=20, cex=0.9)
> abline(h=0, lty=4)
> ## Coefficient of regression of residuals on residuals is effect
> ## for pop15 after accounting for other covariates (Section 5.1)
> coef(sr.x1.x2x3x4.lm<- lm(sr.x2x3x4.res ~ x1.x2x3x4.res))

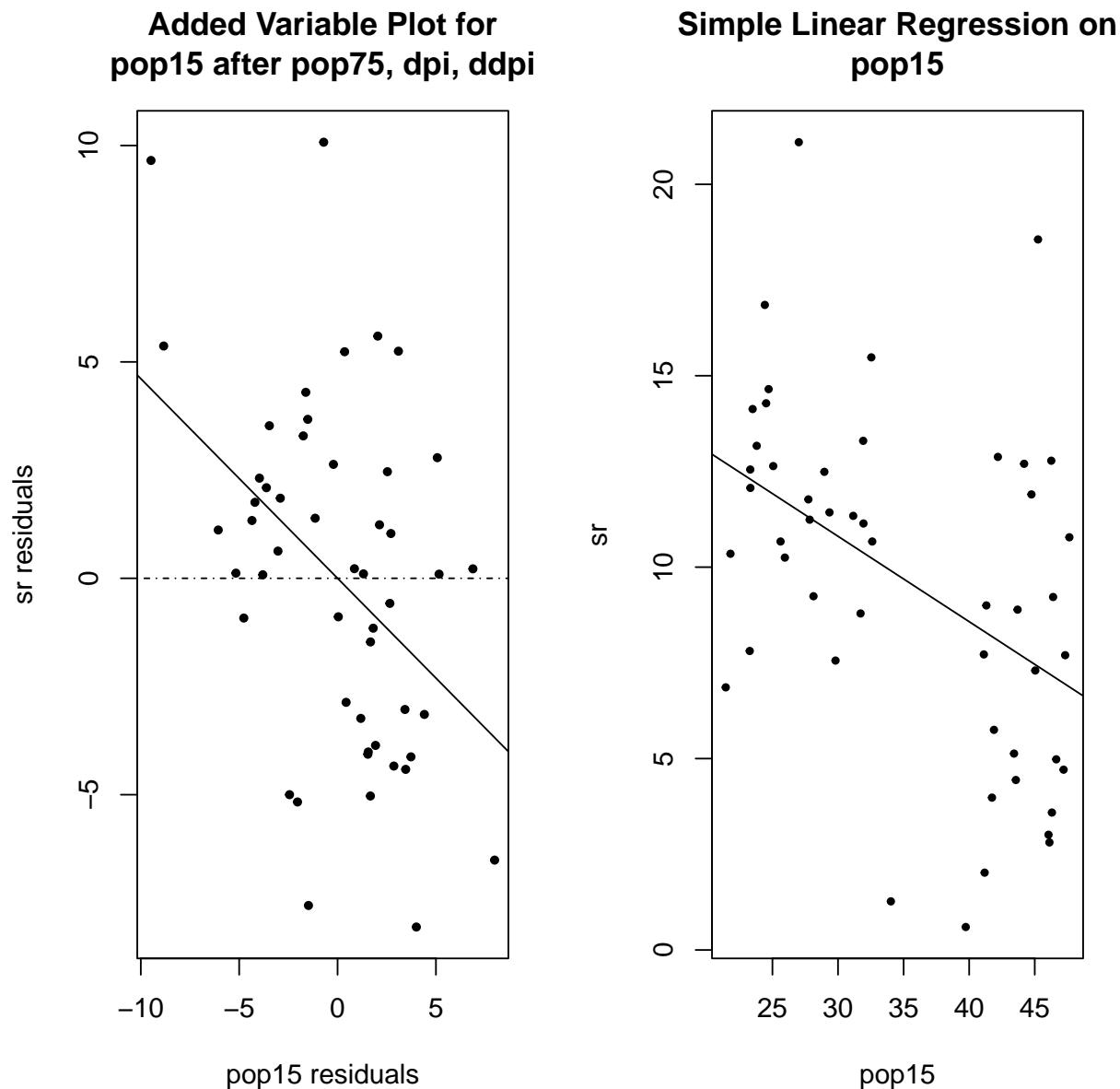
(Intercept) x1.x2x3x4.res
 9.9080e-17 -4.6119e-01

```

```
> coef(lmod<- lm(sr ~ pop15 + pop75 + dpi + ddpi,
+           savings))

(Intercept)      pop15      pop75      dpi      ddpi
28.5660865 -0.4611931 -1.6914977 -0.0003369  0.4096949

> ##
> abline(sr.x1.x2x3x4.lm)
>
> ### See also the avPlots function in the car package.
>
> ## Compare to typical xy plot with no accounting for other x
> plot(sr ~ pop15, data=savings, pch=20, cex=0.8,
+       main="Simple Linear Regression on\npop15")
> abline(lmodpop15<- lm(sr ~ pop15, data=savings))
```



```
> par(mfrow=c(1, 1))
```

Partial Residual Plot

- Not to be confused with partial *regression* plot, above.
- For the partial residual plot, instead of the typical plot of residual vs. covariate, we first remove the contribution of a covariate to the fitted value before subtracting the result from the observed value to get the, now, **partial** residual,

$$\begin{aligned} y_i - (\mathbf{x}_i^t \hat{\boldsymbol{\beta}} - \hat{\beta}_j x_{ij}) &= (y_i - \mathbf{x}_i^t \hat{\boldsymbol{\beta}}) + \hat{\beta}_j x_{ij} \\ &= \hat{\epsilon}_i + \hat{\beta}_j x_{ij}, \end{aligned}$$

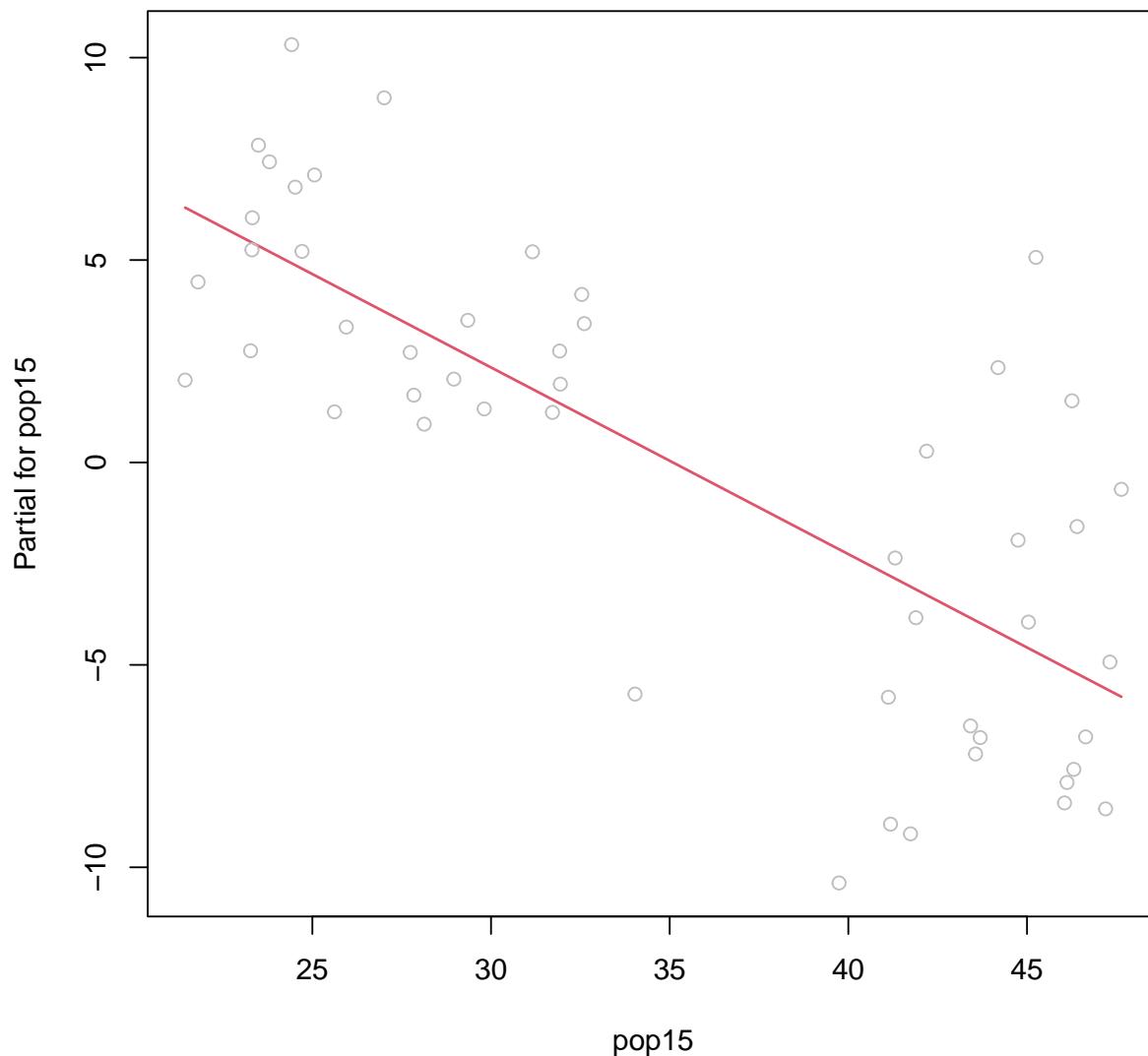
then plot the partial residual, $\hat{\epsilon}_i + \hat{\beta}_j x_{ij}$ vs. x_{ij} , $i = 1, \dots, n$. Thus, the residuals are re-centered on the omitted ‘term’. Residuals that are not spread ‘evenly’ about the (estimated) term, over values of the omitted covariate, x_{ij} , indicate lack of fit, perhaps requiring, e.g., a transformation of x_{ij} .

- **stats::termplot** centers the covariate so that the partial residual $\hat{\epsilon}_i + \hat{\beta}_j(x_{ij} - \bar{x}_j)$ has sum/average zero (over i), like ordinary residuals.
- **effects package** The effects package is convenient for plotting, well, effects (as we discussed in Chapter 5), but also for adding partial residuals to such plots. We may see this on a homework. (?)

```
> ## Savings data example partial residual plot for xj=pop15 (term 1)
> formula(lmod)

sr ~ pop15 + pop75 + dpi + ddpi

> termplot(lmod, partial.resid=TRUE, terms=1)
```



- The slope of the line in the plot is $\hat{\beta}_{pop15}$; while we do not see departures from a linear residual relationship with $pop15$ (after other covariates in the model), the partial residual plot does reveal an interesting grouping, not revealed by our previous analyses.

- Evidently, according to your textbook's author, the grouping corresponds to underdeveloped countries (proportion (percentage) of the population under 15 exceeds 35%, $\text{pop15} > 35$, "younger") and developed countries (proportion (percentage) of the population under 15 is less than 35, $\text{pop15} < 35$, "older").
- How does the relationship of savings rate with covariates change if we consider these two newly identified groups in our modeling?
- It appears that the savings rate remains significantly related to growth in disposable income (ddpi) in developed countries ($p=???$), though not as much as our previous model now that country group explains much of what previously appeared to be explained by growth, but the relationship with growth appears questionable in underdeveloped countries ($p=???$).

```
> ## Allows developed/underdeveloped (old/young) countries to have their own
> ## intercepts and slopes (compare to LMwR2d p. 95)
> savings$status<- relevel(as.factor(ifelse(savings$pop15 > 35, "young", "old")),
+                               ref="young")
>
> ## Is ddpi (growth) significant for undeveloped (young) countrys?
> summary(lmod<- lm(sr ~ status*(pop15 + pop75 + dpi + ddpi),
+                      data=savings))
```

Call:

```
lm(formula = sr ~ status * (pop15 + pop75 + dpi + ddpi), data = savings)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.589	-2.397	0.094	1.992	8.498

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-2.433969	17.224717	-0.14	0.89
statusold	26.395764	20.213436	1.31	0.20
pop15	0.273854	0.357595	0.77	0.45

```

pop75      -3.548477  2.469739  -1.44   0.16
dpi        0.000421  0.004071   0.10   0.92
ddpi       0.395474  0.236204   1.67   0.10
statusold:pop15 -0.659751  0.439579  -1.50   0.14
statusold:pop75  2.220735  2.751008   0.81   0.42
statusold:dpi   -0.000880  0.004180  -0.21   0.83
statusold:ddpi   0.488920  0.452932   1.08   0.29

```

Residual standard error: 3.63 on 40 degrees of freedom
 Multiple R-squared: 0.465, Adjusted R-squared: 0.345
 F-statistic: 3.86 on 9 and 40 DF, p-value: 0.00137

```
> confint(lmod, parm=6)
```

	2.5 %	97.5 %
ddpi	-0.081913	0.87286

```
> ## Is ddpi (growth) significant for developed (old) countries?
```

```
> gmodels::estimable(lmod, cm=c(0,0,0,0,0,1,0,0,0,1), beta0=0,
+                      conf.int=0.95)
```

	beta0	Estimate	Std. Error	t value	DF	Pr(> t)
(0 0 0 0 0 1 0 0 0 1)	0	0.88439	0.38646	2.2884	40	0.027473
		Lower.CI	Upper.CI			
(0 0 0 0 0 1 0 0 0 1)	0.10332	1.6655				

```
> ## or
```

```
> gmodels::estimable(lmod, cm=c("ddpi"=1,"statusold:ddpi"=1), beta0=0,
+                      conf.int=0.95)
```

	beta0	Estimate	Std. Error	t value	DF	Pr(> t)
(0 0 0 0 0 1 0 0 0 1)	0	0.88439	0.38646	2.2884	40	0.027473
		Lower.CI	Upper.CI			
(0 0 0 0 0 1 0 0 0 1)	0.10332	1.6655				

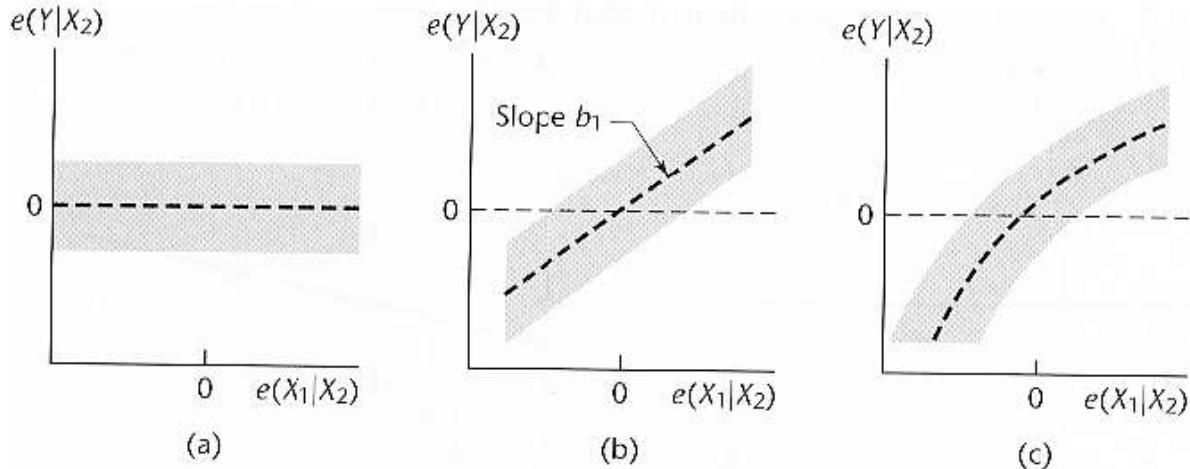
```
> ## It looks like there may be a relationship for developed (old) countries
> ## but not for underdeveloped (young) countries. That is, constant for young,
> ## something more interesting for old.
```

```
> ##savings$status01<- as.numeric(savings$status)-1
```

```
> ##summary(savings$status01)
```

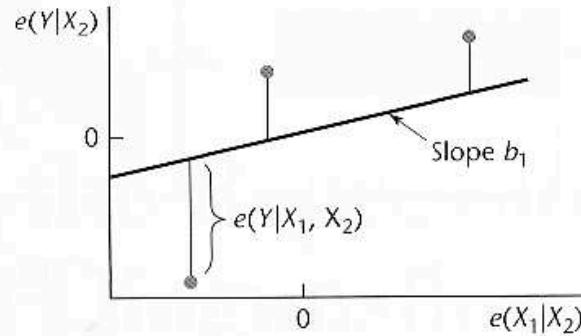
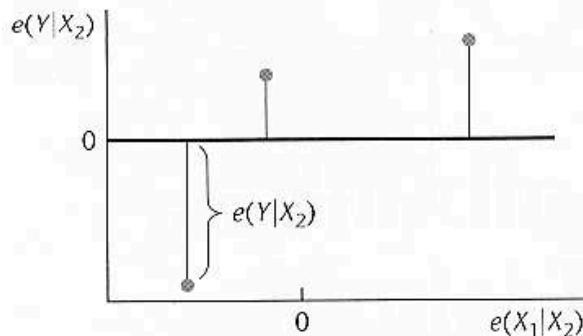
```
> ##summary(lmod<- lm(sr ~ status01 + I(status01*pop15) + I(status01*pop75) + I(status01*ddpi),
+ ##                           I(status01*ddpi), data=savings))
```

Figures 10.1 and 10.2 in [KNNL05] give a bit more insight to added-variable plots.



(a) Deviations around Zero Line
 $SSE(X_2) = \sum[e(Y_i|X_{i2})]^2$

(b) Deviations around Line with Slope b_1
 $SSE(X_1, X_2) = \sum[e(Y_i|X_{i1}, X_{i2})]^2$



6.4 Discussion

See [Far14, §6.4] for a discussion of model assumption violation in order of importance. I offer my own, corresponding interpretation here, some of which we discussed, above.

1. Getting the regression model “correct” is most important for accurate predictions and unbiased and consistent estimation of parameters to help

avoid misleading relationships. If we do not have normality, we still get consistency of regression model parameters if our mean model is correct (and if we have a simple variance model) under broadly applicable conditions.

2. Strong (positive) dependence among errors means that we have fewer than n independent pieces of information to inform our model, and our results may be overoptimistic in terms of overly narrow intervals or p-values that are too small; we need to have our (co)variance model correct to get correct standard errors and correct interval coverage rates, error rates and p-values (if only asymptotically, as we mentioned previously).
3. Non-constant variance. I clumped this with (co)variance in the previous item. (Co)variance modeling is covered in INF 512.
4. As we mentioned, larger sample size n will allow us to rely on the CLT to give us asymptotic normality for correct (though perhaps not optimal) interval coverage rates, error rates and p-values (as long as our mean and (co)variance models are good, and we have not tortured our data).
5. Correct inference also assumes data were not tortured.
6. What if prediction is the main goal, as in much of machine learning?

Lecture 7

Problems with Predictors

Contents

7.1 Errors in the Predictors	306
7.2 Change of Scale	307
7.3 Collinearity	309

READ [Far14, Chap. 7]. For time, we cover this material only briefly.

7.1 Errors in the Predictors

- **Assumed No Error in X.** As briefly mentioned in the unnumbered sections at the beginning of note chapter 5, one of the assumptions of traditional regression analysis is that the predictors (x 's) are measured without error. Note, X may still be viewed as a random variable, with added measurement error or not. If there is no measurement error, we simply proceed to infer about $Y | x$ or $E(Y | x)$, conditioning on X as we have been doing all along. (Still, see §B.7 for an additional implied assumption when conditioning on X .)
- **Error in X May Bias.** If a predictor is contaminated with measurement error then we are at **risk of obtaining biased estimators of regression model coefficients**.
- **'O'bserved is 'A'ctual Plus Error.** Your textbook's author depicts the situation as

$$\begin{aligned} y_i^O &= y_i^A + \epsilon_i && (\text{our familiar linear model}) \\ x_i^O &= x_i^A + \delta_i && (\text{measurement error model}), \end{aligned}$$

where the first equation is our familiar additive error linear model for the response, with mean,

$$y_i^A = \beta_0 + \beta_1 x_i^A,$$

(in the SLR case, e.g.) as we are familiar with (aside from new notation); the second equation expresses **measurement error**, δ_i , in the observed predictor, x_i^O .

- **Two Main Cases.** There are two main special cases of interest when a predictor has measurement error.

1. **Uncorrelated with Error.** The actual value of the predictor, X^A , is uncorrelated with measurement error, δ . In this case, **if the**

measurement error variance, $\text{Var}(\delta_i)$ is small relative to the variability of the actual values in the sample, $(\sum_i(x_i^A - \bar{x}^A)^2)$, then bias may be ignored. This sort of makes sense intuitively: a relatively small measurement error cannot mask a large range of actual predictor values hence cannot mask the relationship of the predictor to the response. In practice, **we may have some information on the size of measurement error** to help us decide if it's small relative to the range of observed/actual predictor values.

2. **Controlled Experiments.** In controlled experiments, there are **two subcases** in which measurement error may arise.

(a) **Measured** x^O . Of course, x^O varies about x^A according to measurement error, δ , so that we would observe different values of x^O for the same x^A in repeated measurements. Thus, we arrive at the **same advice as above** (item 1) about the range of the observed/actual predictor and the size of the measurement error variance.

(b) **Fixed** x^O . This is different. In this case, we fix x^O at some value. E.g., : temperature, humidity, light intensity, chemical concentration, etc. In this case, we have the same value of x^O , but x^A varies according to δ in repeated "measurements" ($x^A = x^O - \delta$, known as a **Berkson measurement error model**). It turns out that, in this fixed case, estimates of effects are **unbiased!** Magic! See the reference to Berkson (1950) in your textbook.

- See [Far14, §7.1] for methods to address measurement error when it should not be ignored. We skip these methods for time.

7.2 Change of Scale

- **Scaling and Centering.** Let's investigate what does and what does not change, and how, when we shift and scale, i.e., when we transform our

inputs x_i to

$$x_i^* = \frac{x_i - a}{b}$$

or transform outputs y_i to

$$y_i^* = \frac{y_i - a}{b}$$

for some a and $b > 0$. For example, we might change from inches to centimeters. Or, we might standardize all inputs (and outputs) by dividing by empirical standard deviations to get parameters, β_j , $j = 1, \dots, k$, on a **unitless, hence comparable, scale**. Generally, we change the **interpretation**. In cases where variables have widely different scales, rescaling can help to avoid **numerical difficulties**.

- **Scaling x Scales β_j Inversely.** For example, comparing

$$\begin{aligned} E(Y | x) &= \beta_0 + \beta_1 x \quad \text{vs.} \\ E(Y | x) &= \beta_0^* + \beta_1^* \frac{x - a}{b} \\ &= (\beta_0^* - \beta_1^* \frac{a}{b}) + \frac{\beta_1^*}{b} x \end{aligned}$$

shows that

$$\beta_1 = \frac{\beta_1^*}{b}$$

or

$$\beta_1^* = b\beta_1.$$

In short, parameters, β_j , $j = 1, \dots, k$, for unscaled inputs become $\beta_j^* = b\beta_j$ for scaled inputs. Also, obviously, the intercepts are shifted versions of one another.

- **Scaling Output Scales β_j In Same Way.** For example,

$$\begin{aligned} E(Y | x) &= \beta_0 + \beta_1 x \quad \text{vs.} \\ E\left(\frac{Y - a}{b} | x\right) &= \beta_0^* + \beta_1^* x \\ &= (a + b\beta_0^*) + b\beta_1^* x \end{aligned}$$

shows that

$$\beta_1 = b\beta_1^*$$

or

$$\beta_1^* = \frac{\beta_1}{b}.$$

In short, parameters, β_j , $j = 1, \dots, k$, for unscaled inputs become $\beta_j^* = \frac{\beta_j}{b}$ for scaled inputs. Also, obviously, the intercepts are shifted/scaled versions of one another.

- **Estimates Change Too (duh).** Of course, if the parameters change, as above, then their estimates change accordingly, too, including their interval estimates (CIs).
- **Some Things Stay the Same.** Scaling and centering, either inputs or output, does not change t or F statistics or test results, including p-values; the scale factor b cancels in these ratios. Also, the fit, as measured by R^2 does not change.

7.3 Collinearity

- **Exact Collinearity is Linear Dependence.** We have discussed exact collinearity as **linear dependence** among the columns of \mathbf{X} in our notes (§B.2.6), which can arise from sloppy data preparation. In this case, we cannot invert $\mathbf{X}^T \mathbf{X}$ to get unique LS estimates of $\boldsymbol{\beta}$. This is most often NOT what is meant by collinearity or multicollinearity in practice.
- **Near Linear Dependence.** A more common and challenging situation arises when two or more columns of \mathbf{X} are correlated but **not exactly linearly dependent**. This is typically what is meant by collinearity or, perhaps more often called **multicollinearity**.
- **Problems.** Multicollinearity can cause problems with estimation variance, $Var(\hat{\boldsymbol{\beta}}) = \sigma^2(\mathbf{X}^T \mathbf{X})$ that causes **numerical instability** and asso-

ciated **inflated standard errors** (roots of diagonal elements of $\text{Var}(\hat{\beta})$) so that estimates of β are imprecise, sometimes leading to counterintuitive signs (positive effect when we expect negative, etc.) or to large changes in estimates with only small changes in the response or as predictors are included/excluded in/from a model (with corresponding large changes to standard errors, p-values, intervals).

- **Wobbly Table Analogy.** Consider the case of $k = 2$ predictor vectors x_1 and x_2 (no intercept for illustration): example in class on the board.
- **Exacerbates Extrapolation.** Further, collinearity increases the chance of extrapolating beyond the data. (Marginal distributions of inputs no longer are good indicators of the joint distribution.)
- **Diagnostics.** Among the symptoms discussed above, collinearity is manifested as high (near 1 or -1) empirical correlation among the columns of X , high R^2 of one predictor when regressed on the remaining predictors (equivalently, a high variance inflation factor (VIF)), and a wide range of eigenvalues (high condition numbers) of $X^T X$ (or of similar matrix of centered and scaled predictors).
- **Remedial measures.** (i) **Exclude some predictors**, perhaps via formal **variable selection** procedures such as forwards, backwards or stepwise selection or via criteria such as AIC or BIC or via cross-validation ([Far14, Chap. 10]); or (ii) use **shrinkage methods** (aka **regularization**) such as principal components regression, ridge regression, partial least squares or lasso ([Far14, Chap. 11]). We may see variable selection, later. If not, then I will cover it in INF 504, which will cover shrinkage methods, too.

Lecture 8

Problems with the Error

Contents

8.3 Testing for Lack of Fit	312
---------------------------------------	-----

We only cover [Far14, §8.3] for time and because much of the remainder of this chapter and more is covered in INF 512.

8.3 Testing for Lack of Fit

- **More Than R^2 .** We know that R^2 alone is not sufficient to diagnose model fit.
- **Graphical Diagnostics.** In §6.3 (notes or textbook), we discussed plots aimed at getting our mean model correct. We'll use plots here, too, but will introduce a more formal test.
- **Prior Knowledge of Error.** As suggested at the end of a previous section on Goodness of Fit (§2.9 (notes or textbook)), we might compare our model's estimate of σ^2 to what we know about σ^2 from other similar studies. Of course, this only works if we have such prior knowledge, and, unless we know that the models from similar studies are somehow good, then comparing $\hat{\sigma}^2$ from our model to that of another model just shifts the question of model fit to the other model.
- **Familiar Approach.** Our approach follows our previous model Full v Reduced approach, aka extra-sum-of-squares approach, aka Ω v ω approach (§3.1, notes or textbook).
- **Shoehorning.** (Also, I shoehorn the test into our GLH “ $C\beta$ ” approach just to make that connection, but this approach may not seem as natural to you as the F v R approach)
- **We Require Repeated Outputs.** To perform a lack of fit (LOF) test, we need repeated observations of Y at one or more unique combinations of the x values. (Otherwise, you may approximate the approach by grouping your data by x values.) Some notation follows.
 - **Unique Inputs.** Denote c unique combinations of x levels as

$$\mathbf{x}_j, \quad j = 1, \dots, c,$$

where c is denoted as $\#\text{groups}$ in your textbook.

- **Replicate Outputs.** Denote the i th replicated Y value at level \mathbf{x}_j as

$$Y_{ij}, \quad i = 1, \dots, n_j, \quad j = 1, \dots, c,$$

where n_j is the number of observations at \mathbf{x}_j ; n_j is denoted $\#\text{replicates}_j$ in your textbook.

- **Total Number.** Then $n = \sum_{j=1}^c n_j$ is the total number of observations.
- **Group/Level Averages.** Denote $\bar{Y}_{\cdot j}$ as the (sample) averages of the n_j observations at level \mathbf{x}_j .
- **Backwards ANOVA Notation.** Note that the subscripts may be reversed compared to how you've seen a one-way ANOVA model! (If you've seen an ANOVA model before. Yes, we are essentially setting up a 1-way ANOVA model here.)

Full vs. Reduced Model Approach

Full Model

$$Y_{ij} = \mu_j + \varepsilon_{ij} \quad \varepsilon_{ij} \stackrel{\text{ind}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, n_j \quad j = 1, \dots, c$$

This is essentially the same as a linear regression model, but $E(Y_{ij} | \mathbf{x}_j)$ is **not constrained to follow a linear relationship** with \mathbf{x}_j . (BTW, it's a one-way ANOVA (cell means) model.)

Full Model Mean:

$$E(Y_{ij} | \mathbf{x}_j) = \mu_j.$$

Reduced Model Mean: Our typical linear model imposes a constraint on the means to follow a linear model (of course)

$$E(Y_{ij} | \mathbf{x}_j) = \beta_0 + \beta_1 X_{j1} + \cdots + \beta_k X_{jk}.$$

Just like other hypothesis tests, this constraint is associated with a null hypothesis, the null of no lack of fit. (In a different class, e.g., INF 512, we might perform a LOF test with a non-linear model (and likely with a likelihood ratio test instead of F test).

- **Same Sort of Testing as Before.** Now proceed as before to compare these two models (Chap. 3), but notice that we typically want to show no lack of fit, i.e., we seek to show the null by showing that the data are not inconsistent with it via a large p-value. (This may be seen as akin to a model selection perspective wherein parsimony suggests adopting a simpler model if it is not inconsistent with the model. This is opposed to seeking to reject the null in support of some interesting alternative usually associated with a more complex model than under the null.)
- **Full Model MLE/LS Estimates.** Incidentally, the least squares (or maximum likelihood if assuming normality) leads to the following estimators of the full model mean parameters:

$$\hat{\mu}_j = \bar{Y}_{\cdot j} \quad j = 1, \dots, c,$$

i.e., the estimate of the j th level mean is just the sample mean for that level \mathbf{x}_j —easy!

- **Etc.** The lack of fit procedure is also presented in [KNNL05, Sec. 3.7, 6.8] and [RS13, Chap. 8], a special case of the GLH (their GLT) approach in [KNNL05, Sec. 2.8.] (they use different notation).

Full

$$[RSS_{\Omega}] = \sum_{j=1}^c \sum_{i=1}^{n_j} (Y_{ij} - \underbrace{\bar{Y}_{\cdot j}}_{\hat{\mu}_j})^2 \equiv [SS_{pe}],$$

where SS_{pe} is **just a new name** for RSS_Ω (Chap. 3) and stands for “sum of squares pure error” (not contaminated by the lack of fit of the (full) mean model that might arise when otherwise constrained to be linear under the null of no LOF from linear model).

Reduced The reduced model is the regression model:

$$Y_{ij} = \beta_0 + \beta_1 x_{j1} + \cdots + \beta_k x_{jk} + \varepsilon_{ij} \quad \varepsilon_{ij} \stackrel{\text{ind}}{\sim} \mathcal{N}(0, \sigma^2)$$

$$RSS_\omega = \sum_{j=1}^c \sum_{i=1}^{n_j} (Y_{ij} - \hat{Y}_{ij})^2.$$

Degrees of freedom

$$n - p_\Omega = n - c \equiv df_{pe} \quad (\text{df error full model})$$

$$n - p_\omega \quad (\text{df error reduced model})$$

Test

$$H_0 : E(Y | \mathbf{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$$

$$H_a : E(Y | \mathbf{x}) \neq \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$$

$$\begin{aligned} F &= \frac{\frac{RSS_\omega - RSS_\Omega}{p_\Omega - p_\omega}}{\frac{RSS_\Omega}{n - p_\Omega}} \\ &= \frac{\frac{RSS_\omega - SS_{pe}}{c - p}}{\frac{SS_{pe}}{(n - c)}} \\ &= \frac{\frac{SS_{lf}}{(c - p)}}{\frac{SS_{pe}}{(n - c)}} \\ &= \frac{MS_{lf}}{MS_{pe}}, \end{aligned}$$

where I've added some suggestive notation, in the last two equalities, similar to what you might encounter in other textbooks.

- **As Unusual.**

$$F \sim F(c - p, n - c)$$

under the null hypothesis that the reduced model (our regression model) is true, as usual. Now proceed as usual with an F-test, but, again, this LOF test is somehow backwards in that we typically do not want to reject; we want our linear model to fit!

Example

We illustrate with the **corrosion** data set from the **faraway** package. The data consist of measurement from thirteen specimens of 90/10 Cu-Ni alloys with varying iron (Fe) content in percent. The specimens were submerged in sea water for 60 days and the weight loss due to corrosion was recorded in units of milligrams per square decimeter per day. We wish to construct a model of weight loss (y) as a function of iron content (x) so that we might, e.g., predict weight loss for some unmeasured iron content.

First, fit our SLR model.

```
> data(corrosion, package="faraway")
>
> ## Is an SLR sufficient (our ``reduced model'' in Chap. 3 jargon)
> lmod <- lm(loss ~ Fe, corrosion)
> summary(lmod)
```

Call:

```
lm(formula = loss ~ Fe, data = corrosion)
```

Residuals:

Min	1Q	Median	3Q	Max
-3.798	-1.946	0.297	0.992	5.743

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)							
(Intercept)	129.79	1.40	92.5	< 2e-16 ***							
Fe	-24.02	1.28	-18.8	1.1e-09 ***							

Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

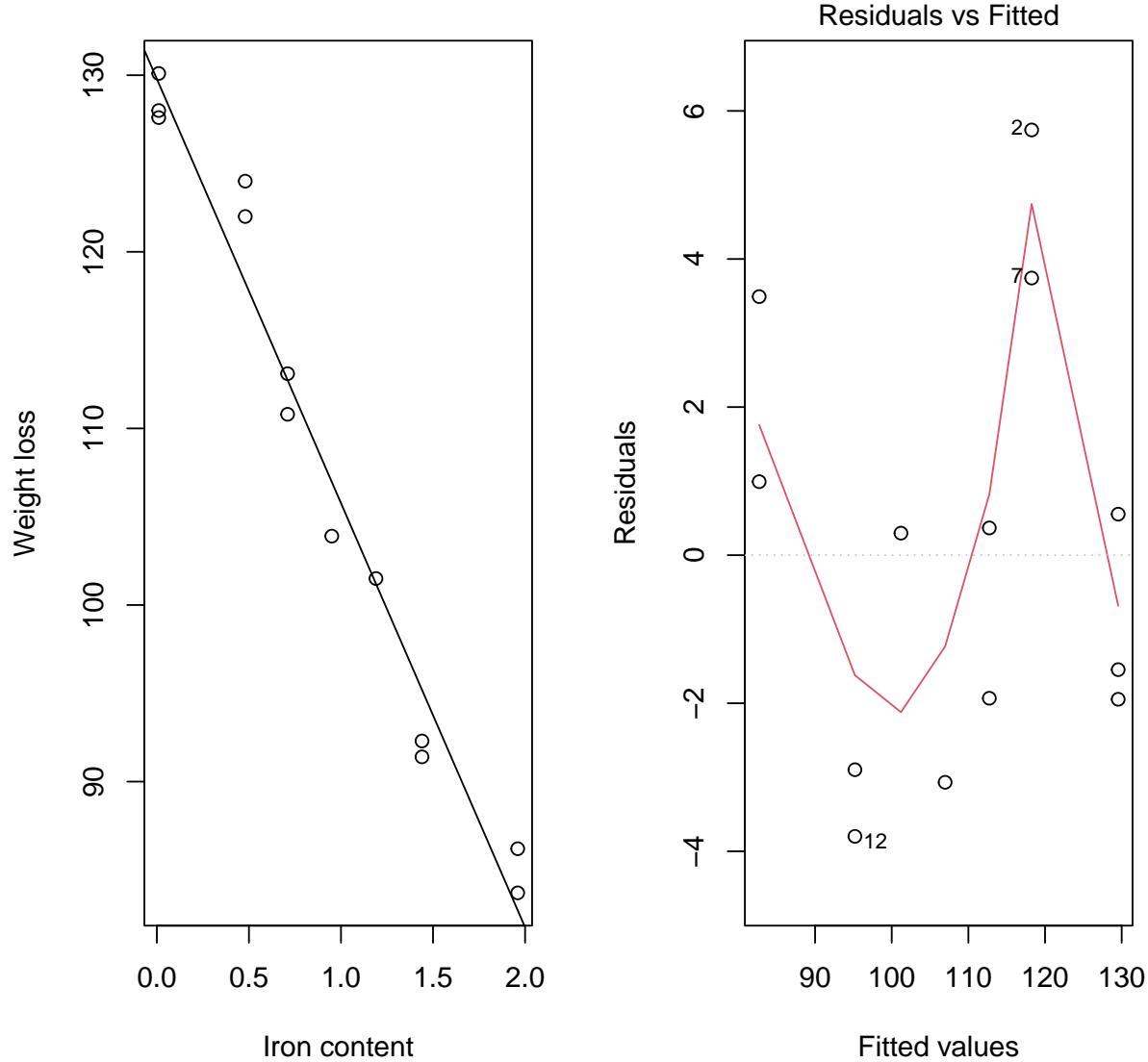
Residual standard error: 3.06 on 11 degrees of freedom

Multiple R-squared: 0.97, Adjusted R-squared: 0.967

F-statistic: 352 on 1 and 11 DF, p-value: 1.06e-09

Next, investigate LOF graphically, as in Chap. 6.

```
> par(mfrow=c(1,2))
> ## Plot. See LOF?
> plot(loss ~ Fe, corrosion, xlab="Iron content", ylab="Weight loss")
> abline(coef(lmod)) ## SLR (reduced) fit
>
> ## Residual plot (Chap 6). Now do you see LOF?
> plot(lmod, which=1)
```



```
> par(mfrow=c(1, 1))
```

- LOF is obvious from the plots, but let's perform a formal LOF test, creating a few more plots on our way.
- We fit the full ("model free") model with unconstrained means at each unique level of iron content and show how there is no LOF with this

“saturated model” that gives a separate mean parameter to each group without constraining the means (e.g., not constrained to lie on a line). While there are different ways to do this, we create an iron **factor**, so that the levels are no longer seen by R as quantitative but qualitative. As we will learn, R will fit our full model, one mean per level, but, by default, does this in a different parameterization (different parameter interpretation) than the cell means model with which we’ve introduced the LOF procedure. More when we discuss ANOVA in INF 512!

```
> ## Full Model (to be somewhat explained in class)
> corrosion$Fefact <- factor(corrosion$Fe)
> contrasts(corrosion$Fefact)

  0.48 0.71 0.95 1.19 1.44 1.96
0.01   0   0   0   0   0   0
0.48   1   0   0   0   0   0
0.71   0   1   0   0   0   0
0.95   0   0   1   0   0   0
1.19   0   0   0   1   0   0
1.44   0   0   0   0   1   0
1.96   0   0   0   0   0   1

> lmada <- lm(loss ~ Fefact, corrosion)
> summary(lmada) ## just looking (more on ANOVA models later)
```

Call:

```
lm(formula = loss ~ Fefact, data = corrosion)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.250	-0.967	0.000	1.000	1.533

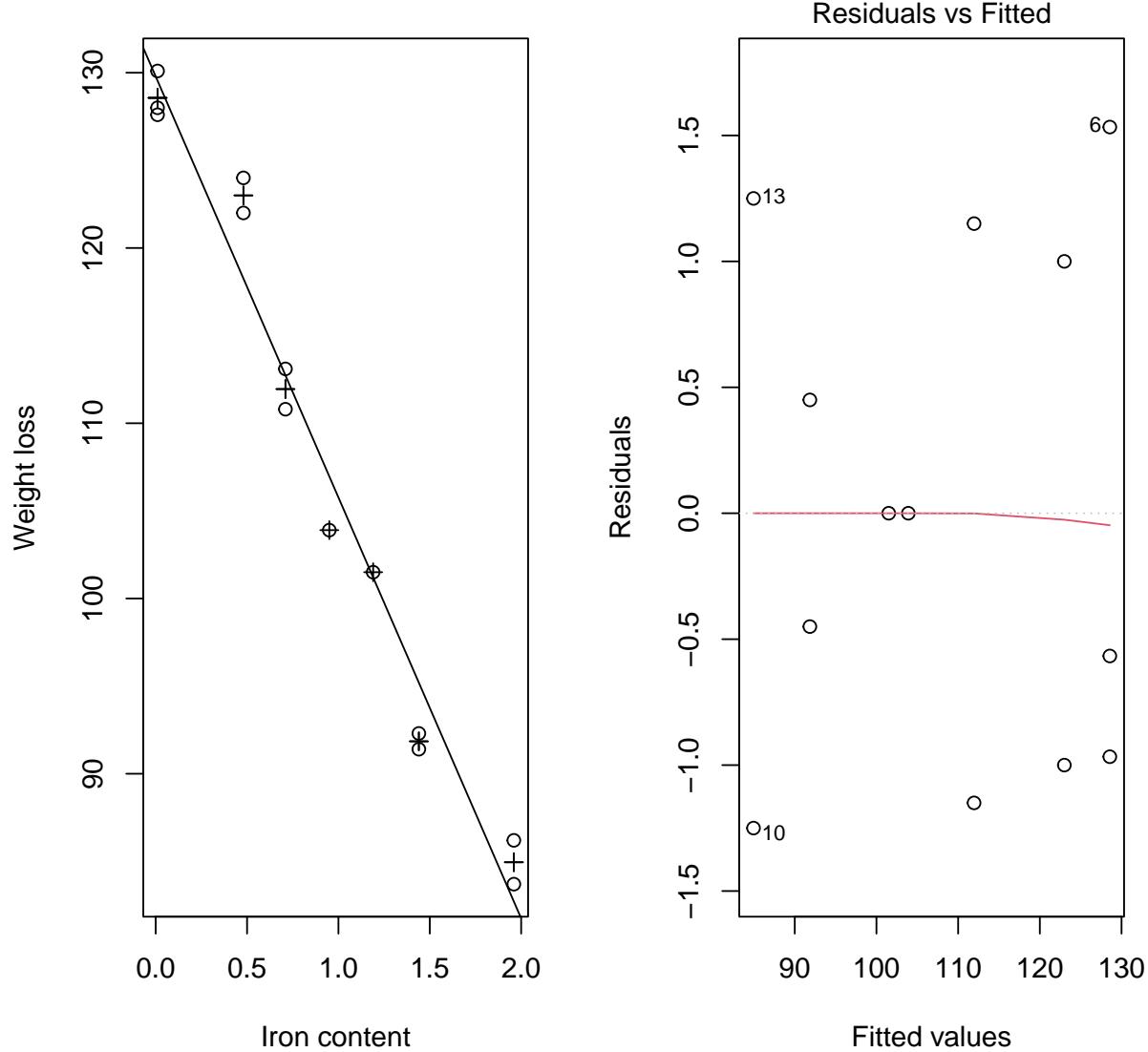
Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	128.567	0.809	158.91	4.2e-12 ***
Fefact0.48	-5.567	1.279	-4.35	0.0048 **
Fefact0.71	-16.617	1.279	-12.99	1.3e-05 ***

```
Fefact0.95   -24.667      1.618   -15.24  5.0e-06 ***  
Fefact1.19   -27.067      1.618   -16.73  2.9e-06 ***  
Fefact1.44   -36.717      1.279   -28.70  1.2e-07 ***  
Fefact1.96   -43.617      1.279   -34.10  4.2e-08 ***  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual standard error: 1.4 on 6 degrees of freedom
Multiple R-squared: 0.997, Adjusted R-squared: 0.993
F-statistic: 287 on 6 and 6 DF, p-value: 4.15e-07

```
> ## Or, more transparently, we see the estimated muj  
> ## lmmoda <- lm(loss ~ -1 + Fefact, corrosion)  
> ## summary(lmmoda)  
>  
> ## No LOF to think about with full model, but I plot anyway  
> par(mfrow=c(1,2))  
> plot(loss ~ Fe, corrosion, xlab="Iron content", ylab="Weight loss")  
> abline(coef(lmmoda)) ## SLR (reduced) fit  
> points(corrosion$Fe, fitted(lmmoda), pch=3) ## full fit  
> plot(lmmoda, which=1)
```



```
> par(mfrow=c(1,1))
```

```
> ## Formal test with our F v R (Omega v omega or ESS approach)
> anova(lmod, lmoda) ## perhaps say R v F?
```

Analysis of Variance Table

Model 1: loss ~ Fe

```
Model 2: loss ~ Fefact
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1       11 102.9
2       6  11.8  5      91.1 9.28 0.0086 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

LOF: Now What?

- Obviously, whether looking at the above plots or the formal test, we have lack of fit, so we might look for an alternative to our SLR model. But, **what model do we use?**
- **Do we use the ANOVA (full) model?** In this case, there is **no opportunity to predict weight loss between levels of iron not observed in the data**, and relatively **little opportunity to reduce the data to a parsimonious relationship that might invite further understanding** beyond just looking at the data, though we might compare (i.e., construct interesting linear combinations of) the means of the full model; see our “ $C\beta$ ” approach to this LOF problem for an example, below, which admittedly may seem relatively unnatural.
- Also, as your author notes, we shouldn’t expect to do too much better than our current SLR in terms of R^2 , at least, as the full model gives us the best R^2 that we can expect, and our SLR R^2 is not much lower.

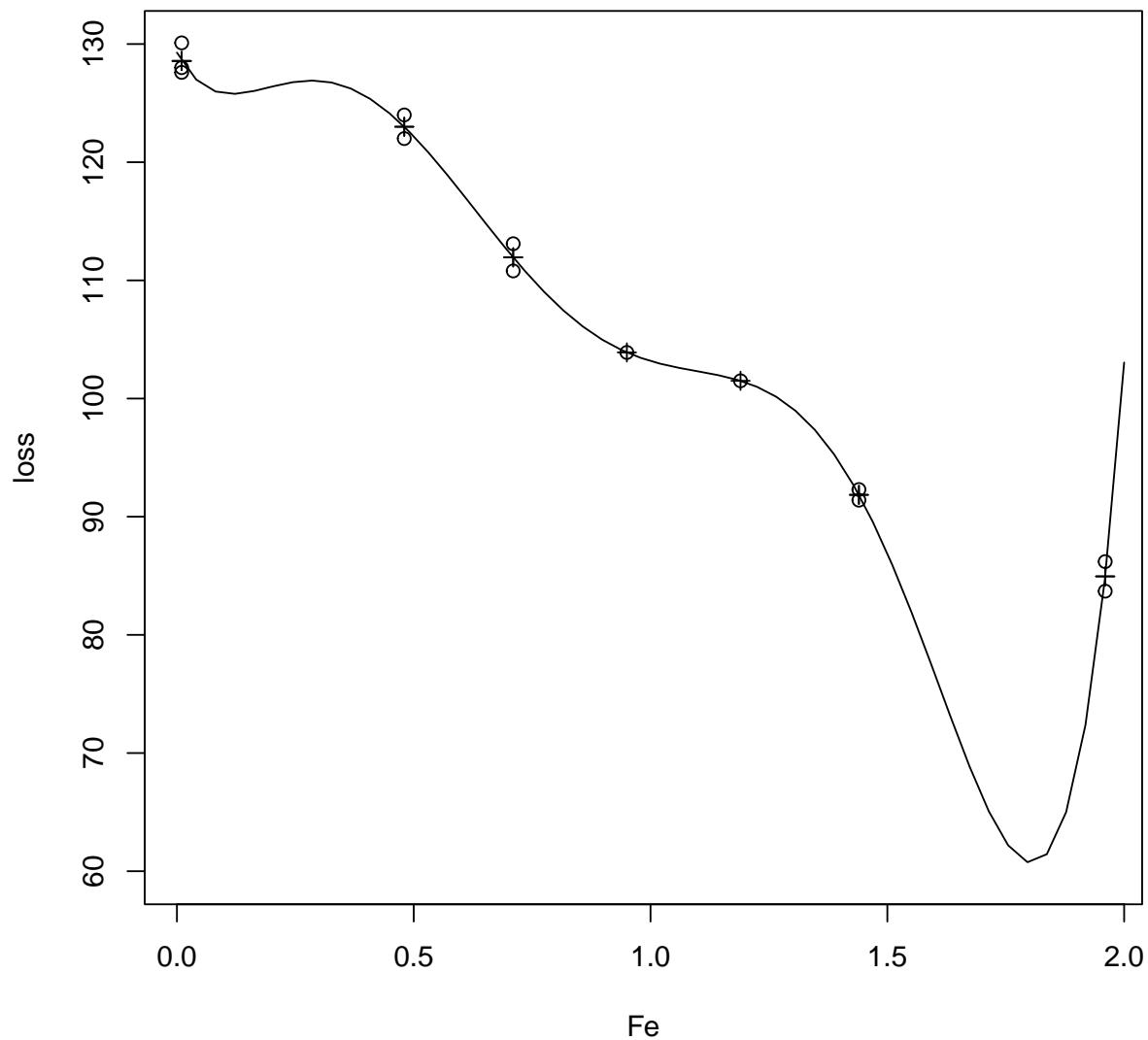
- Along the lines of desire for a better fitting model that we might use for prediction, your author fits (below) a 6th degree polynomial (some might call it a 7th order polynomial, with first order corresponding to a constant (zero degree polynomial)).

- (Without further **theoretical understanding** to suggest a model, or without **another covariate** besides iron content, I might fit a **spline model** ([Far14, Chap. 9], which we skip because splines are covered in INF 504)).
- We see (below) that a blind adherence to R^2 can result in a model that has the highest possible R^2 —same as the full model with same number of parameters—but is too variable in the sense that, if we obtained another data set in a manner comparable to the first data set, our fitted model would be a poor predictor of the new responses. That is, we are **overfitting**; more in INF 504. As your author says, “ridiculous.”

```
> ## LS fit of 6th degree polynomial
> lmodp <- lm(loss ~ poly(Fe, degree=6, raw=TRUE), corrosion)
>
> ## R^2
> summary(lmodp)$r.squared ## poly (as full as full model)
[1] 0.99653

> summary(lmoda)$r.squared ## full
[1] 0.99653
```

```
> ## R^2 isn't everything!
> ## scatterplot
> plot(loss ~ Fe, data=corrosion, ylim=c(60,130))
>
> ## add full model fitted values
> points(corrosion$Fe,fitted(lmoda),pch=3)
>
> ## add fitted polynomial
> grid <- seq(0,2,len=50)
> lines(grid,predict(lmodp, data.frame(Fe=grid)))
```



C β Approach

- Incidentally, this seems to me to be an example where the F v R approach, using the `anova` function, is simpler than our C β approach.

- What would be our C matrix necessary to constrain the (full model) means to fall on a line without constraining the slope or intercept (or overall height of the line)?
- We must have all of the “mean rises over the associated runs,” i.e., slopes, equal, without constraining the common slope value or the overall height of the line, i.e., we must have

$$\frac{\mu_2 - \mu_1}{x_2 - x_1} = \frac{\mu_3 - \mu_1}{x_3 - x_1} = \dots = \frac{\mu_7 - \mu_1}{x_7 - x_1}$$

- (Incidentally, we would have to modify this for a non-linear model, of course.)

```
> ## Cbeta approach not terribly intuitive but not bad.
>
> ## Create preliminary C matrix. Can you see the ``rises``?
> ## (Wh ensured treatment coding in a previous chunk.)
> Cmat<- matrix(c(0,1,-1,0,0,0,0,
+                   0,0,1,-1,0,0,0,
+                   0,0,0,1,-1,0,0,
+                   0,0,0,0,1,-1,0,
+                   0,0,0,0,0,1,-1),
+                   5, 7, byrow=TRUE)
> (r<- nrow(Cmat))

[1] 5

> ## Divide rises by runs
> (xj<- unique(corrosion$Fe))

[1] 0.01 0.48 0.71 0.95 1.19 1.44 1.96

> (xjminusx1<- xj - xj[1]) ## `runs' xj - x1

[1] 0.00 0.47 0.70 0.94 1.18 1.43 1.95

> for (col in 2:7) Cmat[,col] <- Cmat[,col]/xjminusx1[col]
>
> ## test
> gmodels::glh.test(lmoda, cm=Cmat, d=rep(0,r))
```

```
Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmoda, cm = Cmat, d = rep(0, r))
F = 9.2756, df1 = 5, df2 = 6, p-value = 0.008623

> ## same as F v R, as always!:
> anova(lmod, lmoda)
```

Analysis of Variance Table

```
Model 1: loss ~ Fe
Model 2: loss ~ Fefact
Res.Df   RSS Df Sum of Sq    F Pr(>F)
1      11 102.9
2      6  11.8  5      91.1 9.28 0.0086 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ## Cbeta by hand omitted (but obvious).
```

Lecture 14

Categorical Predictors

Contents

14.1 A Two-Level Factor	329
14.1.1 PTSD Data Example	329
14.2 Factors and Quantitative Predictors	342
14.2.1 PTSD Data Example	342
14.3 Interpretation with Interaction Terms	351
14.3.1 Weekly Natural Gas Consumption Data Example	351
14.4 Factors with More Than Two Levels	358
14.4.1 Cell Reference Coding (Again)	358
14.4.2 Longevity and Sexual Activity Data Example	359
14.4.3 Let's Recall Mixing and Fixing (Nothing to Do with Sex)	374
14.5 Alternative Codings of Qualitative Predictors	377

This ([Far14, Chap. 14].) is sort of a transition chapter to the next chapter on one-way ANOVA.

- **Qualitative Variables.** Categorical variables are qualitative, not quantitative, not numeric.
- **Synonyms:** factor, categorical variable, qualitative variable, grouping variable.
- **Levels.** The possible values of a categorical variable are often referred to as its levels.
- **Nominal or Ordinal.** Sometimes, categorical variables are refined into subcategories of nominal (name only) or ordinal (some order implied). We do not use the ordinality of categorical variables in this class.
- **Examples.** For example, a categorical variable X may characterize **color**, with levels of red, green and blue (a nominal variable); **gender**, male and female (nominal); **height class**: short, average, tall (ordinal); **motivation treatment**: extrinsic, intrinsic (nominal) (as in a previous example); **study guide**: A, B (as in a homework); **voting technology** digital, hand (nominal) (as in a previous example). (I use “ X ” because we will restrict our attention to categorical predictors/covariates; we will maintain normality for response, Y).
- **Numerical Coding.** A factor’s levels are assigned numerical codes for computation in our models. A few different **coding schemes** are commonly used, each associated with a particular **parameterization**, i.e., parameter **interpretation**.
- **Examples.** **study guides**: 0 (A), 1 (B); **voting technology**: 0 (digital), 1 (hand).
- **Estimation, inference, diagnostics.** Essentially, **all of our previous material for linear models still applies**.

- **Seen a Bit Before, Now More Thoroughly.** Now, we treat categorical predictors more thoroughly before getting to analysis of variance (**ANOVA**), a method featuring categorical predictor variables, in subsequent chapters.

14.1 A Two-Level Factor

14.1.1 PTSD Data Example

- **Observational Factor 2 Levels.** We illustrate with data from $n = 76$ women, 45 reporting childhood sex abuse and 31 reporting no sex abuse (csa, a factor with two levels).
- **Response and Additional Covariate.** The women were measured for post-traumatic stress disorder (ptsd; response) and childhood physical abuse (cpa; covariate), the latter two variables on standardized numerical scales.

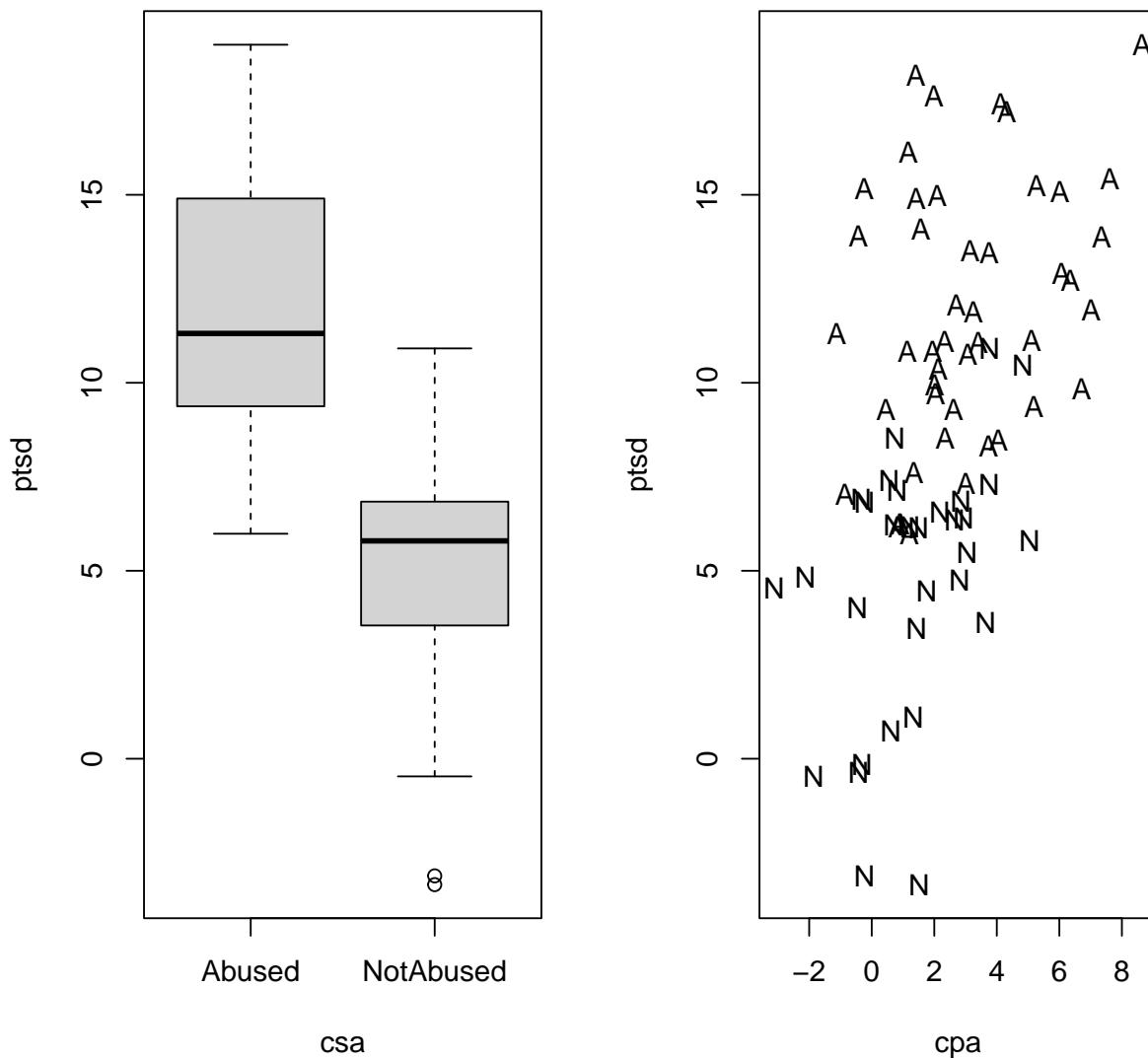
```
> ## A quick look/summary:  
> library(faraway)  
> data(sexab, package="faraway")  
> head(sexab); tail(sexab)  
  
      cpa     ptsd     csa  
1  2.04786  9.7136 Abused  
2  0.83895  6.1693 Abused  
3 -0.24139 15.1593 Abused  
4 -1.11461 11.3128 Abused  
5  2.01468  9.9538 Abused  
6  6.71131  9.8388 Abused  
      cpa     ptsd     csa  
71 -0.31402 -0.14339 NotAbused  
72  2.17626  6.57281 NotAbused
```

```
73 -0.23208 -3.11622 NotAbused
74 -1.85753 -0.46996 NotAbused
75  2.85253  6.84304 NotAbused
76  0.81138  7.12918 NotAbused

> by(sexab,sexab$csa,summary)

sexab$csa: Abused
    cpa          ptsd          csa
Min. :-1.11   Min. : 5.98   Abused :45
1st Qu.: 1.41  1st Qu.: 9.37  NotAbused: 0
Median : 2.63  Median :11.31
Mean   : 3.08  Mean   :11.94
3rd Qu.: 4.32  3rd Qu.:14.90
Max.   : 8.65  Max.   :18.99
-----
sexab$csa: NotAbused
    cpa          ptsd          csa
Min. :-3.12   Min. :-3.35   Abused : 0
1st Qu.:-0.23  1st Qu.: 3.54  NotAbused:31
Median : 1.32  Median : 5.79
Mean   : 1.31  Mean   : 4.70
3rd Qu.: 2.83  3rd Qu.: 6.84
Max.   : 5.05  Max.   :10.91
```

```
> ## Exploratory plots
> par(mfrow=c(1,2))
> plot(ptsd ~ csa, sexab)
> plot(ptsd ~ cpa, pch=as.character(csa), sexab)
```



```
> par(mfrow=c(1,1))
```

Try a 2-Sample t-test from “STAT 101”

- **Do Means Differ Across Groups?** We may test the null hypothesis

$$H_0 : \mu_1 = \mu_2$$

with a “STAT 101” (pooled) t -test.

- (We’ll use our usual tools and notation, shortly.)

```
> ## pooled variance t-test (assume constant variance across csa levels)
> (ttst<- t.test(ptsd ~ csa, data = sexab, var.equal = TRUE))
```

Two Sample t-test

```
data: ptsd by csa
t = 8.94, df = 74, p-value = 2.2e-13
alternative hypothesis: true difference in means between group Abused and group NotAbuse
95 percent confidence interval:
 5.6302 8.8603
sample estimates:
 mean in group Abused mean in group NotAbused
      11.9411              4.6959
```

General Linear Model Formulation

- This test fits into our general linear model formulation (§2.1). How?
- **Code the levels of the factor, csa , with numerical **dummy variables** (aka **indicator** variables or **incidence** variables)**

$$x_j = \begin{cases} 0 & \text{is not level } j \\ 1 & \text{is level } j \end{cases},$$

$j = 1, \dots, f$, where $f = 2$ levels for the csa factor.

- (NOTE: Your textbook’s author uses coded variables d_1 and d_2 , for “d”ummy, evidently. I maintain “ x ” notation for consistency with our typical linear model formulation.)

```

> d1 <- ifelse(sexab$csa == "Abused", 1, 0) ## x1
> d2 <- ifelse(sexab$csa == "NotAbused", 1, 0) ## x2
> lmod <- lm(ptsd ~ d1 + d2, data = sexab) ## bad R coding practice (why?)
> (lmod.sum<- summary(lmod))

Call:
lm(formula = ptsd ~ d1 + d2, data = sexab)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.045 -2.312  0.095  2.164  7.051 

Coefficients: (1 not defined because of singularities)
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  4.696     0.624    7.53  1.0e-10 ***
d1          7.245     0.811    8.94  2.2e-13 ***
d2          NA        NA      NA      NA      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.47 on 74 degrees of freedom
Multiple R-squared:  0.519, Adjusted R-squared:  0.513 
F-statistic: 79.9 on 1 and 74 DF,  p-value: 2.17e-13

> ## notice, btw, t^2 = F (and pvalues same, too)
> round(ttst$statistic^2, 10) == round(lmod.sum$fstatistic["value"], 10)

      t
TRUE

```

- Why the “NA” (missing values) in the above output?
- We have (inadvertently) created **redundancy/dependency in our model's X matrix**; it is **not full rank** (see §B.2.6) so $\mathbf{X}^t \mathbf{X}$ cannot be inverted for estimation (Chap. 2) without further action.

```
> head(xmat<- model.matrix(lmod)); tail(xmat)

(Intercept) d1 d2
1           1  1  0
2           1  1  0
3           1  1  0
4           1  1  0
5           1  1  0
6           1  1  0

(Intercept) d1 d2
71          1  0  1
72          1  0  1
73          1  0  1
74          1  0  1
75          1  0  1
76          1  0  1

> all(xmat[,1] == xmat[,2] + xmat[,3]) ## <-- linear dependency
[1] TRUE
```

- R throws out dummy d_2 (x_2) to achieve non-redundancy in the \mathbf{X} matrix.
- **Side Note: Model Not Identifiable.** Relatedly, notice how regression model parameters of our linear model, as coded with our two dummy variables, d_1 and d_2 (x_1 and x_2), are not identifiable. (add/subtract a constant c to

$$\begin{aligned} E(Y | x_1, x_2) &= \beta_0 + x_1\beta_1 + x_2\beta_2 \\ &= (\beta_0 + c) + x_1(\beta_1 - c) + x_2(\beta_2 - c) \quad (\text{why?}) \\ &\equiv \beta_0^* + x_1\beta_1^* + x_2\beta_2^* \end{aligned}$$

(A picture on the board might help here.) That is, we get the same mean value, $E(Y | x_1, x_2)$, (hence same normal distribution) despite different parameter values. In other words, even if we knew the mean (or entire distribution), it cannot help us to identify uniquely a single parameter β , so how can we expect to estimate a unique β when we just have data?

Eliminate Redundancy (Manually)

- With redundant columns of \mathbf{X} , one solution is to throw out predictor (dummy) variable, d_1 , for example. (R threw out d_2 in response to our manual coding.)
- This essentially amounts to what is known as **treatment coding** or **cell reference coding** (or **corner point coding** when we have more than one factor); the level/dummy variable that is thrown out is the reference level; more later.

```
> ## Throw out d1 (x1) to get  $E(y | x) = b_0 + b_1*x_2$ 
> lmod <- lm(ptsd ~ d2, data = sexab)
> summary(lmod)

Call:
lm(formula = ptsd ~ d2, data = sexab)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.045 -2.312  0.095  2.164  7.051 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 11.941     0.518   23.07 < 2e-16 ***
d2          -7.245     0.811   -8.94  2.2e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.47 on 74 degrees of freedom
Multiple R-squared:  0.519, Adjusted R-squared:  0.513 
F-statistic: 79.9 on 1 and 74 DF,  p-value: 2.17e-13
```

```
> ## Now, no redundancy, of course.
> head(model.matrix(lmod)); tail(model.matrix(lmod))
```

```
(Intercept) d2
1           1  0
2           1  0
3           1  0
4           1  0
5           1  0
6           1  0
(Intercept) d2
71          1  1
72          1  1
73          1  1
74          1  1
75          1  1
76          1  1
```

- Or, perhaps we throw out the column of 1's to omit the overall mean response level.
- This essentially amounts to what is known as **cell means coding**; requires additional consideration with more than one factor; more later, perhaps.

```
> ## Forced cell means coding
> lmod <- lm(ptsdf ~ d1 + d2 - 1, data = sexab)
> ## Or, equivalently
> ## lmod <- lm(ptsdf ~ d1 + d2 + 0, data = sexab)
> ## BEWARE OVERALL F TEST AND R^2!
> summary(lmod)
```

Call:
`lm(formula = ptsdf ~ d1 + d2 - 1, data = sexab)`

Residuals:

Min	1Q	Median	3Q	Max
-8.045	-2.312	0.095	2.164	7.051

```
Coefficients:
  Estimate Std. Error t value Pr(>|t|)
d1     11.941     0.518   23.07 <2e-16 ***
d2      4.696     0.624    7.53  1e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.47 on 74 degrees of freedom
Multiple R-squared:  0.888, Adjusted R-squared:  0.885
F-statistic: 294 on 2 and 74 DF,  p-value: <2e-16
```

```
> ## See cell means coding in X matrix
> head(model.matrix(lmod)); tail(model.matrix(lmod))

  d1 d2
1  1 0
2  1 0
3  1 0
4  1 0
5  1 0
6  1 0
  d1 d2
71  0 1
72  0 1
73  0 1
74  0 1
75  0 1
76  0 1
```

Let R Take Care of Redundancy More Seamlessly

- Of course, we might expect R to do something sensible without us having to deal directly with coding and redundancy. **But, we must know what R is doing in order to interpret results correctly.**
- By default, we may think of R as performing dummy variable coding, as we did, above, then omitting the first dummy variable (or simply does not

create it). (NOTE: when we coded d_1 and d_2 (our x_1 and x_2) directly, by hand, above, R threw out d_2 , but when we let R do the coding itself, the default R behavior is to throw out the 1st level, i.e., omit d_1 .)

- In this case, the **first level** of the factor becomes a **reference level** or **reference treatment** or **corner point** (the latter name makes more sense in the presence of more than one factor).
- SAS users: SAS omits the **last** dummy variable so that the **last** level is the reference level.
- How do we know what R's default behaviour will be?

```
> ## First, we should ensure that R sees factors as factors (yep. good.):
> data.class(sexab$csa)

[1] "factor"

> levels(sexab$csa)

[1] "Abused"     "NotAbused"

> ## Now, look at ``contrasts'' which will tell us the coding R will use.
>
> ## What is the global contrasts setting?
>getOption("contrasts")

      unordered          ordered
"contr.treatment"      "contr.poly"

> ## Any contrasts set for the particular factor at hand?
> ## (No. BTW, this would override the global contrasts setting.)
> attr(sexab$csa, which='contrasts')

NULL

> ## Next says R will code a column in X indicating "NotAbused" with a 1
> ## (0 for Abused (reference level)):
> contrasts(sexab$csa)
```

```
NotAbused  
Abused          0  
NotAbused       1  
  
> ## See X matrix in subsequent chunk.
```

```
> lmod <- lm(ptsd ~ csa, data = sexab)  
> summary(lmod)  
  
Call:  
lm(formula = ptsd ~ csa, data = sexab)  
  
Residuals:  
    Min      1Q  Median      3Q     Max  
-8.045 -2.312  0.095  2.164  7.051  
  
Coefficients:  
            Estimate Std. Error t value Pr(>|t|)  
(Intercept)  11.941     0.518   23.07 < 2e-16 ***  
csaNotAbused -7.245     0.811   -8.94 2.2e-13 ***  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 3.47 on 74 degrees of freedom  
Multiple R-squared:  0.519, Adjusted R-squared:  0.513  
F-statistic: 79.9 on 1 and 74 DF,  p-value: 2.17e-13
```

```
> Xmat<- model.matrix(lmod)  
> head(Xmat)  
  
(Intercept) csaNotAbused  
1             1             0  
2             1             0  
3             1             0  
4             1             0  
5             1             0  
6             1             0  
  
> tail(Xmat)
```

	(Intercept)	csaNotAbused
71	1	1
72	1	1
73	1	1
74	1	1
75	1	1
76	1	1

- **A Natural Reference Level?** Sometimes, one level may seem to be a natural reference level compared to other levels (e.g., placebo, standard treatment, etc.).
- **Compare Abused to NotAbused?** Perhaps we might think more naturally of NotAbused as a reference level, rather than Abused.
- **Know Your Levels.** Note, we may know that R throws out the first level, but we should know what that level is! (not too difficult to see)

```
> ## Change reference level
> sexab$csa <- relevel(sexab$csa, ref="NotAbused")
> ## Now we see R will code a column in X indicating "Abused" with a 1
> ## (0 for NotAbused (reference level)):
> contrasts(sexab$csa)

      Abused
NotAbused     0
Abused        1

> ## Compare this output to the previous, before changing the reference
> ## level.
> lmod <- lm(ptsdb ~ csa, sexab)
> summary(lmod)

Call:
lm(formula = ptsdb ~ csa, data = sexab)
```

Residuals:

Min	1Q	Median	3Q	Max
-8.045	-2.312	0.095	2.164	7.051

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)							
(Intercept)	4.696	0.624	7.53	1.0e-10 ***							
csaAbused	7.245	0.811	8.94	2.2e-13 ***							

Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

Residual standard error: 3.47 on 74 degrees of freedom

Multiple R-squared: 0.519, Adjusted R-squared: 0.513

F-statistic: 79.9 on 1 and 74 DF, p-value: 2.17e-13

```
> head(model.matrix(lmod))
```

	(Intercept)	csaAbused
1	1	1
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1

```
> tail(model.matrix(lmod))
```

	(Intercept)	csaAbused
71	1	0
72	1	0
73	1	0
74	1	0
75	1	0
76	1	0

14.2 Factors and Quantitative Predictors

14.2.1 PTSD Data Example

Four Different Models. Your textbook's author considers four models (I give in to his d notation a bit):

1. **Same linear relationship** with cpa (quantitative) for both levels of csa (factor):

$$y = \beta_0 + \beta_1 x + \epsilon$$

where x is cpa (i.e., csa is not in the model).

2. **No linear relationship** with cpa, but **different mean PTSD values for the different groups**:

$$y = \beta_0 + \beta_1 d + \epsilon,$$

where d indicates “Abused.” We already considered this model, above (after changing the reference level to notAbused (verify this)).

3. **Separate lines with the same slope** for each group (i.e., intercepts differ but slopes do not):

$$y = \beta_0 + \beta_1 x + \beta_2 d + \epsilon$$

4. **Separate lines, different intercepts, different slopes:**

$$y = \beta_0 + \beta_1 x + \beta_2 d + \beta_3 x d + \epsilon.$$

- This is the fullest model considered here.
- The xd model term is the product of cpa (x) and the dummy variable, d , indicating Abused (Again, we changed the reference level to NotAbused.)
- The xd term is an example of an **interaction**, generically, e.g., $x_1 x_2$. We briefly alluded to a problem with our “Simple Meaning” of a parameter when an associated predictor is also involved in an interaction term (§5.1).
- We will attempt to illustrate/clarify how an interaction term affects the meaning/interpretation of regression model parameters.

```

> ## Model (4): Different intercepts and slopes (fullest)
> lmod4 <- lm(ptsd ~ cpa+csa+cpa:csa, data=sexab)
>
> ## Or, slightly more shorthandedly
> ## lmod4<- lm(ptsd ~ cpa*csa, data=sexab)
> summary(lmod4)

Call:
lm(formula = ptsd ~ cpa + csa + cpa:csa, data = sexab)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.200 -2.531 -0.181  2.774  6.975 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.696     0.711   5.20  1.8e-06 ***
cpa          0.764     0.304   2.51   0.014 *  
csaAbused    6.861     1.075   6.38  1.5e-08 ***
cpa:csaAbused -0.314     0.368  -0.85   0.397    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.28 on 72 degrees of freedom
Multiple R-squared:  0.583, Adjusted R-squared:  0.565 
F-statistic: 33.5 on 3 and 72 DF,  p-value: 1.13e-13

> ## Quick look at coding in X
> head(model.matrix(lmod4))

  (Intercept)      cpa csaAbused cpa:csaAbused
1           1  2.04786         1       2.04786
2           1  0.83895         1       0.83895
3           1 -0.24139         1      -0.24139
4           1 -1.11461         1      -1.11461
5           1  2.01468         1       2.01468
6           1  6.71131         1       6.71131

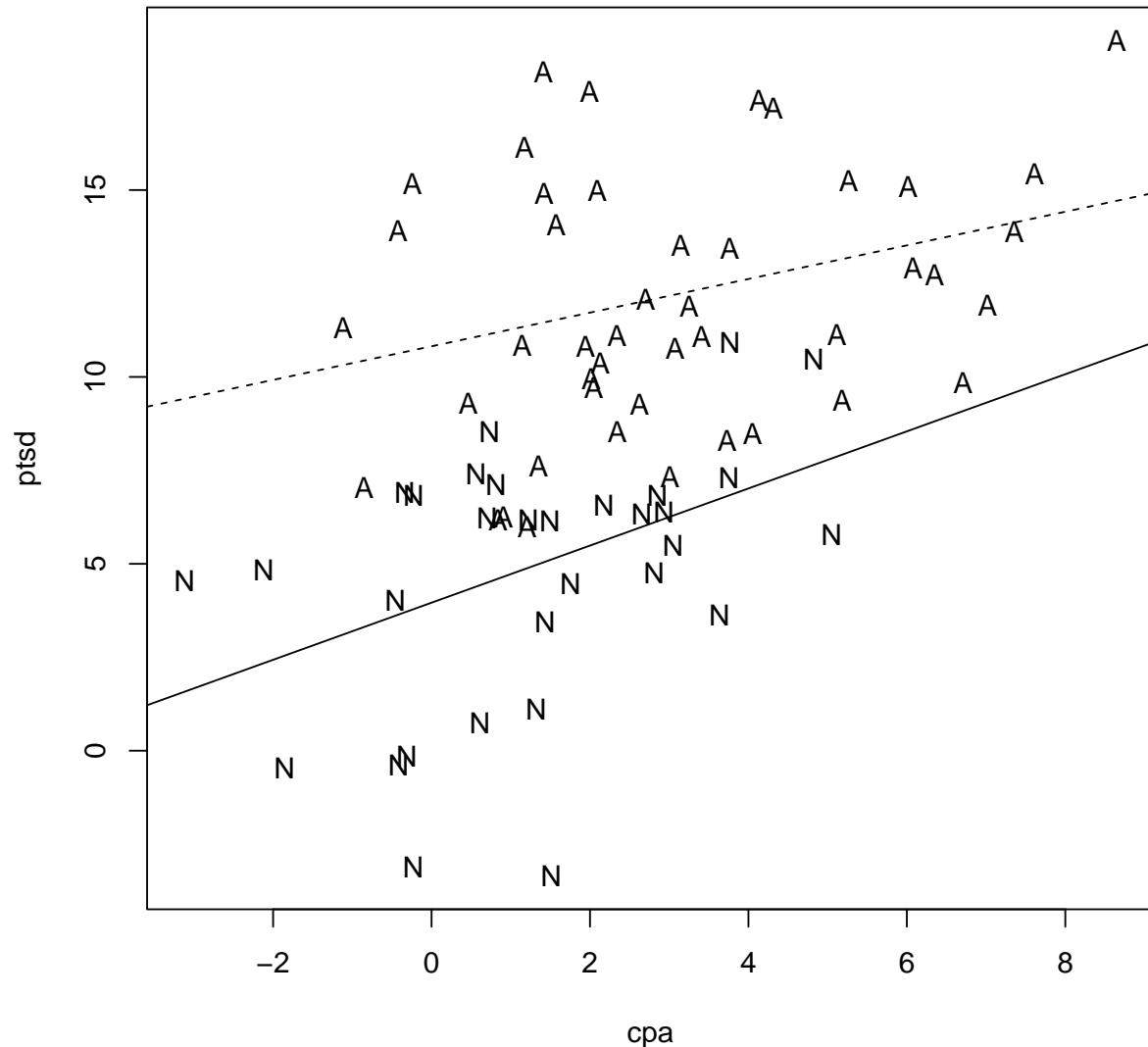
> tail(model.matrix(lmod4))

  (Intercept)      cpa csaAbused cpa:csaAbused
71          1 -0.31402         0         0

```

72	1	2.17626	0	0
73	1	-0.23208	0	0
74	1	-1.85753	0	0
75	1	2.85253	0	0
76	1	0.81138	0	0

```
> plot(ptsd ~ cpa, data=sexab, pch=as.character(csa))
> abline(3.96, 0.764)
> abline(3.96+6.86, 0.764-0.314, lty=2)
```



- **Not So Simple Meaning of Effect.** Notice how our “simple meaning” of effect (§5.1) of csa must be modified when we have an interaction: the effect of csa depends on the value of cpa (and vice versa) (in class and more in the next section, §14.3).
- **Simple Meaning.** But, the interaction term does not appear to be necessary (more shortly).
- **Test for Interaction.** We recall some code illustrated in Testing Examples (§3.2) to (re)draw connections to what we have already learned, perhaps at the risk of overkill in the current example.
- **Been There, Done That.** Again, we should realize that our linear model framework is essentially unchanged when using factors or interactions.

```
> ## Compare Model (4) (fullest) to Model (3)
> ## (This F v R test is already done in a previous chunk, right?):
> anova(lmod3<- update(lmod4, .~. - cpa:csa), lmod4)
```

Analysis of Variance Table

```
Model 1: ptsd ~ cpa + csa
Model 2: ptsd ~ cpa + csa + cpa:csa
  Res.Df RSS Df Sum of Sq   F Pr(>F)
1      73 782
2      72 774  1      7.81 0.73   0.4
```

```
> ## Or, our Cbeta approach (automated):
> (pOmega<- dim(model.matrix(lmod4))[2])
[1] 4
> r<- 1; (pomega<- pOmega - r)
[1] 3
```

```

> Cmat<- cbind(0,0,0,diag(r))
> d<- rep(0, r)
> gmodels::glh.test(reg=lmod4, cm=Cmat, d=d)

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod4, cm = Cmat, d = d)
F = 0.726, df1 = 1, df2 = 72, p-value = 0.397

> ## Note: F in Details section of help(glh.test) is missing an inverse
> ## operation.
>
> ## Or, our Cbeta approach by hand (F Result 3.9)
> x<- model.matrix(lmod4)
> y<- sexab$ptsd
> bhat<- solve(ctx<-crossprod(x), crossprod(x,y))
> Cbhat<- Cmat%*%bhat
> (n<- length(y))

[1] 76

> mse<- sum((y - x%*%bhat)^2)/(n-p0mega)
> Vbhat<- mse*solve(ctx)
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> (F<- t(Cbhat-d)%*%solve(VCbhat)%*%(Cbhat-d) / r)

[,1]
[1,] 0.72596

> pf(q=F, df1=r, df2=n-p0mega, lower=FALSE)

[,1]
[1,] 0.39702

```

- **Omit Interaction.** We adopt the reduced model (3), but no smaller, right?
- **Simple Meaning.** And, now, without an interaction term, we are back to our simple meaning of effect: after adjusting for childhood

physical abuse, childhood sexual abuse is estimated to increase PTSD in women on average by $\hat{\beta}_3 = 6.27$ units compared to women who did not experience childhood physical abuse, regardless of the value of childhood physical abuse as there is now no interaction effect. (Still, this is an observational study, so we must be weary of inferring a causal effect.)

```
> lmod3 <- lm(ptsd ~ cpa+csa,sexab)
> summary(lmod3)

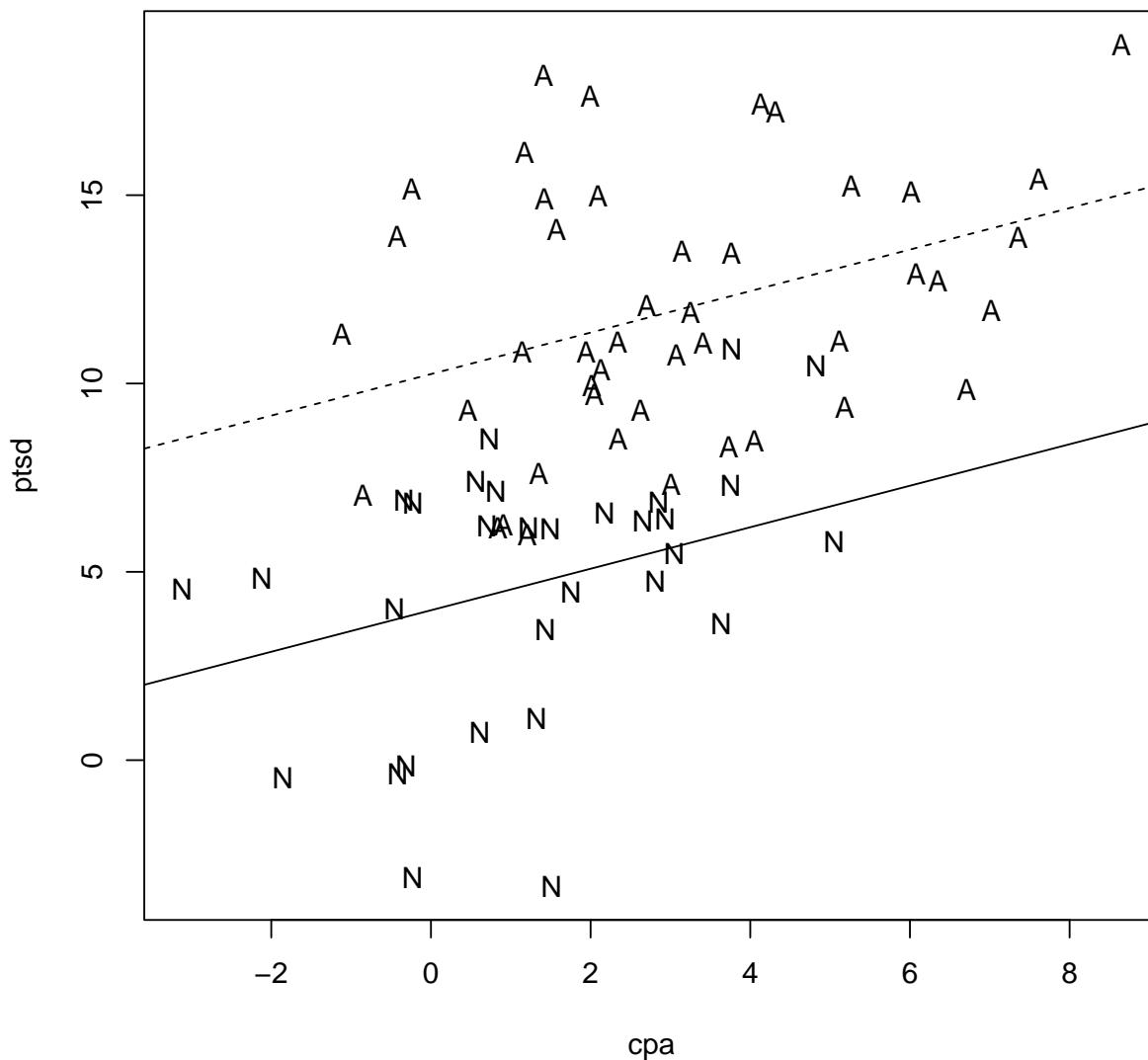
Call:
lm(formula = ptsd ~ cpa + csa, data = sexab)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.157 -2.364 -0.153  2.147  7.142 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.975     0.629    6.32  1.9e-08 *** 
cpa          0.551     0.172    3.21   0.002 **  
csaAbused    6.273     0.822    7.63  6.9e-11 *** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.27 on 73 degrees of freedom
Multiple R-squared:  0.579, Adjusted R-squared:  0.567 
F-statistic: 50.1 on 2 and 73 DF,  p-value: 2e-14
```

```
> plot(ptsd ~ cpa, data=sexab, pch=as.character(csa))
> abline(3.98, 0.551)
> abline(3.98+6.27, 0.551, lty=2)
```

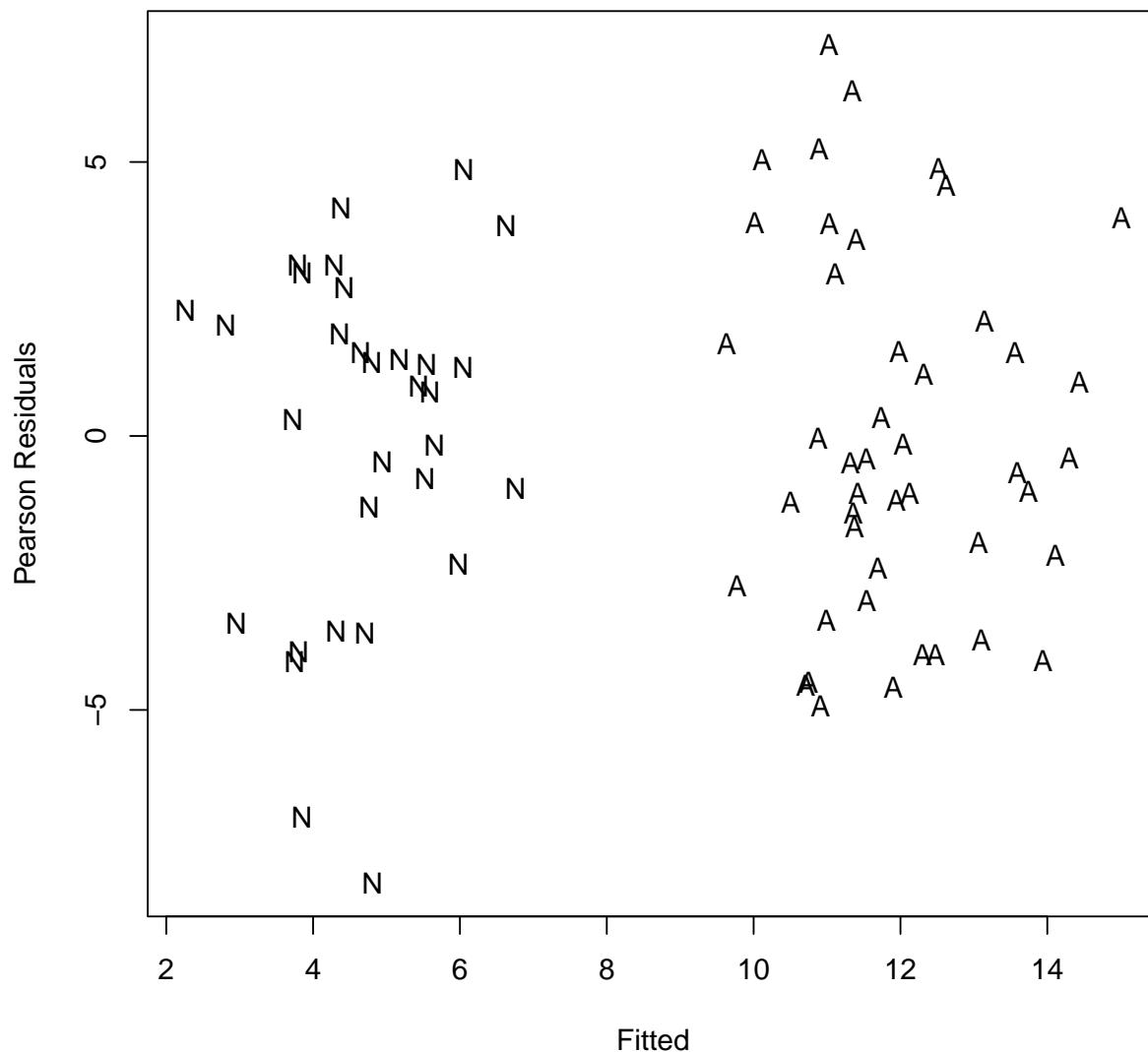


- Etc.: **Confidence Interval**. Of course, we can compute a corresponding **interval estimate of the effect of childhood sexual abuse**, as usual.

```
> confint(lmod3, parm="csaAbused")  
  
    2.5 % 97.5 %  
csaAbused 4.6347 7.9108  
  
> ## Or, ``by hand (with assistance)'' using Results 3.13/3.15  
> ## (What's C?). Using previously created objects.  
> bhat<- coef(lmod3)  
> sehatbhat<- sqrt(diag(vcov(lmod3)))  
> tcrit<- qt(0.975, n-pomega)  
> bhat[3] + c(-1,1) * tcrit * sehatbhat[3]  
  
[1] 4.6347 7.9108
```

- **Etc.: Diagnostics.** And, our diagnostics remain useful, as usual.
- **No Apparent Departures.** The PTSD example data appear consistent with model assumptions.

```
> plot(fitted(lmod3),  
+       residuals(lmod3,type="pearson"),  
+       pch=as.character(sexab$csa),  
+       xlab="Fitted",ylab="Pearson Residuals")
```



- **Just a Reminder: (Un)Adjusting.** Just as we have demonstrated **adjusting** the effect of childhood sexual abuse for childhood physical abuse, we may also view our model as adjusting the effect of childhood physical abuse for childhood sexual abuse, which we can see by omitting csa (we're "unadjusting" the effect of cpa for cpa, so to speak, just to

show it...we would not adopt this model)

```
> ## Of course, omitting a covariate (now csa) will generally change  
> ## ("unadjust") the estimated effect of the remaining covariates  
> ## (cpa), as we already know.  
> summary(lmod1 <- lm(ptsdb ~ cpa, sexab))
```

Call:

```
lm(formula = ptsdb ~ cpa, data = sexab)
```

Residuals:

Min	1Q	Median	3Q	Max
-11.464	-2.385	-0.125	2.261	10.154

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)		
(Intercept)	6.552	0.707	9.26	5.3e-14 ***		
cpa	1.033	0.212	4.87	6.3e-06 ***		

Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'	0.1 ' '	1

Residual standard error: 4.36 on 74 degrees of freedom

Multiple R-squared: 0.242, Adjusted R-squared: 0.232

F-statistic: 23.7 on 1 and 74 DF, p-value: 0.00000627

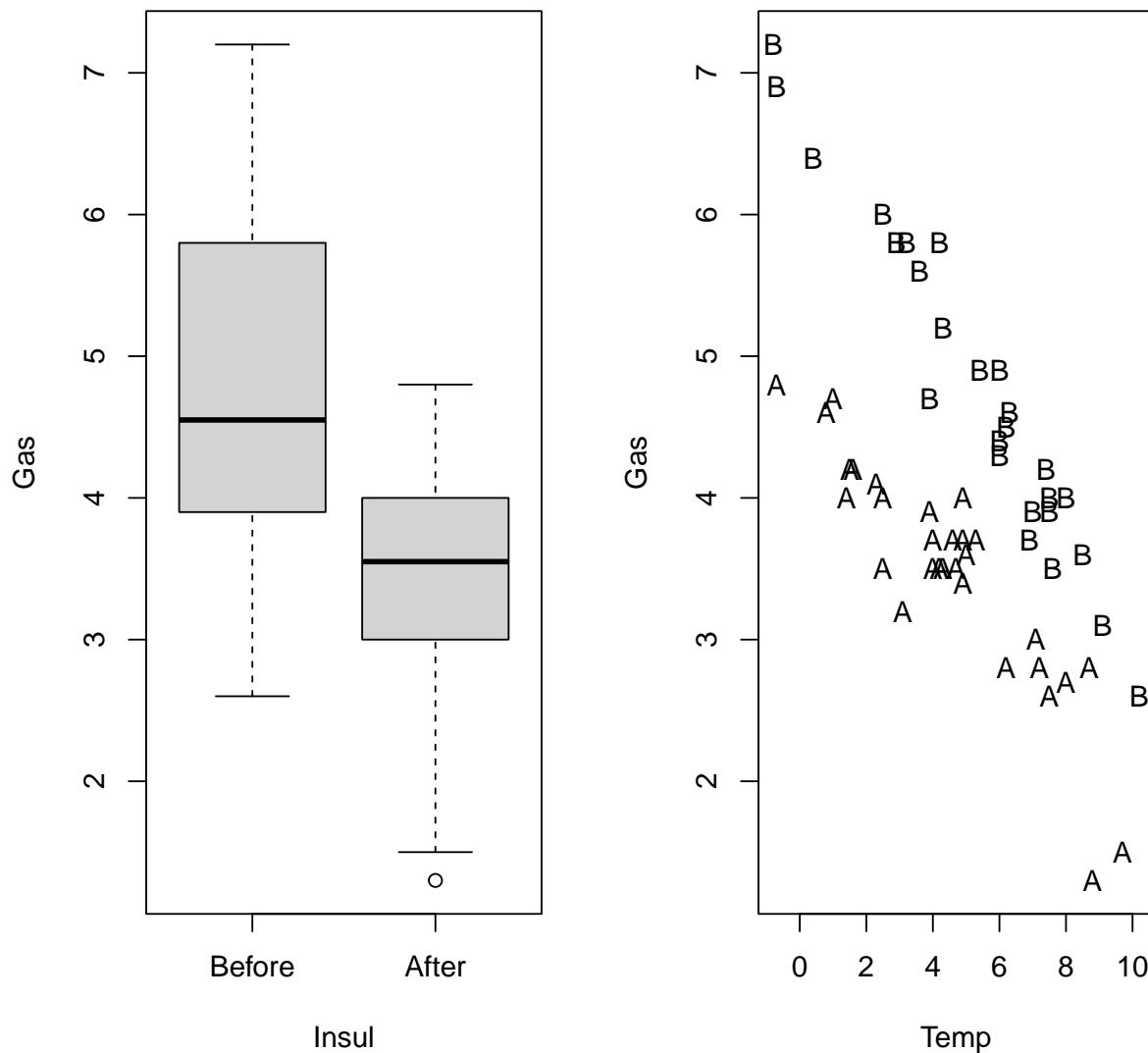
14.3 Interpretation with Interaction Terms

14.3.1 Weekly Natural Gas Consumption Data Example

- **Different but the Same.** These data may seem very different from the PTSD example, but we see how our models for these different data sets are essentially the same.

- **Reillustrations.** Thus, these natural gas consumption data/models **reillustrate essentially the same interpretations in the presence of interacting factors/covariates** as we discussed in the PTSD example.
- **Etc.: Make Intercept More Meaningful.** In particular, the effect of insulation on weekly natural gas consumption appears to depend on temperature; i.e., **insulation and temperature interact**, and we **center the temperature predictor** to interpret the intercept parameters as effects on consumption at average temperature, rather than at temperature zero (Celsius). (See “reparameterize” just prior to our Simple Meaning §5.1.)

```
> data(whiteside, package="MASS")
> par(mfrow=c(1,2))
> plot(Gas ~ Insul, data=whiteside)
> plot(Gas ~ Temp, pch=as.character(Insul), data=whiteside)
```



```
> par(mfrow=c(1,1))
```

```
> ## Interpretation of intercepts at zero temperature
> lmod <- lm(Gas ~ Temp*Insul, data=whiteside)
> summary(lmod)
```

Call:

```
lm(formula = Gas ~ Temp * Insul, data = whiteside)

Residuals:
    Min      1Q  Median      3Q      Max 
-0.9780 -0.1801  0.0376  0.2093  0.6380 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   6.8538     0.1360   50.41 < 2e-16 ***
Temp         -0.3932     0.0225  -17.49 < 2e-16 ***
InsulAfter   -2.1300     0.1801  -11.83 2.3e-16 ***
Temp:InsulAfter 0.1153     0.0321    3.59  0.00073 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.323 on 52 degrees of freedom
Multiple R-squared:  0.928, Adjusted R-squared:  0.924 
F-statistic: 222 on 3 and 52 DF,  p-value: <2e-16

> mean(whiteside$Temp)
[1] 4.875
```

```
> ## Interpretation of intercepts at average temperature
> whiteside$ctemp <- whiteside$Temp - mean(whiteside$Temp)
> lmodc <- lm(Gas ~ ctemp*Insul, data= whiteside)
>
> ## Or, without creating a new centered covariate in data set
> ## (makes printouts ugly) (not run):
> ## lmodc <- lm(Gas ~ I(Temp-mean(Temp))*Insul, data= whiteside)
> summary(lmodc)
```

Call:
`lm(formula = Gas ~ ctemp * Insul, data = whiteside)`

Residuals:

Min	1Q	Median	3Q	Max
-0.9780	-0.1801	0.0376	0.2093	0.6380

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
--	----------	------------	---------	----------

```
(Intercept)      4.9368     0.0642    76.85 < 2e-16 ***
ctemp          -0.3932     0.0225   -17.49 < 2e-16 ***
InsulAfter     -1.5679     0.0877   -17.87 < 2e-16 ***
ctemp:InsulAfter 0.1153     0.0321     3.59  0.00073 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.323 on 52 degrees of freedom
Multiple R-squared:  0.928, Adjusted R-squared:  0.924
F-statistic: 222 on 3 and 52 DF,  p-value: <2e-16
```

```
> ## BTW, we might have checked factor coding beforehand:
> ## What is the global constants setting?
>getOption("contrasts") ## (same as before)

      unordered          ordered
"contr.treatment"      "contr.poly"

> ## Any contrasts set for the particular factor at hand?
> ## (No. This would override a global contrasts setting.)
> attr(whiteside$Insul, which='contrasts')

NULL

> ## Next says R will code a column in X indicating "After" with a 1
> ## (0 for Before (seems like sensible reference level)):
> contrasts(whiteside$Insul)

  After
Before    0
After     1

> ## Xmatrix
> Xmat<- model.matrix(lmodc)
> head(Xmat)

(Intercept)  ctemp InsulAfter ctemp:InsulAfter
1           1 -5.675      0         0
2           1 -5.575      0         0
3           1 -4.475      0         0
4           1 -2.375      0         0
5           1 -1.975      0         0
6           1 -1.675      0         0
```

```
> tail(Xmat)
```

	(Intercept)	ctemp	InsulAfter	ctemp:InsulAfter
51	1	2.325	1	2.325
52	1	2.625	1	2.625
53	1	3.125	1	3.125
54	1	3.825	1	3.825
55	1	3.925	1	3.925
56	1	4.825	1	4.825

- **Same Inference Tools as Usual.** Again risking overkill for this example, we remind ourselves that our usual inference tools apply (Chapter 3).
- **Etc.: Mildly Interesting Confidence Interval.** We illustrate with a 95% **confidence interval** for the difference between the insulation effect at 2 degrees (centered) and the insulation effect at 8 degrees. (Just an illustration that I pulled from the top of my head.)
- **Result:** the effect of insulation on gas consumption at 2 degrees is estimated to be about 0.7 units better than the effect of insulation on consumption at 8 degrees, and is estimated to be between 0.30 to 1.08 units better with 95% confidence. In other words, insulation is more effective at lowering gas consumption at lower temperatures than at higher temperatures, which is no surprise, even before looking at the data.
- **Do We Trust Nominal Results?** We didn't do much data torture, but, also, we didn't check our model.

```
> ## Recall Section 3.6
> Crow2B<- c(1, 2-mean(whiteside$Temp), 0, 0)
> Crow2A<- c(1, 2-mean(whiteside$Temp), 1, 2-mean(whiteside$Temp))
> Crow8B<- c(1, 8-mean(whiteside$Temp), 0, 0)
> Crow8A<- c(1, 8-mean(whiteside$Temp), 1, 8-mean(whiteside$Temp))
```

```

> ## (effect at 2) - (effect at 8):
> (Cmat<- t(as.matrix(Crow2B-Crow2A) - (Crow8B-Crow8A)))

      [,1] [,2] [,3] [,4]
[1,]     0     0     0     6

> gmodels::estimable(lmodc, cm=Cmat, conf.int=0.95)

      Estimate Std. Error t value DF   Pr(>|t|) Lower.CI Upper.CI
(0 0 0 6)  0.69182    0.19267  3.5907 52 0.00073069  0.3052  1.0784

> ##
> ## Or, by hand (Section 3.6)
> x<- model.matrix(lmodc)
> y<- whiteside$Gas
> bhat<- solve(xtx<-crossprod(x), crossprod(x,y))
> (n<- length(y))

[1] 56

> (p0mega<- dim(x)[2])

[1] 4

> (mse<- sum((y - x%*%bhat)^2)/(n-p0mega))

[1] 0.10433

> Vbhat<- mse*solve(xtx)
> VCbhat<- Cmat%*%Vbhat%*%t(Cmat)
> Cbhat<- as.vector(Cmat%*%bhat)
> sehatCbhat<- as.vector(sqrt(Cmat%*%Vbhat%*%t(Cmat)))
> Cbhat + c(-1,1) * qt(1-0.05/2, n-p0mega) * sehatCbhat

[1] 0.3052 1.0784

```

14.4 Factors with More Than Two Levels

14.4.1 Cell Reference Coding (Again)

- aka. treatment or corner-point coding, the default coding in R.
- **Generally Now.** More generally, for cell reference coding, if we have a **categorical**/qualitative variable (factor) with

f categories/values/levels,

then we create/code

$(f - 1)$ new **quantitative** predictor variables,

illustrated as follows.

- Say X is a categorical variable with $f = 3$ levels: “level 1”, “level 2”, and “level 3”.
- Create $(3-1) = 2$ new predictor variables

$$X_1 = \begin{cases} 1 & \text{if } X = \text{level 2} \\ 0 & \text{if } X \text{ value is not level 2} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if } X = \text{level 3} \\ 0 & \text{if } X \text{ value is not level 3} \end{cases}$$

- Then, perform linear regression as usual using the new predictor variables:

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + (\text{other terms}) + \epsilon_i$$

- The matrix \mathbf{X} will include $(f - 1)$ columns for the coded predictors (among other columns for other predictors), e.g.,

X_1	X_2	original X
0	0	level 1
0	0	level 1
0	0	level 1
1	0	level 2
1	0	level 2
1	0	level 2
0	1	level 3
0	1	level 3
0	1	level 3

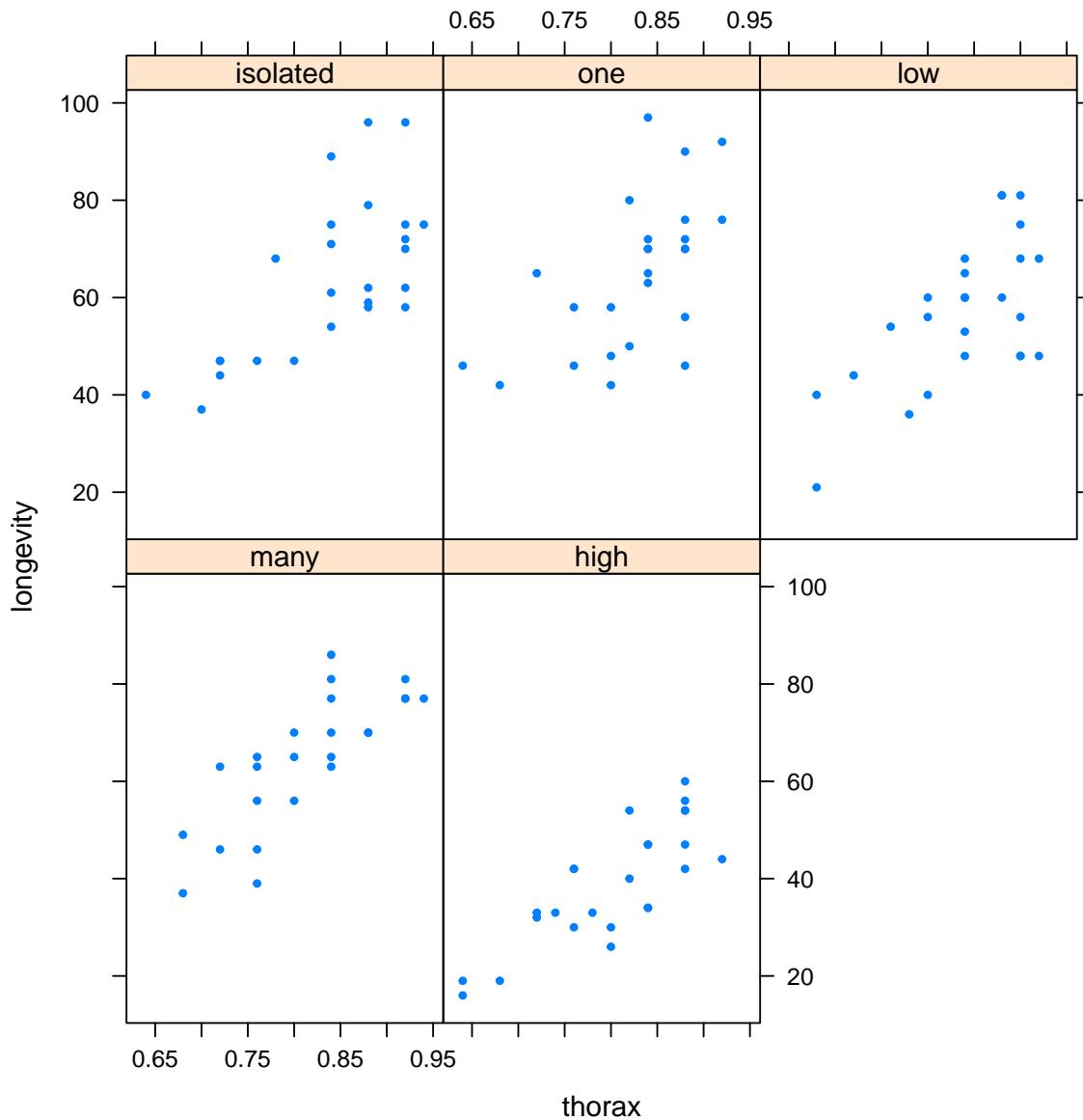
- We will consider alternative coding schemes, later.
- **Interpretation of parameters** for the (cell reference) coded variables.
 - β_1 is the shift (up/down) in the regression model (surface, hyperplane) from **reference level** $X = \text{level 1}$ to $X = \text{level 2}$
 - β_2 is the shift (up/down) in the regression model (surface, hyperplane) from **reference level** $X = \text{level 1}$ to $X = \text{level 3}$
 - $\beta_0 +$ (other terms) corresponds to the “reference level” (or “corner-point”) (level 1 plus other terms).
 - Level 1 is the level to which all other levels of original X are compared.
- **(Again) Parameter Interpretation Depends on Coding**.: There are several ways to numerically code categorical variables, and parameter interpretation depends on coding! See [Far14, §14.5] and [KNNL05, 8.4].

14.4.2 Longevity and Sexual Activity Data Example

- **Yea!** Now, we discuss sexual activity and longevity...
- **Yes!**...of fruit flies.
- **The More You Know.** Did you know that humans and fruit flies share about 44% of their genes and 75% of disease causing genes are similar between fruit flies and humans (two statistics that I shamelessly took from the Internet without citation or further investigation).
- **Randomized Experiment.** $n = 125$ fruit flies were randomly assigned to **five activity groups (levels)**, each group consisting of 25 flies each:
 - isolated = fly kept solitary
 - one = fly kept with one pregnant fruit fly

- many = fly kept with eight pregnant fruit flies
 - low = fly kept with one virgin fruit fly
 - high = fly kept with eight virgin fruit flies.
- **Longevity** (days) is the recorded response.
 - **Thorax length**, known to vary with longevity, is a quantitative predictor.
 - **Effect of Activity on Longevity After Adjusting for Thorax.** As illustrated by this example, we seek to infer how a response varies across the levels of a treatment factor after **adjusting** for relatively uninteresting quantitative predictors; this is sometimes referred to as **ANCOVA** (analysis of covariance).
 - **Different but the Same.** Our analysis is not much different from the previous 2-level factor analyses (PTSD and gas consumption were ANCOVA too, but had an observational factor, not an experimental factor).

```
> data(fruitfly, package="faraway")
> lattice::xyplot(longevity ~ thorax | activity,
+                   data=fruitfly, pch=20, cex=.7,
+                   as.table=TRUE)
```



```
> sapply(fruitfly, data.class)

  thorax longevity activity
"numeric" "numeric" "factor"

> attributes(fruitfly$activity)

$levels
[1] "isolated" "one"      "low"       "many"     "high"
```

```

$class
[1] "factor"

> attr(fruitfly$activity, which="contrasts")

NULL

>getOption(x="contrasts")

      unordered          ordered
"contr.treatment"    "contr.poly"

> ## Model with interaction
> lmod <- lm(longevity ~ thorax*activity, fruitfly)
> summary(lmod)

Call:
lm(formula = longevity ~ thorax * activity, data = fruitfly)

Residuals:
    Min      1Q  Median      3Q     Max 
-25.95  -6.73  -0.91   6.18  30.31 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -50.242    21.801   -2.30   0.023 *  
thorax       136.127   25.952    5.25  7.3e-07 *** 
activityone   6.517    33.871    0.19   0.848    
activitylow  -7.750    33.969   -0.23   0.820    
activitymany -1.139    32.530   -0.04   0.972    
activityhigh -11.038   31.287   -0.35   0.725    
thorax:activityone -4.677   40.652   -0.12   0.909    
thorax:activitylow  0.874    40.425    0.02   0.983    
thorax:activitymany 6.548    39.360    0.17   0.868    
thorax:activityhigh -11.127   38.120   -0.29   0.771    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.7 on 114 degrees of freedom
Multiple R-squared:  0.653, Adjusted R-squared:  0.626 
F-statistic: 23.9 on 9 and 114 DF,  p-value: <2e-16

```

- Re-verify cell reference (treatment) coding by looking at **X** matrix.

```
> head(Xmat<-model.matrix(lmod))

  (Intercept) thorax activityone activitylow activitymany
1           1   0.68         0         0           1
2           1   0.68         0         0           1
3           1   0.72         0         0           1
4           1   0.72         0         0           1
5           1   0.76         0         0           1
6           1   0.76         0         0           1

  activityhigh thorax:activityone thorax:activitylow
1             0             0             0
2             0             0             0
3             0             0             0
4             0             0             0
5             0             0             0
6             0             0             0

  thorax:activitymany thorax:activityhigh
1           0.68         0
2           0.68         0
3           0.72         0
4           0.72         0
5           0.76         0
6           0.76         0

> tail(Xmat)

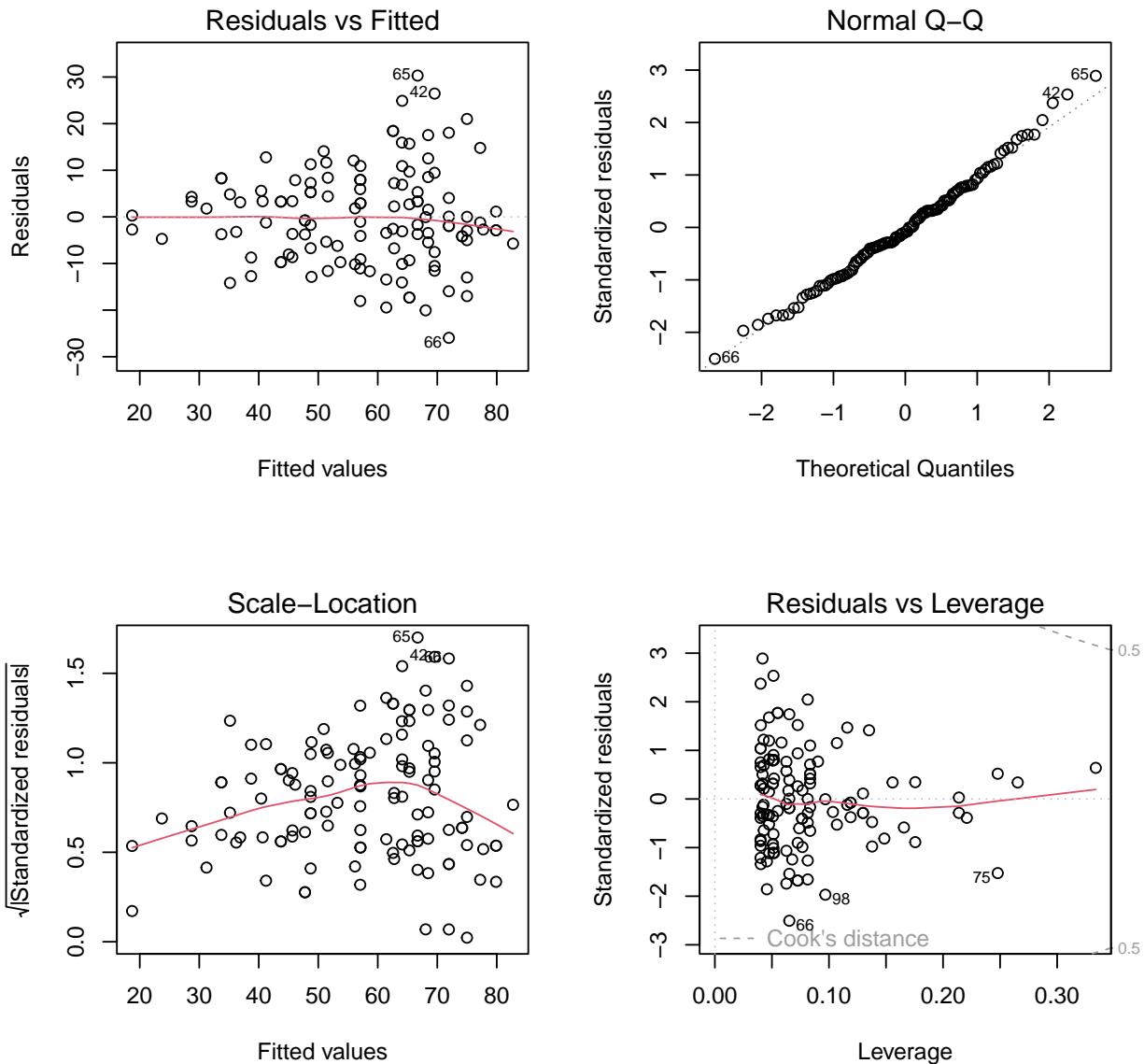
  (Intercept) thorax activityone activitylow activitymany
119          1   0.88         0         0           0
120          1   0.88         0         0           0
121          1   0.88         0         0           0
122          1   0.88         0         0           0
123          1   0.88         0         0           0
124          1   0.92         0         0           0

  activityhigh thorax:activityone thorax:activitylow
119          1             0             0
120          1             0             0
121          1             0             0
```

```
122      1          0          0
123      1          0          0
124      1          0          0
  thorax:activitymany thorax:activityhigh
119          0          0.88
120          0          0.88
121          0          0.88
122          0          0.88
123          0          0.88
124          0          0.92
```

Diagnostics

```
> par(mfrow=c(2,2))
> plot(lmod)
```



```
> par(mfrow=c(1, 1))
```

Testing for Simpler Models: Sequential ANOVA Table

- **A Sequence of Full v. Reduced Model Comparisons.** In short, the sequential ANOVA table reports a sequence of full vs. reduced model

comparisons that is simply a summary of computations that we already know how to do (with a slight twist).

- **Now, More Than 2 Models.** Until now we have only compared two models, one full, one reduced. Now we have more than two.
- **Fullest Model.** All comparisons are made with respect to the residual sum of squares and residual degrees of freedom of the fullest model.
- **Almost Same as Before.** In the context of the notation of the unnumbered section, “Extra Sum of Squares...,” in the numbered section §3.1, we might call these “ $RSS_{fullest}$ ” and $df_{n-p_{fullest}}$, which are used in the denominator of the computed F statistics.
- See [Far14, p. 216] for the models being compared; more in class.

```
> ## Sequential (aka Type I) ANOVA table. Do we need an interaction term?
> anova(lmod)

Analysis of Variance Table

Response: longevity
          Df Sum Sq Mean Sq F value    Pr(>F)
thorax       1 15003   15003 130.73 < 2e-16 ***
activity      4   9635    2409  20.99 5.5e-13 ***
thorax:activity  4      24       6    0.05    0.99
Residuals    114 13083     115
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **Feature or Flaw of Sequential ANOVA?** After omitting the interaction, we might test to see if we can omit one or another of remaining covariates while adjusting for the remaining remaining (!) covariates, but the sequential table will not do this with the exception of the term entering the model last.

- **drop1**. Your author uses this “shortcoming” (or “feature”) of the sequential ANOVA table to motivate the use of the **drop1** function, which will test for the effect(s) of one term while adjusting for (including in the model) the remaining term(s).
- **Partial (Type III) ANOVA Table.** We might also use a partial ANOVA (Type III) table, too, for this, which reports the test of terms after adjusting for all other terms in the model regardless of which row of the table we look at.
- **BE CAREFUL:** we typically do not test for the effect(s) of terms that are also involved in interactions (this is called the **marginality principle** or **hierarchy principle**), which we will discuss in class in the context of the example.

```
> ## Fit the reduced model, without the interaction term effects.
> lmodp <- lm(longevity ~ thorax + activity, fruitfly)
>
> ## Is this what you want?
> anova(lmodp)
```

Analysis of Variance Table

Response: longevity

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
thorax	1	15003	15003	135.1	<2e-16 ***
activity	4	9635	2409	21.7	2e-13 ***
Residuals	118	13107	111		
<hr/>					

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> ## Test for effect(s) of terms adjusted for others:
> drop1(lmodp, test="F")
```

Single term deletions

Model:

longevity ~ thorax + activity

```
Df Sum of Sq   RSS AIC F value Pr(>F)
<none>           13107 590
thorax      1     12368 25476 670    111.3 <2e-16 ***
activity     4     9635 22742 650    21.7  2e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

> ## Or, we may use the partial (aka Type III) ANOVA table (careful with
 > ## interpretation/testing: marginality/hierarchy principle.
 > car::Anova(lmod, type=3, test="F")

Anova Table (Type III tests)

Response: longevity

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	610	1	5.31	0.023 *
thorax	3158	1	27.51	7.3e-07 ***
activity	37	4	0.08	0.988
thorax:activity	24	4	0.05	0.995
Residuals	13083	114		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

> ## Ok.

> car::Anova(lmodp, type=3, test="F")

Anova Table (Type III tests)

Response: longevity

	Sum Sq	Df	F value	Pr(>F)
(Intercept)	2242	1	20.2	1.6e-05 ***
thorax	12368	1	111.3	< 2e-16 ***
activity	9635	4	21.7	2.0e-13 ***
Residuals	13107	118		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **Again, A Reminder about Convenience.** We already know how to do such tests with previous methods. The ANOVA table(s) are merely

a convenient summary of tests that are commonly done, which may or may not be of interest to us (if we're not somehow common), and which we already know how to do. Still, you may find the results convenient if you find our previous approaches too cumbersome. Ultimately, however, you will likely want to test/estimate things whose analyses have not been pre-determined and summarized by someone else; in this case, just use our methods.

- **Revisit Our Tools.** We'll revisit our usual tools and compare them to the ANOVA table on a homework to help us understand connections.

A Final Model?

```
> summary(lmodp)

Call:
lm(formula = longevity ~ thorax + activity, data = fruitfly)

Residuals:
    Min      1Q  Median      3Q     Max 
-26.11   -7.01  -1.10    6.23   30.27 

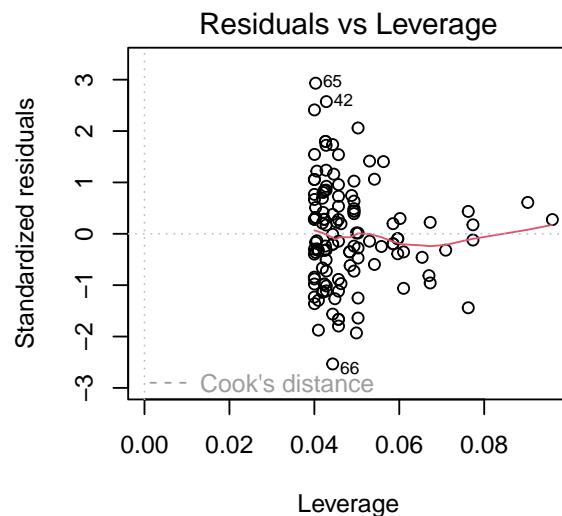
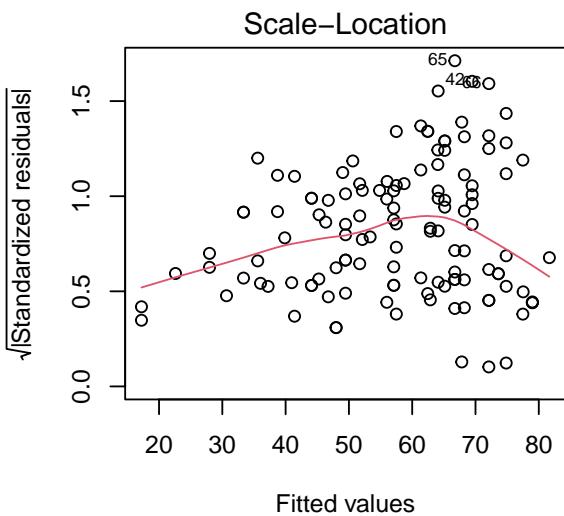
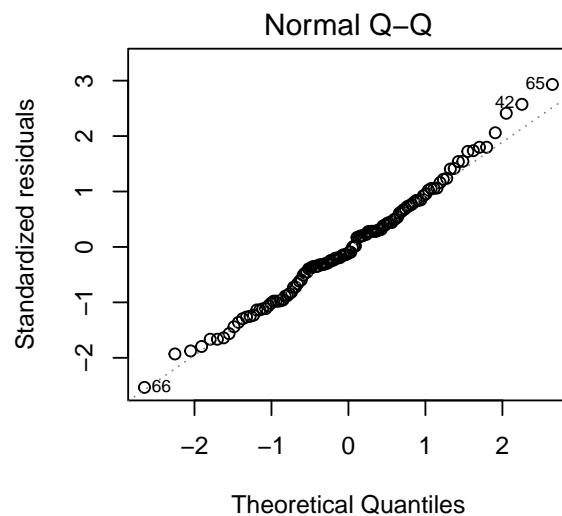
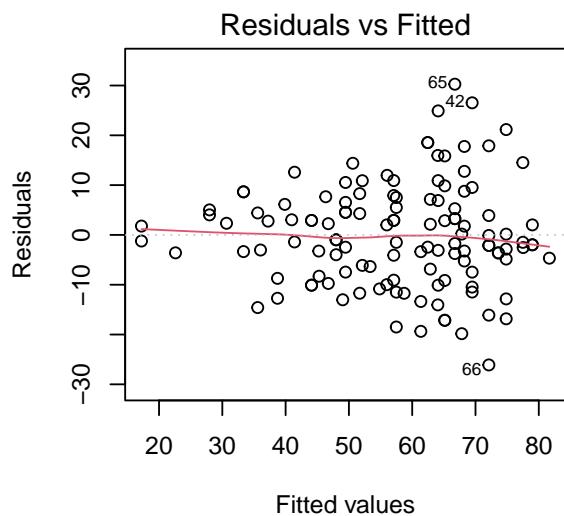
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -48.75     10.85   -4.49  1.6e-05 *** 
thorax       134.34     12.73   10.55 < 2e-16 *** 
activityone   2.64      2.98    0.88    0.38    
activitylow  -7.01      2.98   -2.35    0.02 *  
activitymany  4.14      3.03    1.37    0.17    
activityhigh -20.00     3.02   -6.63   1.0e-09 *** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.5 on 118 degrees of freedom
Multiple R-squared:  0.653, Adjusted R-squared:  0.638 
F-statistic: 44.4 on 5 and 118 DF,  p-value: <2e-16
```

Diagnostics (Again)

- Non-constant variance.

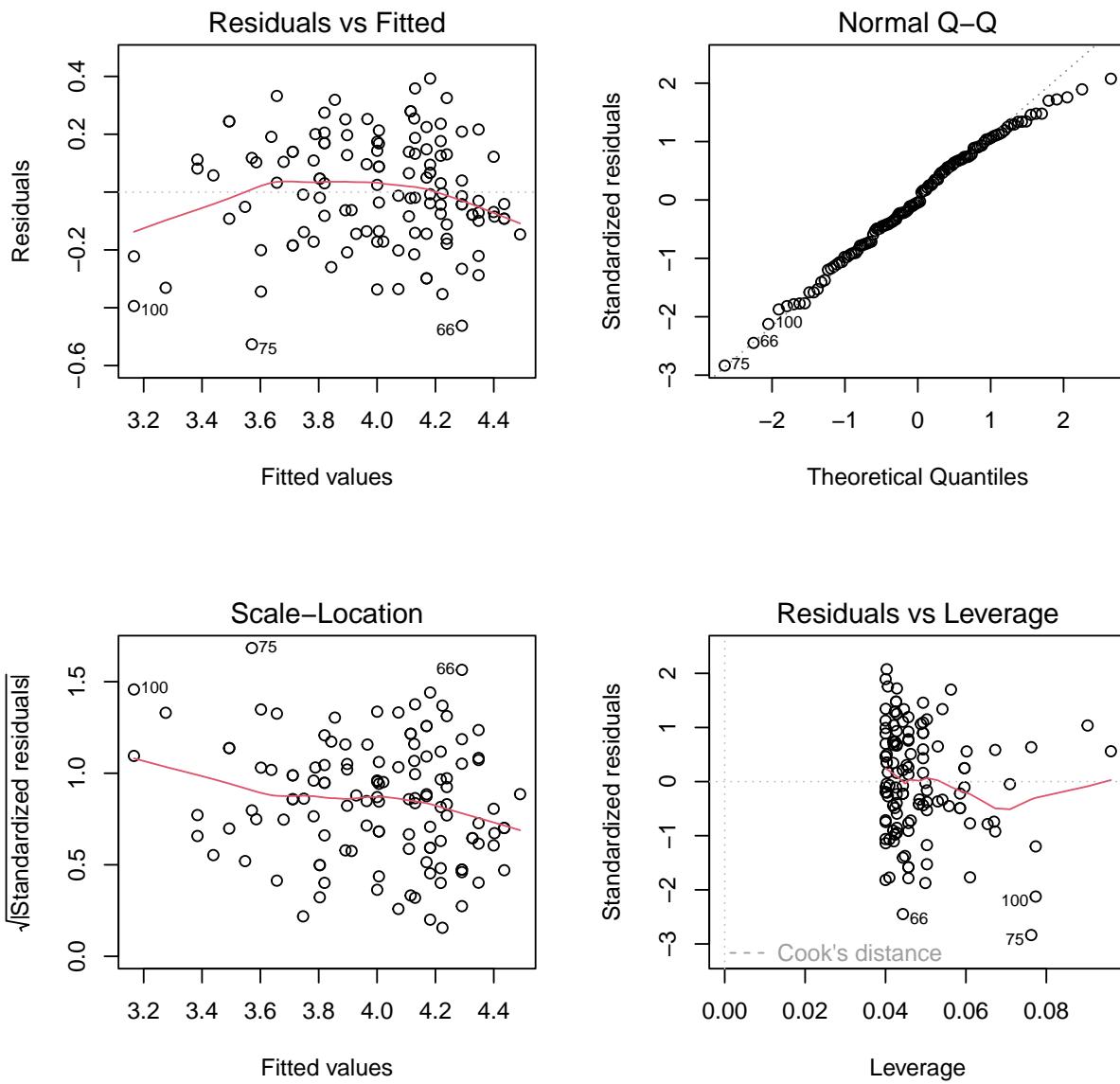
```
> par(mfrow=c(2,2))
> plot(lmodp)
```



```
> par(mfrow=c(1,1))
```

- Try log transformation to remediate non-constant variance.

```
> lmod1 <- lm(log(longevity) ~ thorax+activity, fruitfly)
> par(mfrow=c(2,2))
> plot(lmod1)
```



```
> par(mfrow=c(1,1))
```

- Interpretation on log scale may be unfamiliar.

```
> summary(lmod1)

Call:
lm(formula = log(longevity) ~ thorax + activity, data = fruitfly)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.5264 -0.1363 -0.0082  0.1392  0.3927 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.8442     0.1988   9.28  1.0e-15 ***
thorax       2.7215     0.2333  11.67 < 2e-16 ***
activityone  0.0517     0.0547   0.95   0.346    
activitylow  -0.1239    0.0546  -2.27   0.025 *  
activitymany  0.0879     0.0555   1.59   0.116    
activityhigh -0.4193    0.0553  -7.59  8.4e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.193 on 118 degrees of freedom
Multiple R-squared:  0.702, Adjusted R-squared:  0.69 
F-statistic: 55.7 on 5 and 118 DF,  p-value: <2e-16
```

A Final, Sexy Inference

- **Multiplicative Effects to Median.** Instead of the usual Simple Meaning of parameters, β_j , as additive effects to the mean (when adding 1 to a covariate x_j , others fixed) (§5.1) we may look instead at multiplicative effects to the median, $\exp(\beta_j)$ (when adding 1 to x_j). For example, we can get a 95% CI for a multiplicative effect to the median by simply exponentiating the CI for an additive effect.
- Alas, what does this say about a high level of sexual activity...for fruit flies...compared to a solitary life?

```
> ## 95% CIs for multiplicative effects exp(beta_j)
> exp(confint(lmodl))

              2.5 %    97.5 %
(Intercept) 4.26523 9.37385
thorax       9.57812 24.12948
activityone  0.94503  1.17354
activitylow   0.79291  0.98443
activitymany  0.97832  1.21865
activityhigh  0.58937  0.73359
```

14.4.3 Let's Recall Mixing and Fixing (Nothing to Do with Sex)

- **Mixing.** We know that the random assignment of fruit flies to activity groups ensures that thorax—a relatively uninteresting observational covariate—does not bias (on average) the effects of the relatively interesting experimental activity factor (recall our notion of mixing in Chapter 5).
- Thus, as we expect, the random assignment (mixing) does appear to have prevented any association between thorax and activity (that may cause activity effects on longevity to be biased by an imbalance of the thorax covariate across groups). We're just recalling concepts.

```
> ## Exploring: Did mixing thorax break its potential to bias the effect
> ## of activity?
> summary(lmodh <- lm(thorax ~ activity, fruitfly))
```

Call:

```
lm(formula = thorax ~ activity, data = fruitfly)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.1960	-0.0525	0.0144	0.0544	0.1275

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.8360	0.0152	55.08	<2e-16 ***
activityone	-0.0104	0.0215	-0.48	0.629
activitylow	0.0016	0.0215	0.07	0.941
activitymany	-0.0235	0.0217	-1.08	0.281
activityhigh	-0.0360	0.0215	-1.68	0.096 .

Signif. codes:	0 '***'	0.001 '**'	0.01 '*'	0.05 '.'
	0.1 ' '	1		

Residual standard error: 0.0759 on 119 degrees of freedom

Multiple R-squared: 0.0359, Adjusted R-squared: 0.00354

F-statistic: 1.11 on 4 and 119 DF, p-value: 0.355

> **anova**(lmodh)

Analysis of Variance Table

Response: thorax

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
activity	4	0.026	0.00639	1.11	0.36
Residuals	119	0.685	0.00576		

- **Adjusting in the Spirit of Fixing.** Again, as in Chapter 5, we mixed covariate values among treatment groups to address effect bias, but we also want to “fix” covariate values to prevent masking of effects; i.e., we don’t want the covariability of longevity with observed covariate thorax length to be put into the error term, which would tend to mask activity effects (larger p-values, wider interval estimates).
- Of course, we did not fix thorax length, so, once again, as in Chapter 5, we adjust for it in the spirit of fixing (similar to matching).
- To see this masking effect, below, we remove thorax length from the model to see activity effects estimates similar in value to those from the model that adjusts for thorax length (above), and we see larger p-values and larger intervals, as we said.

- (We expect similar effects estimates compared to the model that adjusts for thorax length because we expect the mixing of thorax length across activity levels to result in no change to activity effects (on average).)
- Keep thorax, of course.
- **Restricted Randomization?** Could we have fixed thorax? That is could we have restricted randomization to fruit flies with the same (similar) thorax lengths? At what cost/benefit? (restricted inference, more flies and more degrees of freedom, no assumption about form of relationship (linear here) of thorax to longevity...)

```
> summary(lmmodu <- lm(log(longevity) ~ activity, fruitfly))
```

Call:

```
lm(formula = log(longevity) ~ activity, data = fruitfly)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.9553	-0.1319	0.0311	0.1981	0.4922

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.1193	0.0564	72.99	< 2e-16 ***
activityone	0.0234	0.0798	0.29	0.77
activitylow	-0.1195	0.0798	-1.50	0.14
activitymany	0.0240	0.0806	0.30	0.77
activityhigh	-0.5172	0.0798	-6.48	2.2e-09 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.282 on 119 degrees of freedom

Multiple R-squared: 0.359, Adjusted R-squared: 0.338

F-statistic: 16.7 on 4 and 119 DF, p-value: 6.96e-11

```
> anova(lmmodu)
```

```
Analysis of Variance Table
```

```
Response: log(longevity)
          Df Sum Sq Mean Sq F value Pr(>F)
activity      4   5.32    1.33    16.7  7e-11 ***
Residuals 119   9.48    0.08
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> exp(confint(lmodu,which=c(2:5)))

           2.5 % 97.5 %
(Intercept) 55.01421 68.79329
activityone  0.87406  1.19900
activitylow  0.75763  1.03929
activitymany 0.87308  1.20159
activityhigh 0.50902  0.69825
```

14.5 Alternative Codings of Qualitative Predictors

We skip [Far14, §14.5] because codings/contrasts/parameterizations are typically discussed in a manner specialized to factor (qualitative/categorical) covariates as we will see shortly in [Far14, Chap. 15], next. Indeed, we could have economized/streamlined by presenting the material of this [Far14, Chap. 14] in the context of the next [Far14, Chap. 15], as your author seems to allude to in his introduction to [Far14, Chap. 15].

Lecture 15

One Factor Models

Contents

Introduction	383
15.1 The Model and Example	384
Motivating Example	384
Notation & Initial Concepts	386
15.1.1 Cell Means Model	388
15.1.2 Cell Means Model (Example Continued)	392
15.1.3 Effects Model: Before Constraints	397
15.1.4 Effects Model: Treatment Coding/Constraint	401
15.1.5 Effects Model: Treatment Coding/Constraint Example	402
15.1.6 Effects Model: Sum (to Zero) Coding/Constraint	408
15.1.7 Effects Model: Sum to Zero Coding/Constraint Example	409
15.1.8 Regression Approach to ANOVA	414
15.1.9 Model, Parameterization, Reparameterization, Coding, Constraints	415
15.2 An Example (NOT)	416
15.3 Diagnostics	416
15.4 Pairwise Comparisons	421
Simultaneous Inferences & Multiple Comparison Procedures	422
Confidence Intervals	423
Tests	424
Hypothetical Replications	424
Example: Confidence Intervals	425
Family-wise Confidence Level	426
Family-wise Error Rate	427

Bonferroni Inequality & Multiple Comparison Procedure	427
Tukey–Kramer Pairwise Comparison Procedure	430
Tukey Pairwise Comparison Example	431
Scheffé's Procedure for Contrasts	435
Pre-planned Comparisons and Data Snooping	435
Remarks on Multiple Comparison Procedures	439
15.5 False Discovery Rate	440

Main Objectives:

- Factor covariates/inputs
 - Cell means model
 - Effects model
 - Sum to zero constraints/coding
 - Treatment constraints/coding aka cell reference coding or corner point coding
 - Multiple comparison (test/interval) procedures including: concepts of individual (comparison-wise) error rates and confidence levels vs. family-wise error rates (FWER) and confidence levels; Bonferroni tests/intervals, Tukey-Kramer pairwise tests/intervals, Scheffé tests/intervals; pre-planned comparisons and data snooping/torture; false discovery rate (FDR).
-
-

Reading:

- Some considerations for using a categorical variable as a covariate/input along with numerical covariates are given in [Far14, Chapter 14]. We may largely leave this material to a homework problem.
- Mainly: [Far14, Chapter 15]
- Very briefly: [Wak13, §5.8.1]
- More thoroughly and classically: [KNNL05, Chapter 16]

R

Introduction

- **Categorical Variables.** Categorical variables are qualitative, not quantitative, not numeric.
- **Synonyms:** factor, categorical variable, qualitative variable, grouping variable, classification variable (and probably more that I'm forgetting!)
- **Levels.** The possible values of a categorical variable are often referred to as its levels; i.e., classes, groups, categories
- **Nominal or Ordinal.** Sometimes, categorical variables are refined into subcategories of nominal (name only) or ordinal (some order implied). We do not use the information contained in the ordinality of categorical variables in this class.
- **Examples.** For example, a categorical variable X may characterize **color**, with levels of red, green and blue (a nominal variable); **gender**, male and female (nominal); **height class**: short, average, tall (ordinal); **motivation treatment**: extrinsic, intrinsic (nominal) (as in a previous 511 example); **study guide**: A, B (as in a homework); **voting technology** digital, hand (nominal) (as in a previous 511 example).
- **As Covariates or Inputs (in Part I at least).** I use " X " because we will restrict our attention to categorical predictors/covariates/inputs; we will maintain normality for response, Y , until we get to Part II of INF 512.
- **Numerical Coding.** A factor's levels are assigned numerical codes for computation in our models. These numerical codes go into the columns our **X** matrix, as do the observation of numerical covariates in the usual manner of INF 511. A few different **coding schemes** are commonly used, each associated with a particular **parameterization**, i.e., parameter **interpretation**.

- **Examples.** **study guides:** 0=A, 1=B; **voting technology:** 0=digital, 1=hand. We'll see other coding schemes, too.
- **Estimation, inference, diagnostics.** Essentially, **all of our previous material for linear models still applies!** Once we have built our "regression" (ANOVA) matrix, \mathbf{X} , previous methods apply! Easy! (though we will adopt special notation, and interpretation needs special care now)
- **One Factor Models** are those models with only a single categorical variable, i.e., factor, as a covariate/input. These models are frequently called **one-way ANOVA** (ANalysis Of VAriance) models. When one or more numerical, i.e., "regression," covariates are included with (one or more) factor variables [Far14, Chapter 14], sometime we hear the term analysis of covariance (**ANCOVA**), depending on context.

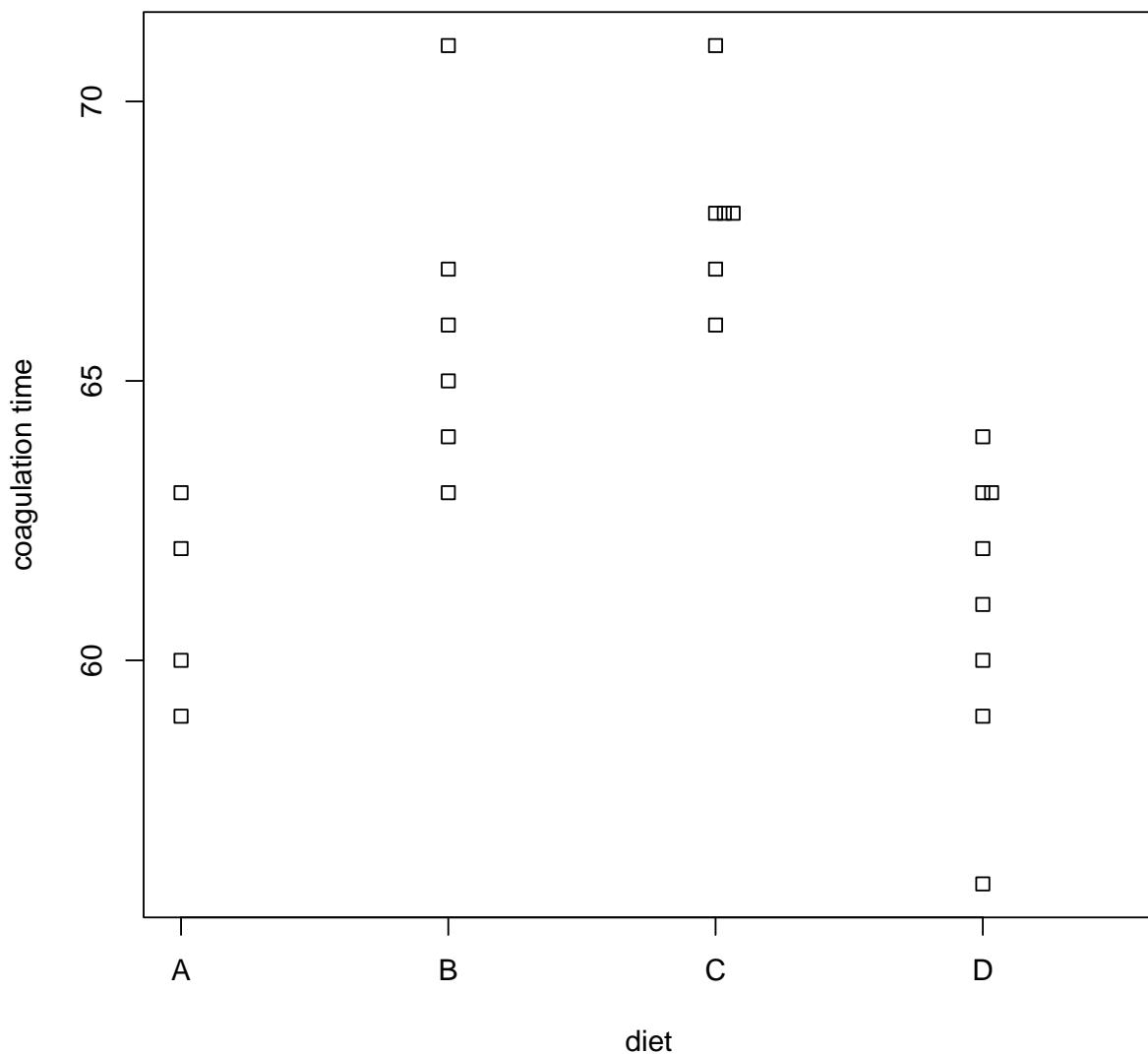
15.1 The Model and Example

- Note, we group together [Far14, §15.1 & 15.2] into §15.1, here, and my typical (INF 511) use of numbered (sub)sections to correspond to textbook section begins to fall apart a bit. In subsequent chapters, we will abandon entirely any attempt to have note sections numbered similarly to text sections, though I will continue to use references to texts.

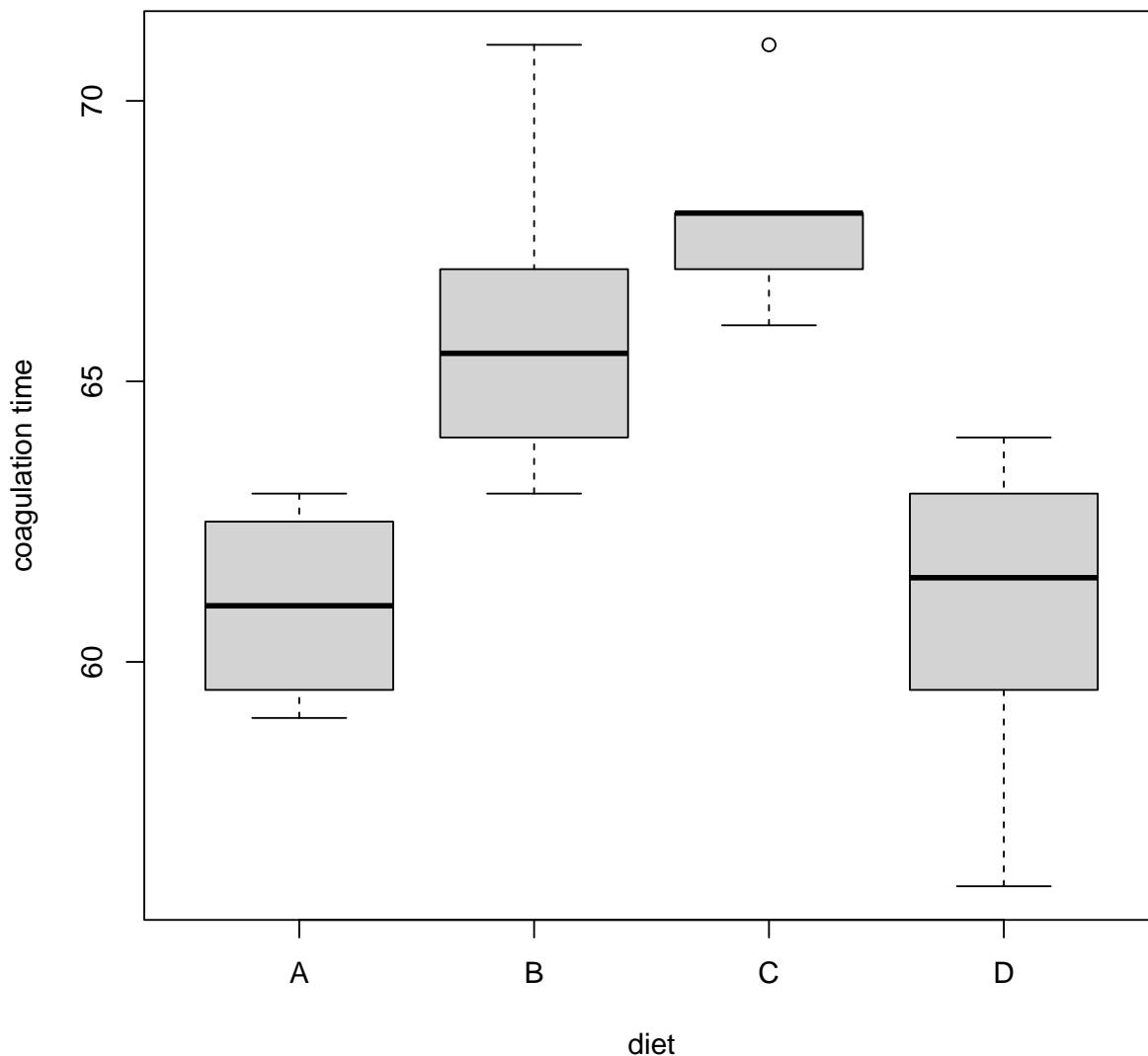
Motivating Example

For motivation, let's begin by looking at the blood coagulation times of $n_T = 24$ animals randomly assigned to one of $a = 4$ possible diets. (More on notation, shortly.)

```
> data(coagulation, package="faraway")
> stripchart(coag ~ diet, coagulation, vertical=TRUE,
+             method="stack", xlab="diet", ylab="coagulation time")
```



```
> plot(coag ~ diet, coagulation, ylab="coagulation time")
```



Notation & Initial Concepts

The above example motivates notation and concepts. In other words, use the previous plots to make concrete the following notation/concepts.

- **Obvious Differences.** Our notation differs a bit from [Far14].
- **Number of Factor Variables.** $\boxed{1}$; hence the name of this chapter! We may use generic notation to refer to this factor, e.g., Factor A. Again, factor is just another name a variable that's categorical (and we restrict our attention to covariate/input/predictor/regressor/x factors for now).
- **Multiple Factors.** We will consider multiple factor variables when we get to multiple factor models (multi-way ANOVA), later, where this generic notation will extend naturally: Factor A, Factor B, Factor C, etc., ([Far14, Chap. 16]).
- **Number of Factor Levels.** \boxed{a} . (Note, [Far14, §14.4] uses f to denote the number of levels, and [Far14, §15.1] uses I . So much for consistency of notation!)
- **Multiple Factors.** Again, when we get to multiple factor models (multi-way ANOVA), this notation will extend to indicate number of levels for each factor under consideration: a, b, c , etc.
- **Treatments.** Treatments are the set of conditions defined by the **unique combinations of factors levels across all factors**. Here, we have only **one factor**, so that factor levels and treatment levels are **synonymous**. This will not generally be the case for multiple factor models (higher-way ANOVA).
- **Observations.** $\boxed{Y_{ij}}$ is the j th observation (response) in the i th treatment level, $i = 1, \dots, a$, $j = 1, \dots, n_i$. (Notice how we index cases differently now from INF 511.)
- **Numbers of Observations.** Sample sizes or number of units per treatment level are denoted $\boxed{n_i}$, $i = 1, \dots, a$.
- **Total Number of Observations**, i.e., total samples size, is denoted as $\boxed{n_T}$, i.e., $n_T = n_1 + \dots + n_a = \sum_{i=1}^a n_i$.

- **Replicates.** If we have $n_i = 2$, $i = 1, \dots, a$, then we say the treatments have been replicated (twice). Of course, we can generally have $n_i \geq 1$, possibly **unbalanced** (or imbalanced) among treatments (see **Balance** below).
- **Balance.** If we have an equal number of observations per treatment level, i.e., if our treatments are replicated equally, then we say our treatment design is balanced. Otherwise it is unbalanced (or imbalanced). That is, $n_1 = n_2 = \dots = n_a = n$, where \boxed{n} is the common number of observations in each treatment. Thus, in the balanced case, $\boxed{n_T = na}$. (Note that we distinguish n from n_T .) Balance plays a relatively small role in single factor models (one-way ANOVA) compared to the multiple factor models (multi-way ANOVA), where it has historically received much more attention.
- **Observational or Experimental Treatment Level Means.** Denoted by $\boxed{\mu_i}$, $i = 1, \dots, a$. These are unknown parameters to be estimated. These are not averages of the observations, Y_{ij} . Again, for single factor models (one-way ANOVA), factor level means are synonymous with treatment level means.
- **What is an observational study? An experimental study?** (Recall INF 511 note chapter 5. See also [Far14, Chapter 5] and [KNNL05, Chapter 15].)

15.1.1 Cell Means Model

With the above example, notation and concepts, we can begin to see a model for our data.

- Compared to our previous models, the cell means model is as simple as things get.

$$Y_{ij} \stackrel{\text{ind}}{\sim} N(\mu_i, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, n_i$$

or, equivalently,

$$Y_{ij} = \mu_i + \epsilon_{ij} \quad \epsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, n_i.$$

The **model of the regression function (mean)** is

$$E(Y_{ij} | \mathbf{x}_{ij}) = \mu_i,$$

(the (often) vector of covariates, \mathbf{x}_{ij} , arises from numerical coding of the j th observation at the i th level of a factor variable (e.g., the j th animal in diet i) (in addition to any other covariates we might have); more on numerical coding as we go).

- **STAT 101.** You may notice this is the natural extension to the “pooled $a = 2$ sample t” model (“pooled” translates to an assumed constant variance across treatment levels).
- **Cell Means vs. SLR.** What would a corresponding plot of the error distributions and mean(s) look like for this model? For a simple linear regression (SLR) model? (in class)

- **Cell Means Model in Familiar, General Linear Model Form**

Happily, our matrix expressions from INF 511 apply! (I'll walk us through these details in class.)

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_{n_T})$$

- **Uncoded Factor Levels.** Let X denote an uncoded factor (categorical/classification/grouping variable) that takes on, e.g., $a = 4$ levels, 1,

2, 3, and 4, or diet A, diet B, diet C, and diet D. Thus, a vector of n_T observations X (yet to be coded), corresponding to the n_T observed responses/outputs, y_{ij} , may be look like, e.g., $(x_{11}, \dots, x_{ij}, \dots, x_{a,n_a})^t = (A, \dots, C, \dots, D)^t$. Simple, but not numeric.

- **Code the Factor Levels with Dummies.** To get the cell means model into our linear model form, we (or R) creates a **dummy variable** (aka **indicator variable** or **incidence variable**) for each level of a factor. (This is essentially the same as **one-hot encoding**.) That is, create a new covariates for each observation (for each j th observation at the i th level), $\mathbf{x}_{ij}^t = (x_{ij1}, \dots, x_{ijk}, \dots, x_{ija})$, such that

$$x_{ijk} = \begin{cases} 0 & \text{if } i \neq k \text{ (if not an obs. of level } k) \\ 1 & \text{if } i = k \text{ (if an obs. of level } k) \end{cases},$$

In other words, for each observation, create a new covariates, $X_1, \dots, X_k, \dots, X_a$, such that X_k indicates having observed the k th level with a 1, zero otherwise, $k = 1, \dots, a$. Thus, we will have a “indicator columns” in our \mathbf{X} matrix.

```
> ## coded variables (aside uncoded factor nominal levels)
> cbind.data.frame(model.matrix(~ 0+diet,data=coagulation),
+                    diet=coagulation$diet)
```

	dietA	dietB	dietC	dietD	diet
1	1	0	0	0	A
2	1	0	0	0	A
3	1	0	0	0	A
4	1	0	0	0	A
5	0	1	0	0	B
6	0	1	0	0	B
7	0	1	0	0	B
8	0	1	0	0	B

9	0	1	0	0	B
10	0	1	0	0	B
11	0	0	1	0	C
12	0	0	1	0	C
13	0	0	1	0	C
14	0	0	1	0	C
15	0	0	1	0	C
16	0	0	1	0	C
17	0	0	0	1	D
18	0	0	0	1	D
19	0	0	0	1	D
20	0	0	0	1	D
21	0	0	0	1	D
22	0	0	0	1	D
23	0	0	0	1	D
24	0	0	0	1	D

- **Compact Notation.** In INF 511, how would we denote the linear model for the regression function, $E(Y_{ij} | \mathbf{x}_{ij})$, using the coded values in \mathbf{x}_{ij} ? Answer (ignoring the column of 1's, i.e., ignoring the intercept for now):

$$\begin{aligned} E(Y_{ij} | \mathbf{x}_{ij}) &= \mathbf{x}_{ij}^t \boldsymbol{\beta} = x_{ij1}\beta_1 + \cdots + x_{ija}\beta_a \\ &= \beta_i, \end{aligned}$$

right? At this early point, we only begin to see why we use the compact notation of the observation-wise (cell means) model (using the more suggestive notation μ_i instead of β_i , of course). This sort of compact notation comes in handy when our (ANOVA) models include multiple factors, each with several levels, with possible interactions among factors, which introduces many coded variables and products thereof, but our familiar (if soon to become unwieldy) underlying linear model form may come in handy at times.

- **Redundancy.** If we created these *a* numerically coded variables by hand, and included these in a data set and then performed regression in R in the “familiar” way, with a column of 1’s for the intercept, we would have redundancy in our \mathbf{X} matrix. (See INF 511 (fall 2021) notes, Appendix B.2.6 for linear dependency, rank, redundancy among covariates, etc.).
- **R Deals with Redundancy**, but perhaps not in a way that we want. We know how to omit the column of 1’s (for the intercept parameter), right ? (Answer: +0 or -1 in the R formula.) If we don’t tell R to do this for the cell means model, R will deal with redundancy in a different manner than omitting the intercept, thus creating a different parameterization with parameter interpretation that does not correspond to the cell means model. More on other parameterizations, shortly. (By default, R would throw out the coded indicator column for the *first* level of the factor, leaving the column of 1’s. SAS users: SAS throws out the coded column for the *last* level of the factor.)

15.1.2 Cell Means Model (Example Continued)

Beware of Overall F Test (and R^2). This warning is related to that given in INF 511 (fall 2021) notes §2.9: when we ask R omit the intercept, R thinks our null model (no covariates) is the zero mean model (instead of constant mean model), and the overall F test and the R^2 values change and are likely misleading. The default tests in the summary output are fine, but are they terribly relevant?

```
> ## Notice the minus 1 to omit the column of 1's (for the intercept) to
> ## get the cell means model (works only for single factor models).
> lmod <- lm(coag ~ diet - 1, coagulation)
>
> ## Beware of Overall F test (and  $R^2$ )!
> summary(lmod)
```

```

Call:
lm(formula = coag ~ diet - 1, data = coagulation)

Residuals:
    Min     1Q Median     3Q    Max 
-5.00  -1.25   0.00   1.25   5.00 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
dietA      61.000    1.183    51.5 <2e-16 ***  
dietB      66.000    0.966    68.3 <2e-16 ***  
dietC      68.000    0.966    70.4 <2e-16 ***  
dietD      61.000    0.837    72.9 <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.37 on 20 degrees of freedom
Multiple R-squared:  0.999, Adjusted R-squared:  0.999 
F-statistic: 4.4e+03 on 4 and 20 DF,  p-value: <2e-16

> ## ATYPICAL! ANOVA table. Beware F!: 
> anova(lmod)

Analysis of Variance Table

Response: coag
          Df Sum Sq Mean Sq F value Pr(>F)    
diet       4  98532  24633    4399 <2e-16 ***  
Residuals 20    112        6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ## LS estimates. ("betahat" (now muhat notation))
> round(coef(lmod),1)

dietA dietB dietC dietD
  61    66    68    61

> ## BTW, LS parameter estimates of _treatment_ means are just cell averages.
> tapply(coagulation$coag, coagulation$diet, mean)

 A  B  C  D
61 66 68 61

```

- **Some Items for Discussion.** We will address the following items in class with the intent to remind us of our previous (511) matrix know-how and how that know-how still holds here. In short, we still have a linear model and 511 carries over here.
- **X Matrix.** What does the R \mathbf{X} matrix look like? (See previous output.)
- **LS Estimates.** What are the LS estimators/estimates of the μ_i ?
- **Estimated Error Variance.** What is the estimator/estimate of the variance, σ^2 (or of the standard deviation, σ)? Given that our model is just a linear regression model on $p = a = 4$ dummy variables, one for each level of diet, we use MSE (or root thereof) as usual, and realize that it is given by summary here in the same way as we've seen for linear regression (INF 511). (Again, we're essentially doing the same things whether we call them regressions or ANOVAs, though this may be a bit simplistic because there is a lot of specialized jargon, notation and output for ANOVA.)
- **Estimated Standard Errors.** What are the estimated standard errors of the estimators of the μ_i ?
- **Default t-tests.** R gives default t-tests for each of the parameters associated with the diet factor and assumes a null value of zero by default. How are these tests computed? Are these tests interesting in the current case of the cell means model?
- **p-values.** How are the p-values for the above tests computed? (Again, are these interesting?)
- **Etc.** What are the remaining quantities in the output of the summary function? (TO BE SURE: The R^2 and adjusted R^2 in the R output are not what we would expect, which you would know if you read `help(lm)`, which talks about what happens when we (R) omit the column of 1's from \mathbf{X} , as we did here. Recall our **WARNING** in INF 511 notes §2.9 Goodness of Fit. Also, the F-test is not what we might expect, either!)

Why? Answer: By throwing out the intercept (with $+0$ or -1 in the formula, R thinks our null model, without covariates, is not just a constant mean model, but zero mean model!)

We fix the problem with the overall F test by explicitly fitting a constant mean model that does not restrict that constant to be zero, then use or Full vs Reduced (F v R) approach via the `stats::anova` function.

```
> ## Explicitly fit reduced model for null of
> ## equal (common) means in overall F test
> lmodR<- lm(coag ~ 1, data=coagulation)
> summary(lmodR)

Call:
lm(formula = coag ~ 1, data = coagulation)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.00   -2.25  -0.50   3.00    7.00 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 64.000     0.785   81.5 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.84 on 23 degrees of freedom

> ## Usual F-test for equal means via F v R approach:
> anova(lmodR, lmod)

Analysis of Variance Table

Model 1: coag ~ 1
Model 2: coag ~ diet - 1
Res.Df RSS Df Sum of Sq    F    Pr(>F)
```

```

1      23 340
2      20 112  3      228 13.6 0.000047 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Or, we can use our linear combinations ($C\beta$) approach using the `gmodels::glh.test` function to specify that differences of means are equal, so that the means are the same, but we do not specify what the means are equal to. (We'll skip the "by hand" $C\beta$ approach that has us sport our matrix know-how for linear models (but see the numerous examples in INF 511).)

```

> ## Or Cbeta approach (via glh.test)
> Cmat<- matrix(c(1, -1, 0, 0,
+                  0, 1, -1, 0,
+                  0, 0, 1, -1),
+                  ncol=4, byrow=TRUE)
> b0<- rep(0,3) ## why 3?
> gmodels::glh.test(lmod, cm=Cmat, d=b0)

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = b0)
F = 13.571, df1 = 3, df2 = 20, p-value = 0.00004658

> ## ``By hand?'' You do it.

```

We can also fix the R^2 value, too, in the manner suggested in §2.9 of INF 511 notes, but, R^2 (even fixed) is **not as relevant** in this case; we are already fitting separate means for every different covariate value (every different level of the factor) that are not restricted to, e.g., fall on a line or parabola (as in 511) so our model cannot become any richer to somehow improve the fit to get a higher R^2 , our regression model is saturated. (more later in a more relevant context)

```
> ## More sensible yet relatively uninteresting goodness of fit
> ## R^2 value:
> cor(lmod$model$coag, fitted(lmod))^2
[1] 0.67059
```

15.1.3 Effects Model: Before Constraints

The cell means model is straight-forward for 1-way ANOVA models (with just a single factor input). But, there are other popular models, especially for higher-way ANOVA (with multiple factor inputs). (These “models” are often the same; only the parameterization and interpretation change. More as we go.)

- **Effects Model (before constraints):**

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij} \quad \epsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, n_i,$$

which suggest a common, alternative way to think of our treatment means as deviating about a common value.

Obviously, by definition of cell means, we have,

$$\mu_i \equiv E(Y_{ij} | \mathbf{x}_{ij}) = \mu + \alpha_i.$$

- **Overparameterized/Non-Identifiability/Redundancy.** As is, our mean (regression) model (effects model) is currently **overparameterized** (notice we have one more parameter than our previously discussed cell means model yet the same number of means being modeled). Another way to say this is that our mean model parameters are **not identifiable** or **not estimable**.

- **Same Mean, Arbitrary Parameter Values.** One way to see this non-identifiability is by adding and subtracting the same constant to the mean model (aka adding zero!). For example,

$$\begin{aligned} E(Y_{ij} | \mathbf{x}_{ij}) &= \mu + \alpha_i \\ &= (\mu + 5.73) + (\alpha_i - 5.73) \\ &\equiv \mu^* + \alpha_i^*. \end{aligned}$$

- **'Splain it To Me, Lucy.** That is, we get the same mean value, $E(Y | \mathbf{x})$, (hence same normal distribution) despite different parameter values. In other words, even if we knew the mean (or entire distribution), it cannot help us to identify uniquely a single parameter β , so how can we expect to estimate a unique β when we just have a finite number of observations from the distribution? In other words, our model cannot help us to identify particular values for μ and α_i , i.e., without a constraint on the parameters, the interpretation of the individual parameters is arbitrary. (Note that their sum $\mu + \alpha_i$ is identified: it's the i th cell mean, of course.)
- **Redundancy.** Yet another way to see this non-identifiability/redundancy is by considering the (now, non-identifiable) parameter vector

$$\boldsymbol{\beta} = (\mu, \alpha_1, \dots, \alpha_a)^T$$

and its associated \mathbf{X} matrix, constructed with a column of 1's for the constant, μ , and a (number or factor levels) columns of 0/1's, one for each α_i —a dummy (indicator) variable column for each α_i . (We discussed this sort of linear dependency or redundancy in \mathbf{X} in §B.2.6 of INF 511 notes 2021.)

Eliminate Redundancy

- **Not Back to Cell Means.** Once again, with redundant columns of \mathbf{X} , we could throw out the column of 1's associated with the μ (intercept)

parameter. Okay, but this would lead back to our **cell means** model parameterization and associated interpretation, not to the effects model with treatment coding and associated parameterization/interpretation (coming up shortly).

- **Remove that Dummy!** Another solution is to throw out a dummy variable for one of the levels of the (diet) factor.
- **Coding Synonyms.** This essentially dummy throwing amounts to what is known as **treatment coding** or **cell reference coding** (or **corner point coding** when we have more than one factor); the level/dummy variable that is thrown out is the **reference level** or **baseline level** or **reference treatment** (with 1 factor) or **corner point** (when one dummy from each of two or more factors are removed); more later.
- **Know Your Software.** Which dummy to throw out? By **default** R removes the dummy associated with the first level (and which one is R thinking that is?!), which corresponds to the **constraint** $\alpha_1 = 0$. Note to SAS users: SAS removes the dummy associated with the last level of a factor, corresponding to the constraint, $\alpha_a = 0$.

- **Cell Reference Coding Example.** In short, for cell reference coding of the effect model (resolving non-identifiability or redundancy), we code dummies just as in the cell means model, but we keep the column of 1's, and we throw out the first dummy. For example, for $a = 3$ levels, the X matrix looks like (e.g.)

(intercept)	X_2	X_3	uncoded X
1	0	0	level 1
1	0	0	level 1
1	0	0	level 1
1	1	0	level 2
1	1	0	level 2
1	1	0	level 2

- Then, perform linear regression as usual using the (now non redundant) predictor/input variables:

$$\begin{aligned} Y_{ij} &= \beta_0 + \beta_2 X_{ij2} + \beta_3 X_{ij3} + \epsilon_{ij} \\ &= \beta_0 + \text{beta}_i + \epsilon_{ij} \quad (\beta_1 = 0) \\ &= \mu + \alpha_i + \epsilon_{ij} \quad (\alpha_1 = 0) \end{aligned}$$

- We will consider alternative coding schemes, later.
- **Interpretation of parameters** for the (cell reference) coded variables.
 - β_2 (α_2) is the shift (up/down) (effect) in the regression model (surface, hyperplane) when moving from **reference level** $X = \text{level 1}$ ($\beta_0 + 0$ or $\mu + 0$) to $X = \text{level 2}$ ($\beta_0 + \beta_2$ or $\mu + \alpha_2$)
 - β_3 (α_3) is the shift (up/down) (effect) in the regression model (surface, hyperplane) when moving from **reference level** $X = \text{level 1}$ ($\beta_0 + 0$ or $\mu + 0$) to $X = \text{level 3}$ ($\beta_0 + \beta_3$ or $\mu + \alpha_3$)
 - $\beta_0 + 0$ ($\mu + 0$) corresponds to the “reference level” (or “corner-point”). Level 1 is the level to which all other levels of original X are compared. We can write $\mu + \alpha_1$, with $\alpha_1 = 0$ to remind ourselves that this constant (β_0 or μ) is the mean of the reference level (by default the first level in R).
- (We'll tend to drop the β notation...)
- **(Again) Parameter Interpretation Depends on Coding**:: There are several ways to numerically code categorical variables, and parameter interpretation depends on coding! See [Far14, §14.5] and [KNNL05, 8.4].

15.1.4 Effects Model: Treatment Coding/Constraint

- Thus, with our (R's) treatment coding of the effects model, our corresponding model is

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij} \quad \epsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, n_i,$$

now including the constraint,

$$\alpha_1 = 0.$$

Thus, we work with the remaining α_i , $i = 2, \dots, a$ (and μ) and the associated non-redundant (full column rank) \mathbf{X} . (See INF 511 for matrix rank, etc.)

- **As Before.** The mean of the observations within factor level 1 is $\mu_1 \equiv E(Y_{1j} | \mathbf{x}_{1j}) = \mu + 0$ so that μ is now the mean of the observations associated with the first ("reference/baseline") factor level.
- **Not the Overall Mean Despite Notation.** Generally, the (cell) mean of the observations within factor level i is

$$\mu_i \equiv E(Y_{ij} | \mathbf{x}_{ij}) = \mu + \alpha_i$$

so that α_i , $i = 2, \dots, a$, are the effects of being associated with the i th treatment level *relative* to the first (reference/baseline) treatment level whose mean is μ ; i.e., μ is not an overall mean effect in this treatment coding; i.e., $\alpha_1 = 0$.

- What is β in this case of cell reference coding of the effect model? (perhaps more in class) (perhaps this is obvious at this point)

- What is X in this case? (perhaps more in class) (obvious at this point?)
- We'll see how to implement this constraint/ X coding in R in a straightforward manner.

15.1.5 Effects Model: Treatment Coding/Constraint Example

```
> ## First, we should ensure that R sees factors as factors (yep.
> ## good.) Which level of the diet factor does R see as ``first?''
> data.class(coagulation$diet)

[1] "factor"

> ## Now, look at ``contrasts,'' which will tell us the coding R will use.
>
> ## What is the global contrasts (constraints/coding/parameterization)
> ## setting for factors? (Should have done this before cell means, above!)
>getOption("contrasts")

      unordered          ordered
"contr.treatment"      "contr.poly"

> ## Any contrasts set for the particular factor at hand?
> ## No. BTW, this would OVERRIDE the global contrasts setting.
> attr(coagulation$diet, which='contrasts')

NULL

> ## Next says R will code a dummy column in X for levels (diets) B, C, D
> ## (leaving the column of 1's to estimate reference level A mean mu + 0)
> contrasts(coagulation$diet)

  B C D
A 0 0 0
B 1 0 0
C 0 1 0
D 0 0 1

> ## See X matrix in a subsequent chunk.
```

```
> lmod<- lm(coag ~ diet, coagulation)
> ## (Overall F (and R^2) are back to normal now. Why?
> ## Are default tests interesting now?):
> summary(lmod)
```

Call:

```
lm(formula = coag ~ diet, data = coagulation)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.00	-1.25	0.00	1.25	5.00

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)							
(Intercept)	6.10e+01	1.18e+00	51.55	< 2e-16 ***							
dietB	5.00e+00	1.53e+00	3.27	0.00380 **							
dietC	7.00e+00	1.53e+00	4.58	0.00018 ***							
dietD	2.99e-15	1.45e+00	0.00	1.00000							

Signif. codes:	0	'***'	0.001	'**'	0.01	'*'	0.05	'. '	0.1	' '	1

Residual standard error: 2.37 on 20 degrees of freedom

Multiple R-squared: 0.671, Adjusted R-squared: 0.621

F-statistic: 13.6 on 3 and 20 DF, p-value: 0.0000466

```
> ## X matrix aside nominal factor (treatment) levels
> cbind.data.frame(model.matrix(lmod), diet=coagulation$diet)

(Intercept) dietB dietC dietD diet
1           1     0     0     0     A
2           1     0     0     0     A
3           1     0     0     0     A
4           1     0     0     0     A
5           1     1     0     0     B
6           1     1     0     0     B
7           1     1     0     0     B
8           1     1     0     0     B
9           1     1     0     0     B
10          1     1     0     0     B
11          1     0     1     0     C
```

12	1	0	1	0	C
13	1	0	1	0	C
14	1	0	1	0	C
15	1	0	1	0	C
16	1	0	1	0	C
17	1	0	0	1	D
18	1	0	0	1	D
19	1	0	0	1	D
20	1	0	0	1	D
21	1	0	0	1	D
22	1	0	0	1	D
23	1	0	0	1	D
24	1	0	0	1	D

- **X Matrix.** What does the R **X** matrix look like? (see output!)
- **LS Estimates.** What are the LS estimators/estimates of the parameters?
- **Interpretation.** How do we interpret these (estimated) parameters?
- **Compare to Previous Model.** How do these estimates compare to those given for the cell means model, previously? ($\widehat{\mu + \alpha_i} = \widehat{\mu}_i$)
- **Estimated Error Variance.** What is the estimator/estimate of the variance, σ^2 (or of the standard deviation, σ)? How does this compare to the estimate given by the cell means analysis? (Same.)
- **Estimated Standard Errors.** What are the estimated standard errors of the estimators of the parameters?
- **Default t-tests.** R gives default t-tests for each of the parameters associated with the diet factor and assumes a null value of zero by default. How are these tests computed? Are these tests interesting?
- **p-values.** How are the p-values for the above tests computed? (Again, are these interesting?)

- **Etc.** What are the remaining quantities in the output of the `summary` function? Unlike the automatic overall F-test given with the cell means analysis, the overall F-test here *is* for equality of means without constraining the common mean to be zero (all $\alpha_i = 0$, all i , right?), and the R^2 is okay.
- **Scope of inference** does not change with different coding / parameterization. (See unnumbered section, Scope of Inference: Summary, at the end of INF 511 note chapter 5.)

- **Mimicking Above Cell Means Analysis.** Though we have a “good” overall F-test for equality of means given in the output, above, still we mimic our previous, cell means presentation by showing the F v. R approach to the overall F-test for equality of means and the linear combinations approach, which have slight changes here to accomodate the treatment constraint/coding. See following chunks.

```
> ## Good sequential anova overall F test (unlike cell means model anova)
> anova(lmod)

Analysis of Variance Table

Response: coag
          Df Sum Sq Mean Sq F value    Pr(>F)
diet        3   228    76.0    13.6 0.000047 ***
Residuals  20   112     5.6
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## F v R approach for overall F test (never an issue here)
> lmodR<- lm(coag ~ 1, coagulation)
> anova(lmodR, lmod)
```

Analysis of Variance Table

```

Model 1: coag ~ 1
Model 2: coag ~ diet
  Res.Df RSS Df Sum of Sq    F    Pr(>F)
1      23 340
2      20 112  3      228 13.6 0.000047 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

> ## Or Cbeta approach (bit different C matrix compared to cell means).
> Cmat<- matrix(c(0, 1, 0, 0,
+                  0, 0, 1, 0,
+                  0, 0, 0, 1),
+                  ncol=4, byrow=TRUE)
> d<- rep(0,nrow(Cmat))
> gmodels::glh.test(lmod, cm=Cmat, d=d)

```

```

Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 13.571, df1 = 3, df2 = 20, p-value = 0.00004658

> ## You do it ``by hand.''

```

- **A Natural Reference Level?** Sometimes, one level may seem to be a natural reference level compared to other levels (e.g., placebo, standard treatment, etc.).
- **A Change to Reference Level.** To illustrate, we change the reference level to Diet D (not that this intuits a natural reference level in the current example).

```
> ## Change reference level
> coagulation$diet<- relevel(coagulation$diet, ref="D")
> ## Now we see that R will code columns in X indicating dietA-DietC but
> ## not dietD, now the reference level
> contrasts(coagulation$diet)
```

	A	B	C
D	0	0	0
A	1	0	0
B	0	1	0
C	0	0	1

```
> ## And, we see the corresponding change in LS model estimates.
> summary(lm(coag ~ diet, coagulation))
```

Call:

```
lm(formula = coag ~ diet, data = coagulation)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.00	-1.25	0.00	1.25	5.00

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	6.10e+01	8.37e-01	72.91	< 2e-16 ***
dietA	4.38e-15	1.45e+00	0.00	1.00000
dietB	5.00e+00	1.28e+00	3.91	0.00086 ***
dietC	7.00e+00	1.28e+00	5.48	0.000023 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.37 on 20 degrees of freedom

Multiple R-squared: 0.671, Adjusted R-squared: 0.621

F-statistic: 13.6 on 3 and 20 DF, p-value: 0.0000466

```
> ## You should be able to modify the previous Cbeta approach (omitted here)
```

```
> ## We switch back to previous level order before proceeding.
```

```
> coagulation$diet<- factor(coagulation$diet, levels=c("A", "B", "C", "D"))
```

15.1.6 Effects Model: Sum (to Zero) Coding/Constraint

- Another (common) resolution to non-identifiability/redundancy is to impose the constraint $\sum_{i=1}^a \alpha_i = 0$.
- Thus, our model is then

$$Y_{ij} = \mu + \alpha_i + \epsilon_{ij} \quad \epsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, n_i,$$

now including the (sum-to-zero) constraint,

$$\sum_{i=1}^a \alpha_i = 0.$$

- **Now μ is Overall Mean.** Notice that this way of defining the mean model in the effects parameterization, *with sum-to-zero constraint*, makes μ interpreted as an overall mean (not an arbitrary, unidentified constant, before we imposed constraints, and not a reference level mean as in treatment coding/constraint).

How Do We See This? We have means

$$\mu_i = \mu + \alpha_i,$$

so that taking the mean (i.e., average over i) of the μ_i , and using the sum-to-zero constraint, shows that

$$\mu$$

is the **overall mean**.

- **Now Deviation from Overall Mean.** And,

$$\alpha_i = \mu_i - \mu$$

is the deviation of the i th treatment mean from the overall mean, μ , not just a deviation from some arbitrary constant level, μ , or from a reference level mean as in treatment coding. Of course, with the effects model and treatment constraints, the α_i are deviations from μ , but μ , under the treatment constraints, is not the overall mean, but the reference treatment mean.

- **Can Back Out an Effect.** With the sum-to-zero constraint, we have

$$\alpha_a = - \sum_{i=1}^{a-1} \alpha_i.$$

That is, we can work with (e.g.) the first $(a - 1)$ α_i parameters (and the parameter μ) and solve for α_a if we need to. We could have chosen any other α_i to solve for, but R solves for the last level's effect, α_a . Remember, know your levels.

- Thus, we can throw at the last (not first) dummy and change the other dummies to indicate the last level with -1's.
- What is β in this case?
- What is \mathbf{X} in this case?
- We'll see how to implement this constraint/ \mathbf{X} coding in R in a straightforward manner.

15.1.7 Effects Model: Sum to Zero Coding/Constraint Example

```
> ## Once again, we ask, `What is the global contrasts setting?'
>getOption("contrasts")
```

unordered "contr.treatment"	ordered "contr.poly"
--------------------------------	-------------------------

```

> ## Any contrasts set for the particular factor at hand?
> ## (No. BTW, this would OVERRIDE the global contrasts setting.)
> attr(coagulation$diet, which='contrasts')

NULL

> ## Evidently, next shows _global_ setting if contrast attribute is
> ## NULL (as above).
> contrasts(coagulation$diet)

  B C D
A 0 0 0
B 1 0 0
C 0 1 0
D 0 0 1

> ## Need to change coding/contrains/contrasts either locally...
> contrasts(coagulation$diet)<- contr.sum(levels(coagulation$diet))
> attr(x=coagulation$diet, which="contrasts") ## now have local setting...

 [,1] [,2] [,3]
A     1     0     0
B     0     1     0
C     0     0     1
D    -1    -1    -1

>getOption("contrasts") ## ...different from global setting...

      unordered          ordered
"contr.treatment"      "contr.poly"

> contrasts(coagulation$diet) ## ...and now local overrides global...

 [,1] [,2] [,3]
A     1     0     0
B     0     1     0
C     0     0     1
D    -1    -1    -1

> ## ...or change contrasts globally for factors not having a local setting
> options(contrasts = c("contr.sum", "contr.poly"))
> getOption("contrasts") ## global change

[1] "contr.sum"  "contr.poly"

```

```
> contrasts(coagulation$diet) ## still shows local (now same as global)

 [,1] [,2] [,3]
A     1   0   0
B     0   1   0
C     0   0   1
D    -1  -1  -1

> ## Above says R will code (almost) an indicator column for
> ## each of the first (a-1) = diet levels (A,B,C), but all columns ALSO
> ## `indicate' the last level a (D) with -1.
>
> ## See X matrix in a subsequent chunk.
```

```
> lmod<- lm(coag ~ diet, coagulation)
> ## (Overall F and R^2 okay for stz coding (not for cell means)):
> summary(lmod)
```

Call:

```
lm(formula = coag ~ diet, data = coagulation)
```

Residuals:

Min	1Q	Median	3Q	Max
-5.00	-1.25	0.00	1.25	5.00

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	64.000	0.498	128.54	< 2e-16 ***
diet1	-3.000	0.974	-3.08	0.00589 **
diet2	2.000	0.845	2.37	0.02819 *
diet3	4.000	0.845	4.73	0.00013 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.37 on 20 degrees of freedom

Multiple R-squared: 0.671, Adjusted R-squared: 0.621

F-statistic: 13.6 on 3 and 20 DF, p-value: 0.0000466

```
> ## X matrix aside nominal factor (treatment) levels
> cbind.data.frame(model.matrix(lmod), diet=coagulation$diet)
```

	(Intercept)	diet1	diet2	diet3	diet
1	1	1	0	0	A
2	1	1	0	0	A
3	1	1	0	0	A
4	1	1	0	0	A
5	1	0	1	0	B
6	1	0	1	0	B
7	1	0	1	0	B
8	1	0	1	0	B
9	1	0	1	0	B
10	1	0	1	0	B
11	1	0	0	1	C
12	1	0	0	1	C
13	1	0	0	1	C
14	1	0	0	1	C
15	1	0	0	1	C
16	1	0	0	1	C
17	1	-1	-1	-1	D
18	1	-1	-1	-1	D
19	1	-1	-1	-1	D
20	1	-1	-1	-1	D
21	1	-1	-1	-1	D
22	1	-1	-1	-1	D
23	1	-1	-1	-1	D
24	1	-1	-1	-1	D

- **X Matrix.** What does the R **X** matrix look like? (see output!)
- **LS Estimate.** What are the LS estimators/estimates of the parameters?
- **Interpretation.** How do we interpret these (estimated) parameters?
- **Compare to Previous Models.** How do these estimates compare to those given for the cell means model, previously? To those of the treatment coding model? ($\widehat{\mu} + \widehat{\alpha}_i = \widehat{\mu}_i$, where $\widehat{\mu} + \widehat{\alpha}_i$ can refer to the effects model with treatment coding or with sum-to-zero coding).

- **Estimated Error Variance.** What is the estimator/estimate of the variance, σ^2 (or of the standard deviation, σ)? How does this compare to the estimate given by the cell means and treatment coding analyses? (Same.)
- **Estimated Standard Errors.** What are the estimated standard errors of the estimators of the parameters?
- **Default t-tests.** R gives default t-tests for each of the parameters associated with the diet factor and assumes a null value of zero by default. How are these tests computed? Are these tests interesting?
- **p-values.** How are the p-values for the above tests computed?
- **Etc.** What are the remaining quantities in the output of the `summary` function? Unlike the automatic overall F-test given with the cell means analysis, the overall F-test here *is* for equality of means (all $\alpha_i = 0$, all i , right?), and the R^2 is okay (as F and R^2 were okay (and exactly the same) for the treatment coding analysis).
- **Scope of inference** does not change. (See unnumbered section, Scope of Inference: Summary, at the end of INF 511 note chapter 5.)

- **Mimicking Above Analyses.** Though we have a “good” overall F-test for equality of means given in the output above, still we mimic our cell means and treatment coding analyses, above, by showing the F v. R approach to the overall F-test for equality of means and the linear combinations approach, which have slight changes here to acknowledge the sum-to-zero constraint/coding. See following chunks.

```
> ## Reduced model (same as before) for overall F-test of equal means.  
> lmodR<- lm(coag ~ 1, coagulation)
```

```
> ## Usual F-test for equal means via F v R approach:
> anova(lmodR, lmod)

Analysis of Variance Table

Model 1: coag ~ 1
Model 2: coag ~ diet
  Res.Df RSS Df Sum of Sq    F    Pr(>F)
1      23 340
2      20 112  3      228 13.6 0.000047 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## Same Cmat as for treatment coding but picks off different
> ## parameters, right?
> Cmat<- matrix(c(0, 1, 0, 0,
+                  0, 0, 1, 0,
+                  0, 0, 0, 1),
+                  , ncol=4, byrow=TRUE)
> d<- rep(0,nrow(Cmat))
> gmodels::glh.test(lmod, cm=Cmat, d=d)
```

```
Test of General Linear Hypothesis
Call:
gmodels::glh.test(reg = lmod, cm = Cmat, d = d)
F = 13.571, df1 = 3, df2 = 20, p-value = 0.00004658
```

15.1.8 Regression Approach to ANOVA

- **Constraints for Full Rank X.** Notice that imposing constraints led us to an \mathbf{X} matrix that is full rank, just like in the regression modeling we've done so far (usually), whereby we can get the usual LS solution for β (i.e., $\mathbf{X}'\mathbf{X}$ is invertible).
- **Regression Matrix.** Each column of the resulting \mathbf{X} matrix can be thought of as the observations of a regression variable (with specially coded values).

- **Regression Approach to ANOVA.** Such coding of regression variables is discussed in textbooks under such headings as “Regression Approach to ANOVA” or something similar. See., e.g., [KNNL05, Sec. 16.7, 16.8] for an illustration using the effects parameterization with sum-to-zero constraint/coding.
- **Embarrassing.** I once had an instructor with a PhD in statistics who didn’t realize that regression and ANOVA are fundamentally the same thing. (Granted, the nature of ANOVA makes it seem a lot different than regression.)

15.1.9 Model, Parameterization, Reparameterization, Coding, Constraints

- **Same Model.** Technically, we have treated only a single model in this section so far (excluding the null model without covariates except for the column of 1’s), despite our terminology suggesting that we have different models: cell means model; factor effects model with treatment coding; factor effects model with sum-to-zero coding.
- **Technically.** Each give the same mean, i.e., same regression function model: technically the columns of their \mathbf{X} matrices span the same space, but the columns constitute a different coordinate system (basis) for that space with different coefficients (our regression model parameters) having different interpretations, thus $\mathbf{X}\boldsymbol{\beta}$ is the same across these “models” though details of \mathbf{X} and $\boldsymbol{\beta}$ differ.
- **Less Technically.** We’ve seen the necessary consequences of the technical equivalence of these mean (regression) models: same MSE ($\hat{\sigma}^2$), same ANOVA tables, same overall F-test and same R^2 , same results for inferences of treatments means (notwithstanding the strangeness of F and R^2 arising from our omitting the 1’s column in the cell mean model

(in R anyway)). Also, fitted values, residuals, hat matrix, etc., are the same.

- **Free to Choose.** You are free to choose the parameterization (constraint/coding) though a model's usefulness or convenience may depend on circumstances that we will not likely develop an appreciation for. Generally and loosely speaking, for well behaved ANOVA situations, one of the effects models are typically used. For messier data situations, we often fall back to a cell means model.

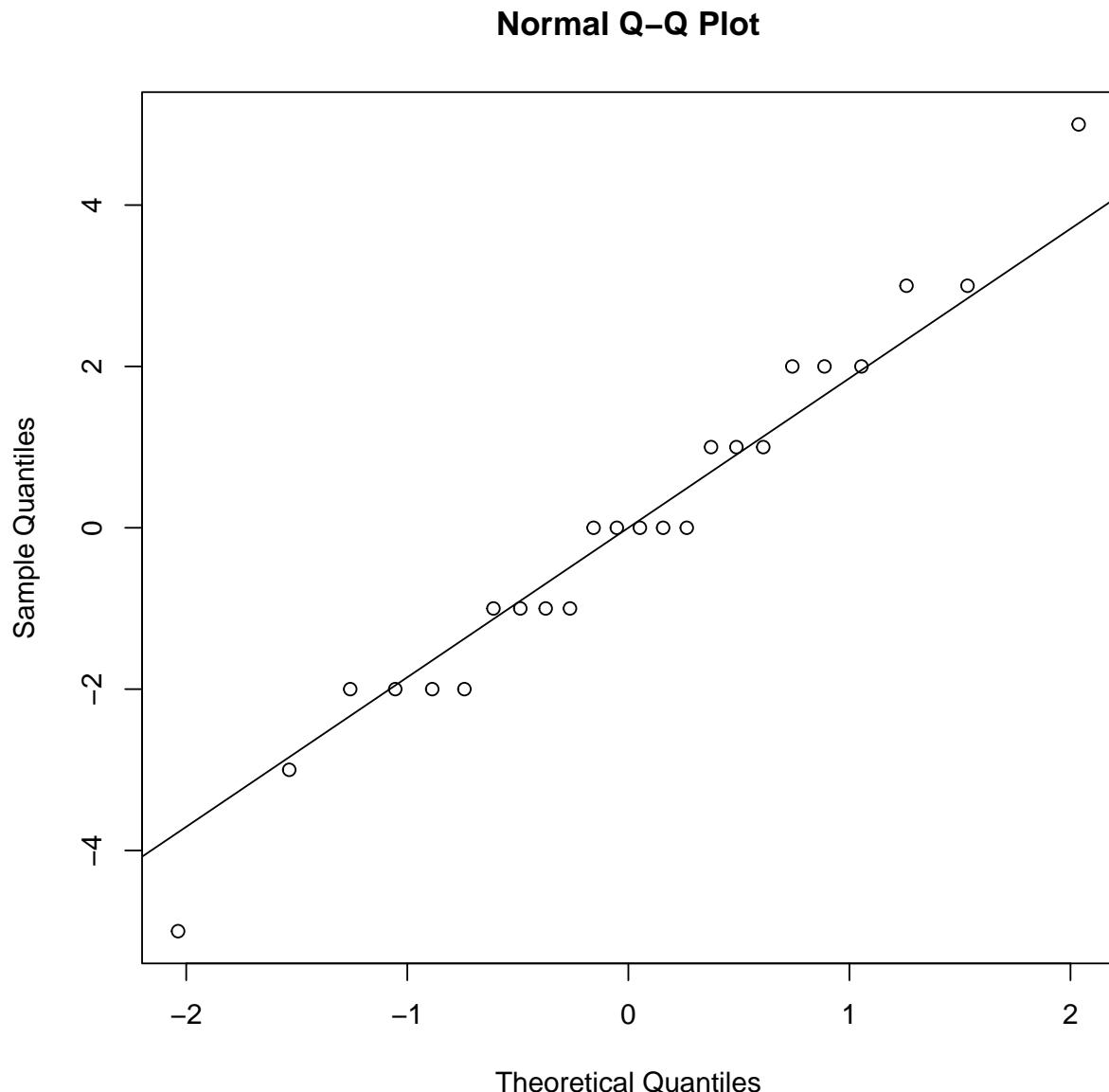
15.2 An Example (NOT)

The textbook material for this section [[Far14](#), §15.2] was integrated into the previous note section. (Again, we're still sort of maintaining subsection numbering to correspond to [[Far14](#)], but, again, this will cease soon.)

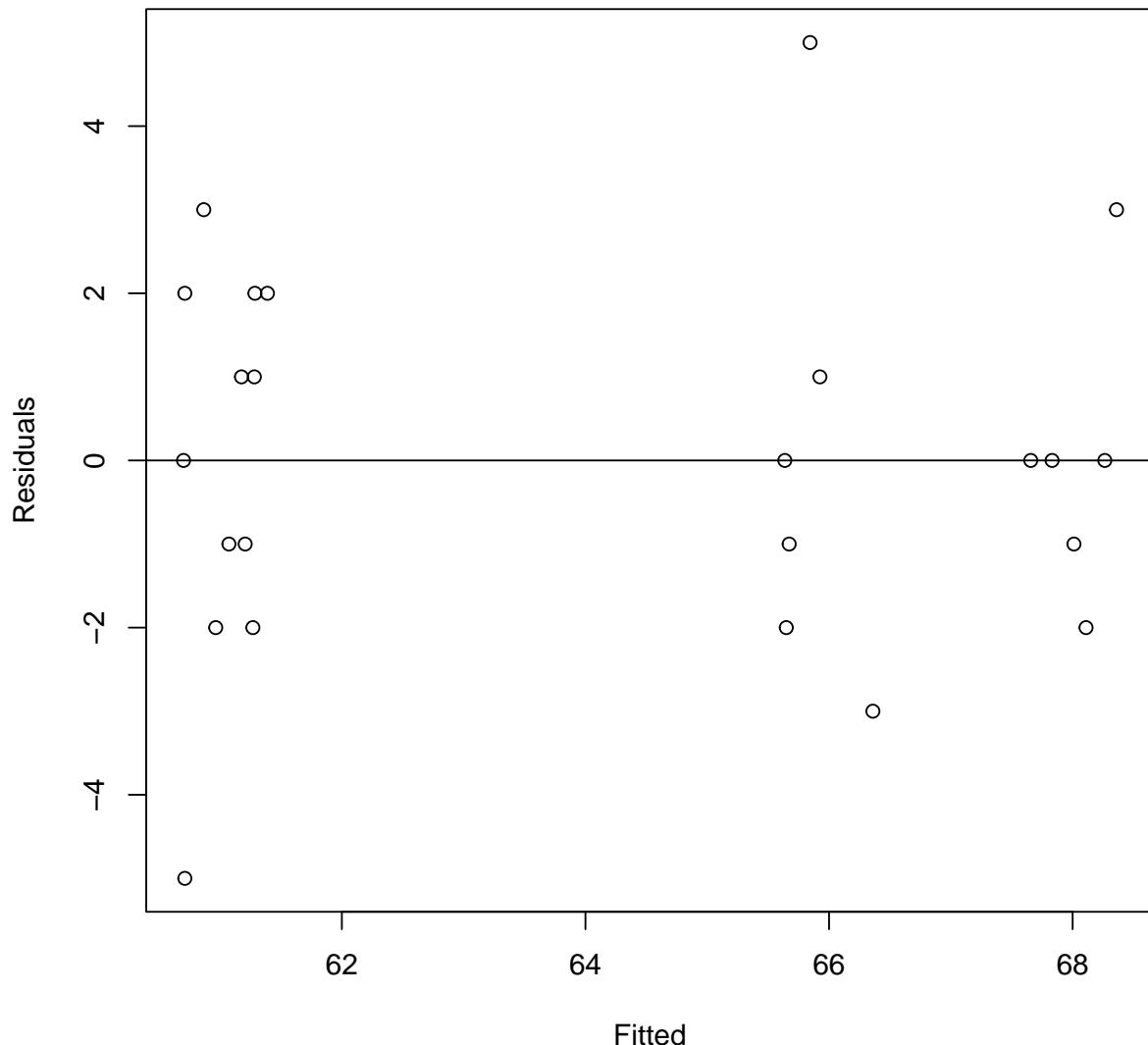
15.3 Diagnostics

- **Again, Same Model.** As just alluded to, it does not matter which parameterization/coding/constraint(s) we use for diagnostics. We use the latest created objects (effects model with sum-to-zero coding, I think).
- **No Problems.** In short, there do not appear to be obvious departures from our assumptions.
- **Discussion.** We'll discuss the following, hopefully familiar, diagnostics in class.

```
> ## Typical graphical diagnostics.  
> qqnorm(residuals(lmod))  
> qqline(residuals(lmod))
```



```
> plot(jitter(fitted(lmod)), residuals(lmod), xlab="Fitted", ylab="Residuals")
> abline(h=0)
```



```
> ## Lavene's Test: ANOVA: ``/y - med/ ~ diet''. We called this the
> ## Brown-Forsyth test or modified Lavene's test in 511 note chapter 6.
> ## Robust to outliers and non-normality. (My 511 bf.test function is
> ## at the chunk end. Same result.) How are our groups defined for BF?
> ## (are you awake?)
>
> ## Compute median response by diet
> med <- with(coagulation,tapply(coag,diet,median))
```

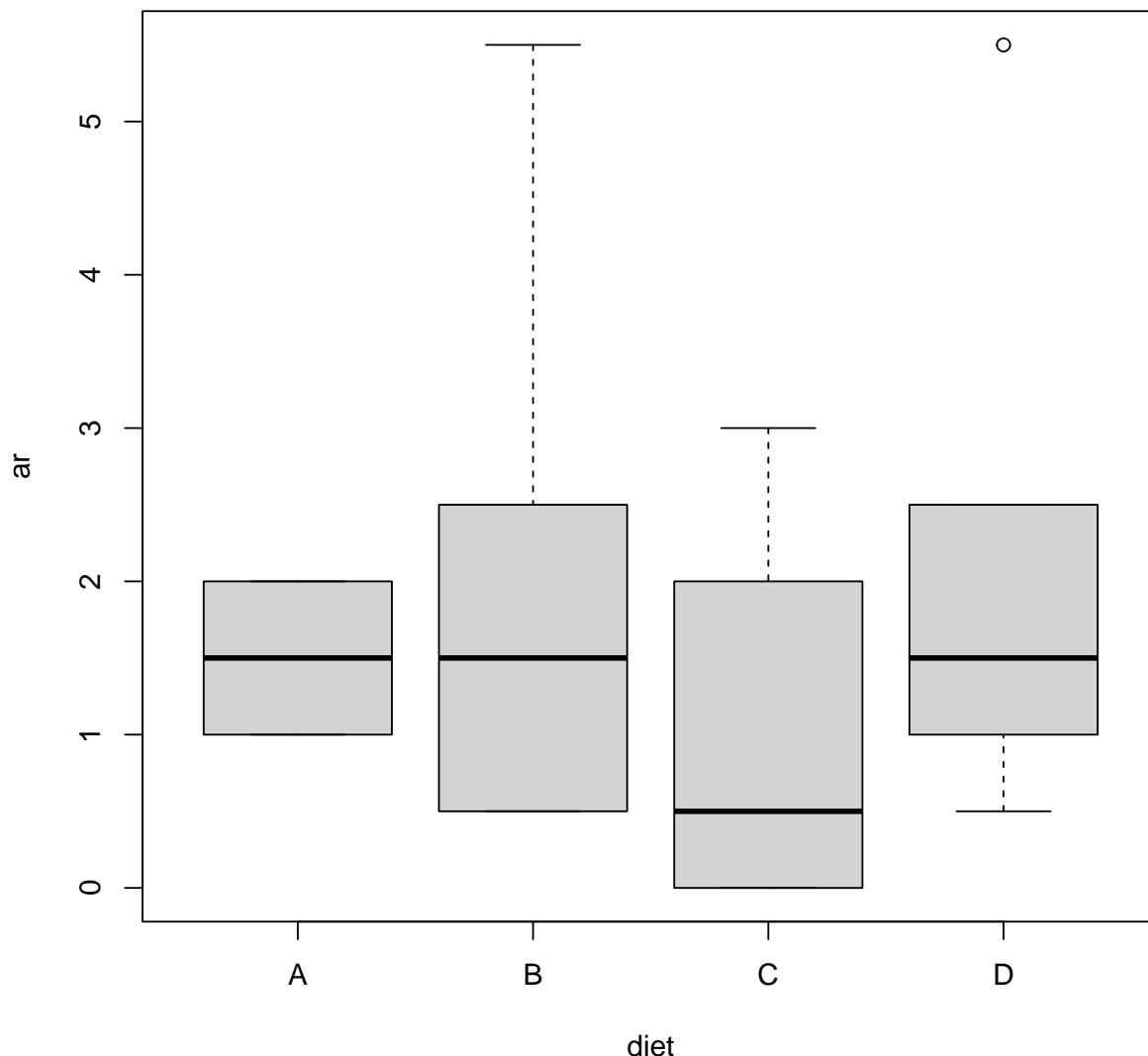
```
>  
> ## Subtract medians from responses (detrend) and make positive so  
> ## that differences among diets are now attributed to differences in  
> ## variability...  
> ar <- with(coagulation,abs(coag -med[diet]))  
>  
> ## ...which we will try to detect formally with ANOVA.  
> anova(lm(ar ~ diet,coagulation))
```

Analysis of Variance Table

Response: ar

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
diet	3	4.3	1.44	0.65	0.59
Residuals	20	44.5	2.23		

```
> ## Graphically:  
> plot(ar ~ diet, coagulation)
```



```
> ## Another test (less robust to outliers)
> bartlett.test(coag ~ diet, coagulation)

Bartlett test of homogeneity of variances

data: coag by diet
Bartlett's K-squared = 1.67, df = 3, p-value = 0.64
```

```
> ## Our BF (modified Lavene's) test from note chapter 6.  
> ## Same as your textbook author's Lavene's test, above.  
> bf.test<- function(x,groups)  
+ {  
+   medians <- sapply(split(x,groups), median, na.rm = TRUE)  
+   dij <- abs(x - rep(medians, table(groups)))  
+   anova(lm(dij ~ groups))  
+ }  
> bf.test(residuals(lmod), groups=coagulation$diet)
```

Analysis of Variance Table

Response: dij

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
groups	3	4.3	1.44	0.65	0.59
Residuals	20	44.5	2.23		

15.4 Pairwise Comparisons

- **Not Much So Far.** So far, we might argue that we have not done much with our previous data analysis: just an overall F test and some diagnostics.
- **We Want More.** We typically want to make more detailed comparisons among means, i.e., infer about more interesting linear combinations of the mean model, which translates to linear combinations of the regression parameters $\mathbf{C}\beta$. (Deja vu INF 511 stuff all over again.)
- **But, What Do We Want?** Failing intuition or insight or preconceptions about interesting linear combinations about which to infer, pairwise comparisons among treatment (factor level) means may be seen as a sort of default first step toward more detailed analyses.
- **How Many Pairwise Comparisons?** $(4 \text{ choose } 2) = 6$, for the diet example. More generally $(a \text{ choose } 2)$.

- **Type I Error Anxiety (Multiple Comparison Anxiety).** We might simply forge ahead to estimate all pairwise differences of means (using, e.g., our $C\beta$ approach with `gmodels::estimable`), but your author takes this opportunity to worry about Type I errors. (We briefly talked about Bonferroni in 511.)
- **The Basic Fear:** The more tests we do, each at a **comparison-wise** or **individual-wise** error rate, α , the more likely we are to make at least one Type I error among the **family** of tests performed, even if all null hypotheses are true (in the current case, even if all pairwise differences are zero). A similar fear exists about the confidence levels and coverage rates of intervals.
- **It's about Time!** I often felt that we should have featured this real problem (not just anxiety) surrounding the errors involved with multiple comparisons before now, back in INF 511. But, your author waits to discuss multiple comparisons until now, so I waited, too. TBD: add a bit of this error problem to INF 511, before chapter 14 (!), including looking at/editing data and other potential problems that may go by various names: pre-processing, data cleaning, data wrangling, data snooping, p-hacking, trying and trying again...and being more than honest (Richard Feynman via Mayo). Potti and Nevins, and other Mayo examples (crisis in science panels, etc.)
- **Value Added.** We add some material to [Far14, §15.4], including a different section heading.

Simultaneous Inferences & Multiple Comparison Procedures

- **Basic Concepts Before Proceeding.** We introduce basic concepts before proceeding with pairwise comparison and other multiple comparison procedures.

- **Simultaneous Inferences.** By simultaneous inferences we mean **conducting multiple tests and/or constructing multiple confidence intervals/regions**. Such simultaneous inferences are often referred to as **multiple comparisons**.
- **Our Old Friend $C\beta$.** Our inferences, i.e., our “comparisons,” may be phrased in terms of a linear combination of our regression function parameters ($C\beta$). So, in what follows, we consider tests and/or CI’s for these multiple, or simultaneous, $C\beta$ ’s. We phrase much of our discussion assuming C to be a matrix with 1 row ($r = 1$) so that $C\beta$ is a scalar quantity, but concepts extend to C with multiple rows (multiple comparisons of multiple comparisons in some sense).
- **What’s the “problem” with multiple comparisons?** Stay tuned.

Confidence Intervals

Definition 15.1 (Individual (Comparison-wise) Confidence Level). *The individual confidence level for a single interval about a $C\beta$ (assumed scalar) value is the rate at which a procedure for constructing confidence intervals has its resulting interval successfully cover the true value of $C\beta$, i.e.,*

$$1 - \alpha = \lim_{\# \rightarrow \infty} \frac{\text{number of intervals covering the true } C\beta \text{ value}}{\# \text{ of intervals constructed}}$$

- **Coverage Rate.** For confidence intervals that we’ve constructed so far, we’ve referred to this as coverage rate, or **confidence level**, denoted as $(1 - \alpha)$.
- **Now Qualified.** We now qualify this $(1 - \alpha)$ as an *individual* or

comparison-wise confidence level because it refers to only a single confidence interval, one “comparison” so-to-speak.

- **Long-Run Relative Frequency (Frequentist) Interpretation.** This is just the frequentist long-run interpretation of probability, but we don’t call it probability because we have one fixed interval and one fixed unknown target; no remaining randomness; no remaining distribution; no probability; the interval contains $C\beta$ or it doesn’t. Instead, we say “confidence” to indicate our (un)certainty.

Tests

Definition 15.2 (Individual (Comparison-wise) Error Rate). *The individual error rate is the rate at which a testing procedure for performing a single test would falsely reject the null hypothesis of the test (hence the name “error” rate—Type I error rate).*

$$\alpha = \lim_{\# \rightarrow \infty} \frac{\text{number of tests falsely rejecting the null}}{\# \text{ of tests}}$$

- **Type I Error Rate Now Qualified.** We have been calling this (Type I error) rate or level α , but now qualify it as a *comparison-wise* α .

Hypothetical Replications

- **Hypothetical Replications.** The above definitions are akin to our familiar, fanciful, hypothetical notion of repeated experiments or repeated

sampling studies; for each hypothetical and identical study—identical except for the data—we construct a confidence interval and/or do a test.

- **Long-Run Rates or Proportions.** We do this repeatedly (hypothetically)—many, many times—and, in the long run, the coverage rate or the proportion of intervals that include the true $C\beta$ approaches $1 - \alpha$, and the proportion of falsely rejected null hypotheses approaches α .
- **Often Hidden.** When using a mathematical model (e.g., normal or t or F), we often forget this fundamental frequentist notion, but, again, we saw it directly when discussing the randomization distribution and sampling distribution (INF 511).
- **Sometimes Not Strictly Applicable.** E.g., global average temperature tomorrow. Uncertain? Yes. Can we reasonably characterize our uncertainty with a long-run relative frequency, i.e., frequentist, interpretation of probability? In the beginning... Still, the error and coverage rates say something about our procedures being wrong, and we, as scientists want to make sure that we are extra carefull about being the ways in which we could be wrong or misconstrued.

Example: Confidence Intervals

Example 15.1 (What's “Wrong” with 100 Confidence Intervals?). Suppose you constructed a family (i.e., group) of 100 different confidence intervals for 100 different $C\beta$'s, and suppose for simplicity you chose the individual levels to be the same value, say $1 - \alpha = 0.90$.

- If all of the above intervals are independent—unlikely—then how many of these intervals in a long run average sense would contain its corresponding

true $C\beta$ value? About 90 out of 100 or 90%, right? This is the idea of the individual (comparison-wise) confidence level. It's what we've been using all along. Nothing new here.

- What's the probability that all 100 intervals **simultaneously** contain their respective true $C\beta$ value? (We say "probability" assuming we have not yet constructed the intervals so that intervals are still random before we plug in data to fix them, which would cause us then to say "confidence.") This is difficult to say, but, *assuming* the confidence intervals are independent—again, unlikely—then basic rules of probability give $(1 - \alpha)^{100} = .9^{100} =$ about 27 out of 100000 (hypothetical replications)—small confidence!!! In other words, if you wanted to make a statement of confidence about the *family* of intervals all *simultaneously* containing their true values, your—let's say *family-wise*—confidence level is tiny! Often, you may want to make this *family-wise* confidence level high. Thus, we now seek methods to ensure that a nominal confidence level now applies to an entire *family* of intervals so we can make statements like, "We are 95% confident that all 100 intervals *simultaneously* contain their respective true $C\beta$ values."

Family-wise Confidence Level

Definition 15.3 (Family-wise Confidence Level). *The family-wise confidence level for a group of k confidence intervals about their respective $C\beta$ values is the rate at which all k confidence intervals simultaneously successfully cover their respective true values of $C\beta$ in repeated sampling or experimentation.*

$$\lim_{\# \rightarrow \infty} \frac{\text{number of interval families of size } k \text{ covering all of their true } C\beta \text{ values}}{\# \text{ of size } k \text{ interval families constructed}}$$

- **New Notion of Confidence Level.** This is a new notion of confidence level that now applies to a group, i.e., *family*, of intervals, to multiple inferences, multiple “comparisons”.
- **Long-Run Relative Frequency of Family Coverage.** Note we still use the same fanciful notion of repeated sampling as an interpretation of the confidence level, except now we hypothetically repeat the entire family of k intervals many, many times.

Family-wise Error Rate

Definition 15.4 (Family-wise Error Rate (FWER)). *The family-wise error rate is the rate at which at least one member of a family of tests falsely rejects its null hypothesis in repeated sampling or experimentation.*

$$\lim_{\#\text{ of families} \rightarrow \infty} \frac{\text{number of families with at least one test falsely rejecting its null}}{\#\text{ of families}}$$

Bonferroni Inequality & Multiple Comparison Procedure

Skip Details?

The probability result of the “What’s Wrong with 100 Confidence Intervals?” example, above, assumed independence of intervals—once again, unlikely—to arrive at some family-wise level of confidence. There is another rule of probability that does not require any assumption about the dependence of intervals that can be of use to us. If you were to construct k intervals, each with its own individual (comparison-wise) confidence level $1 - \alpha_j$, then the

family-wise confidence level must be at least as large as $1 - \sum_{j=1}^k \alpha_j$. That is,

$$(\text{family-wise confidence level}) \geq 1 - \sum_{j=1}^k \alpha_j$$

or

$$(\text{family-wise confidence level}) \geq 1 - k\alpha$$

if $\alpha_j = \alpha$ for all j . Note that this inequality appears useless if $\sum_{j=1}^k \alpha_j \geq 1$. But, the idea is to adjust the individual (comparison-wise) levels α_j (perhaps all the same, α) so that the *family-wise* confidence level $1 - \sum_{j=1}^k \alpha_j$ is acceptably high. Generally speaking, we must make the α_j small. (Of course, if we want our family of intervals to simultaneously contain things with high confidence, then individual intervals must have higher confidence (smaller α_j). Duh.) But, this approach is not so efficient when k is large since making individual α_j small will, all else being equal, make your individual confidence intervals wide, perhaps unacceptably wide, perhaps to the point of being practically useless. So, while using the Bonferroni inequality is simple and very generally applicable to linear combinations, $\mathbf{C}\boldsymbol{\beta}$, we may want to consider other methods of ensuring high family-wise confidence levels, especially when the number k of family members is large.

The correspondence between intervals and tests gives the Bonferroni multiple comparison procedure for tests. Assuming that the null hypotheses of k tests are all true, the event that the corresponding family of intervals all simultaneously contain their respective true $\mathbf{C}\boldsymbol{\beta}$ values is the same as the event that corresponding tests all simultaneously do not falsely reject the null hypothesis—probability greater than or equal to $1 - k\alpha$ by Bonferroni, now assuming for simplicity that all individual α_j are the same value α . By fundamental rules of probability, the probability of falsely rejecting at least one (or more) tests in the family of tests is less than or equal to $1 - (1 - k\alpha) = k\alpha$. This is the concept of the *family-wise* error rate defined above. Thus, to ensure that your *family-wise* error rate is small, choose your individual rates to be small. (Again, duh.)

Take Home Message on Bonferroni Confidence Intervals

- If you want a *family-wise* confidence level of *at least* $1 - \alpha$, then set your *individual* interval levels to be $1 - \alpha/k$, where k is the number of members in your family of intervals, i.e., choose your individual error rate to be α/k .
 - $\mathbf{C}\hat{\boldsymbol{\beta}} \pm t(1 - \alpha/(2k), n - p)\hat{s}e(\mathbf{C}\hat{\boldsymbol{\beta}})$ 2-sided
 - $\mathbf{C}\hat{\boldsymbol{\beta}} - t(1 - \alpha/k, n - p)\hat{s}e(\mathbf{C}\hat{\boldsymbol{\beta}})$ 1-sided
 - $\mathbf{C}\hat{\boldsymbol{\beta}} + t(1 - \alpha/k, n - p)\hat{s}e(\mathbf{C}\hat{\boldsymbol{\beta}})$ 1-sided
- This is almost exactly the same as our regular t-based interval procedure except we now use α/k for each interval.
- Boo: More conservative (wider) intervals as number of intervals k increases.
- Yea: Applies to general linear combinations (general families) $\mathbf{C}\boldsymbol{\beta}$.

Take Home Message on Bonferroni Tests

- If you want a *family-wise* error rate at least as small as α , then set your *individual* rate to be α/k , where k is the number of members in your family of tests. That is, compare your typical p-value for testing about $\mathbf{C}\boldsymbol{\beta}$ to α/k instead of α . Equivalently, multiply our usual individual p-value by k and compare to the individual α (see `p.adjust` with the `method='bonferroni'` option).
- This is almost exactly the same as our regular t-based testing procedure except we now use α/k for each test.

- Boo: More conservative (harder to reject) tests as number of tests k increases.
- Yea: Applies to general linear combinations (general families) $\mathbf{C}\boldsymbol{\beta}$.
- Boo: Perhaps an example on a homework.

Tukey–Kramer Pairwise Comparison Procedure

- **Restricted Family.** The Tukey procedure ensures family-wise confidence levels for the family of all $\binom{a}{2}$ **pairwise differences of means** $\mu_i - \mu_{i'}, i \neq i'$, where a is the number of treatment levels (means) in the cell means model or number of levels of your factor in a single factor model. (Of course, Bonferroni is applicable, but we somehow hope that restricting ourselves to a particular family will lead to gains (reject more often (when we should) or be more precisely confident.)
- Reiterating, the family only includes members of the form $\mathbf{C}\boldsymbol{\beta} = \mu_i - \mu_{i'}$, $i \neq i'$. For our effects model, the form is $\mathbf{C}\boldsymbol{\beta} = \alpha_i - \alpha_{i'}$, $i \neq i'$ (either sum-to-zero or treatment constraints ($\alpha_1 = 0$)).
- **Ensures Nominal Family Coverage Rate.** Thus, the Tukey procedure ensures that all $\binom{a}{2}$ intervals for pairwise differences of means **simultaneously** successfully contain their respective true differences at a specified family-wise confidence level of $1 - \alpha$.
- **Correct Model.** The procedure assumes that our model is correct or at least approximately so.
- **Exact or Approximate.** The confidence level is exact if cell sample sizes, n_i , are equal and is approximately correct if sizes are not too different.
- **2-sided.** We'll only consider 2-sided intervals (and tests) for the Tukey procedure and omit further details.

- **Form of Confidence Interval.** In short, the form of Tukey–Kramer confidence intervals is

$$(\hat{\alpha}_i - \hat{\alpha}_{i'}) \pm \frac{q(a, n-a, 1-\alpha)}{\sqrt{2}} \widehat{se}(\hat{\alpha}_i - \hat{\alpha}_{i'}),$$

where $q(a, n-a, 1-\alpha)$ is the $1-\alpha$ quantile from the **Tukey Studentized range distribution** with parameters a (number of treatment means / single factor levels) and $n - a$ degrees of freedom.

- **NOTE Notation.** We use $q(a, n - a, 1 - \alpha)$ to denote a quantile, your author uses $q_{I,df}$ ([Far14, §15.4]), and the corresponding R function is (note order of arguments) `qtukey(1-alpha, a, n-a)`
- **R Functions.** Quantile function `qtukey(1-alpha, a, n-a)` and interval/test function `TukeyHSD` in R.
- So, once again we see the same form of confidence interval as we've been talking about, except our multiplier is different. Don't forget the $\sqrt{2}$! Again, it is 2-sided only.
- **Form of Test.** The form of the Tukey–Kramer test statistic is

$$qstat = \frac{((\hat{\alpha}_i - \hat{\alpha}_{i'}) - 0)\sqrt{2}}{\widehat{se}(\hat{\alpha}_i - \hat{\alpha}_{i'})}.$$

We can compare this to $q(a, n-a, 1-\alpha)$ or we can compute the p-value using ‘cdf’ function `ptukey(abs(qstat), a, n-a, lower.tail=FALSE)` and compare to our desired family-wise error rate α . Again, don't forget the $\sqrt{2}$!

Tukey Pairwise Comparison Example

```
> ## First, typical individual-wise interval for the difference of 2 means.
> ## (Just one difference for comparison: Diet B - Diet A). Note the use
> ## of the usual t multiplier without any (e.g. Bonferroni) adjustment. We
> ## use the latest lmod object (sum to zero constraint).
```

```

>
> ## By hand:
> summary(coagulation)

      coag      diet
Min.   :56.0   A:4
1st Qu.:61.8   B:6
Median :63.5   C:6
Mean    :64.0   D:8
3rd Qu.:67.0
Max.    :71.0

> bhat<- coef(lmod) ## using latest fit
> (alphaBmalphaA<- bhat[3] - bhat[2])

diet2
5

> sighat <- summary(lmod)$sigma
> (seBmA<- sighat * sqrt(1/6 + 1/4))

[1] 1.5275

> alphaBmalphaA + c(-1,1) * qt(0.975, 24-4) * seBmA

[1] 1.8136 8.1864

> ## Or, Cbeta approach (using estimable)
> Cmat<- matrix(c(0, -1, 1, 0),
+                  ncol=4, byrow=TRUE)
> d<- rep(0,nrow(Cmat))
> gmodels:::estimable(obj=lmod, cm=Cmat, beta0=d, conf.int=0.95)

      beta0 Estimate Std. Error t value DF Pr(>|t|) Lower.CI
(0 -1 1 0)      0        5     1.5275  3.2733 20 0.0038025   1.8136
      Upper.CI
(0 -1 1 0)     8.1864

> ## Now, Tukey intervals (wider to achieve family-wise confidence level).
> ## Note the use of qtukey (Tukey pairwise comparison distribution). Note
> ## the sqrt(2), too. (Diet B - Diet A)
> alphaBmalphaA + c(-1,1) * qtukey(0.95, 4, 24-4)/sqrt(2) * seBmA

[1] 0.72455 9.27545

```

```
> ## Tukey p-value for 2-sided test
> ptukey(abs(alphaB - alphaA), nmeans=4, df=24-4, lower.tail=FALSE)

  diet2
0.010288

> ## Tukey automatically (ah, that's nice). (all pairwise differences now)
> (tci <- TukeyHSD(aov(coag ~ diet, coagulation)))

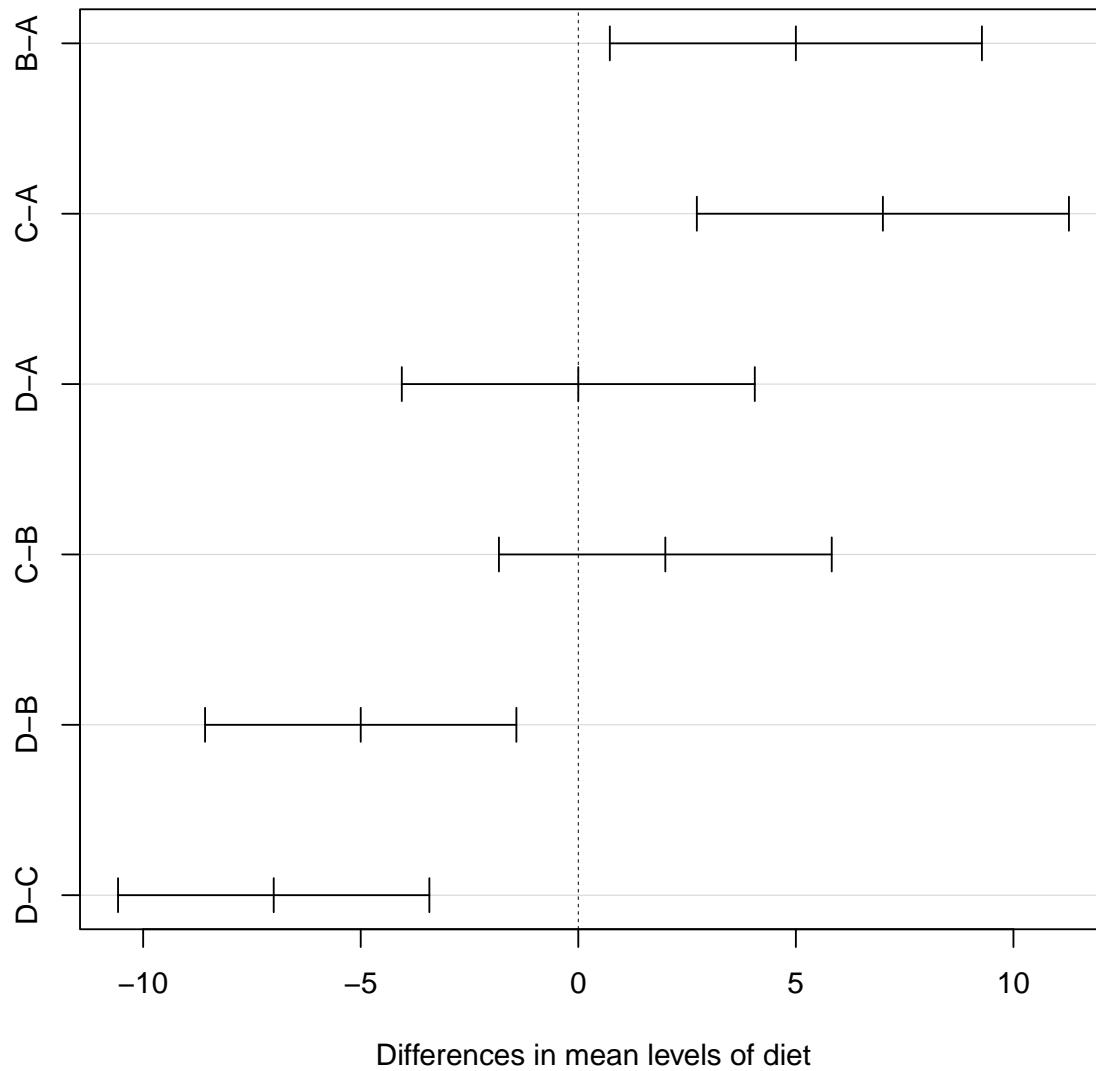
Tukey multiple comparisons of means
  95% family-wise confidence level

Fit: aov(formula = coag ~ diet, data = coagulation)

$diet
    diff      lwr      upr     p adj
B-A     5  0.72455  9.2754 0.01833
C-A     7  2.72455 11.2754 0.00096
D-A     0 -4.05604  4.0560 1.00000
C-B     2 -1.82407  5.8241 0.47660
D-B    -5 -8.57709 -1.4229 0.00441
D-C    -7 -10.57709 -3.4229 0.00013

> ## Tukey visually
> plot(tci)
```

95% family-wise confidence level



```
> ## Evidently, TukeyHSD adjusts for imbalance, which may explain
> ## the different p-values from ptukey and TukeyHSD.
```

In short, make your intervals longer to guarantee a family-wise coverage level, i.e., to guarantee that your family of 6 intervals simultaneously, at the same time, cover their respective difference of means targets at the nominal family-wise coverage rate (in the long run relative frequency sense as we've discussed before for individual intervals). In short, for p-values associated with

Scheffé's Procedure for Contrasts

- **Contrast.** A contrast $\mathbf{C}\boldsymbol{\beta}$ is a linear combination of means, μ_i (or of α_i in our effects model) where the coefficients on the μ_i (or on the α_i) sum to zero.
- **Scheffé CI.** The form of the Scheffé (2-sided) confidence interval for our one factor models is

$$\mathbf{C}\widehat{\boldsymbol{\beta}} \pm \sqrt{(a - 1)F(1 - \alpha, a, n_T - a)}\widehat{se}(\mathbf{C}\widehat{\boldsymbol{\beta}}),$$

where $\mathbf{C}\boldsymbol{\beta}$ must end up being a contrast of means, μ_i , (or α_i).

- **Scheffé Test** The form of the Scheffé (2-sided) test statistic for our one factor models is:

$$F_{stat} = \frac{(\mathbf{C}\widehat{\boldsymbol{\beta}} - 0)^2}{(a - 1)\widehat{se}(\mathbf{C}\widehat{\boldsymbol{\beta}})^2}$$

which we compare to $F(1 - \alpha, a - 1, n - a)$ or compute a p-value using `pf(Fstat, a, n-a, lower.tail=FALSE)` and compare this to our family-wise error rate α .

- Note the square-roots, the squares and the $(a - 1)!$
- **E.g.** 'Diet B - Diet A,' $\mathbf{C}\widehat{\boldsymbol{\beta}} = \widehat{\alpha_B} - \widehat{\alpha_A}$, but...
- Perhaps on a homework.

Pre-planned Comparisons and Data Snooping

- **Pre-Planned Comparisons.** Pre-planned comparisons refer to confidence intervals and/or tests conceived of **before inspecting your data**.

- **Always Assumed Pre-Planned For Us.** All of our *individual* confidence levels $(1 - \alpha)$ error rates (α) and p-values apply to individual, pre-planned intervals and tests, respectively (as long as model assumptions are met, at least approximately), though we did not discuss this much until now. (This is a big deal. How often are your inferences planned out before looking at your data? How often are inferences suggested after looking at data? How often do scientists report when inferences are pre-planned or not? How often do you think reported levels $(1 - \alpha)$, error rates (α) or p-values are actually valid (pre-planned)?)
- **Data Torture.** Inspecting your data and letting the data suggest to you the comparisons to make is called **data snooping** or **data dredging** or **data torture**. Most connotations of this are bad unless you reveal your snooping explicitly, and, moreover, remind your reader how it changes the nominal inferences (p-values, error rates and coverage rates); i.e., just because someone declares 95% confidence doesn't make it true, especially if it is cherry picked from among multiple confessions of the data under duress!
- **Does Your Procedure Allow Snooping?** Generally speaking, comparisons suggested by the data no longer have the nominal confidence levels (for intervals) or error rates (for tests), unless the procedure you are using allows for data snooping.
- **Snoop Among All Contrasts.** The Scheffé MC procedure applies to the family of *all*—that's right, all—contrasts of treatment (factor level) means.
- **Snoop Among All Pairwise Differences.** Tukey is restricted to particular contrasts—all pairwise comparisons of means.
- **Snoop Among Pre-Specified Linear Combinations.** Bonferroni lets us snoop among all forms of linear combinations of means or all forms of $C\beta$, contrasts or not, **but**, for Bonferroni, we need to **pre-specify** (before looking at our data) the particular linear combinations among which we would like to snoop.

Example 15.2 (What's "Wrong" with Torturing My Data?). Consider a series of 100 identical, independent experiments, each with a factor with $a = 5$ levels with $n_i = 1$ observation per level (one treatment replication)—okay, so this isn't very realistic, but it's simple and instructive. Furthermore, assume there is no difference in means among the levels and that the observations are normally distributed $N(\mu, \sigma^2)$. Then, by chance alone we would expect $\bar{Y}_1 - \bar{Y}_2$ (estimated difference of means, $\mu_1 - \mu_2$) to exceed (in absolute value) $1.96 * \sqrt{2\sigma^2}$ about 5 times out of the 100 experiments (what's 1.96? we assume we know σ^2 just for simplicity).

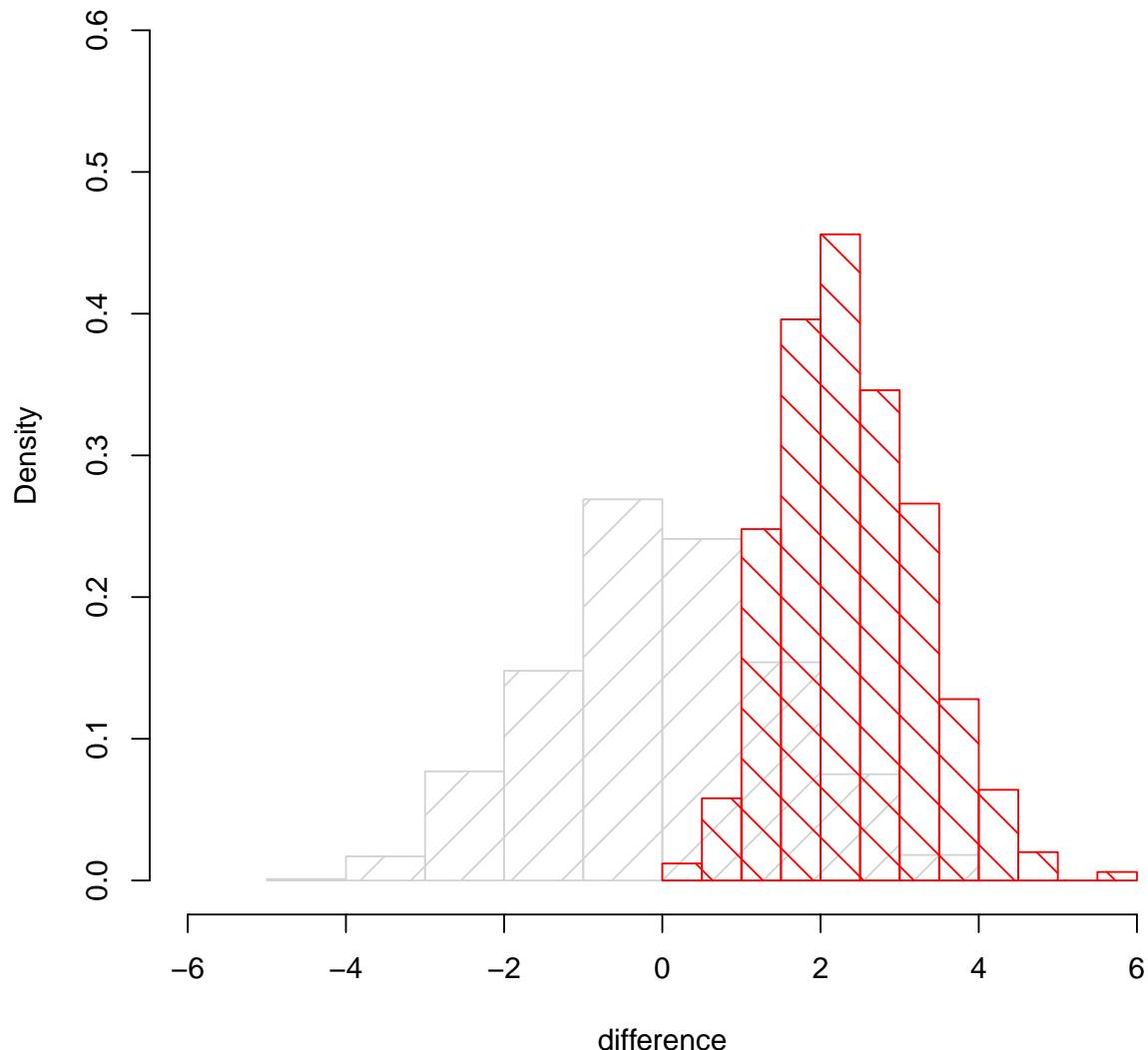
Now suppose that instead of computing $\bar{Y}_1 - \bar{Y}_2$ you compute $\max(\bar{Y}_i) - \min(\bar{Y}_i)$. That is, for each experiment, you wait to see the results—snoop dawg—and then compute the difference between the largest average and the smallest average. It should be clear that, even though there is no difference among means, the difference in extreme average values (max–min) is not zero on average (see simulation below), and this difference will tend to exceed $1.96 * \sqrt{2\sigma^2}$ many more times than 5 out of 100, i.e., you might falsely conclude a difference of means many more times than 5 out of 100. Thus, a nominal error rate of $\alpha = 0.05$ does not apply in this case where the data was used ("dredged" or "tortured") to suggest the comparison to make.

```
> ## Simulation of M=1000 such experiments. Assume sigma2 = 1 just
> ## for illustration.
> set.seed(8675309)
> M<- 1000
> ymat<- matrix(rnorm(5*M), nrow=M, byrow=TRUE)
>
> ## Pre-specified y2 - y1
> y2my1<- apply(ymat, 1, function(y) y[2] - y[1])
> sum(abs(y2my1) >= 1.96 * sqrt(2*1))
[1] 51
```

```
> ## Snoopin'
> ymaxmymin <- apply(ymat, 1, function(y) diff(range(y)))
> sum(ymaxmymin >= 1.96 * sqrt(2*1))

[1] 312

> ## May be thinking of this reference distribution (hist approx thereof)
> ## for your 0.05 error rate...
> hist(y2my1, density=5, xlim=c(-6, 6), ylim=c(0,0.6), prob=TRUE,
+       main=NULL, xlab="difference")
> ## ...but actual error rate comes from this reference distribution
> ## (hist approx thereof) and is much larger than 0.05. See?
> hist(ymaxmymin, density=5, angle=-45, border="red", col="red",
+       add=TRUE, prob=TRUE)
```



Remarks on Multiple Comparison Procedures

- **Snooping.** Again, Scheffé and Tukey allow data snooping, preplanned or not. Bonferroni does, too, but you must pre-specify—before looking

at the data—the family that you wish to snoop in.

- **Which MC procedure should you use?** First, you can only use a procedure on a member of the appropriate family. So, this might be the deciding factor. Bonferroni may always be applied. But, assuming you have a legitimate choice among 2 or 3 procedures, choose the one having the **smallest multiplier** in the expression of its confidence interval. So, if the multiplier for a Bonferroni confidence interval is 2.2, and the multiplier for a Scheffé multiplier is 3.9, choose Bonferroni to conduct your tests and/or construct your CIs. Etc.

15.5 False Discovery Rate

Another approach to adjusting p-values for multiple comparisons is to control the **proportion of effects identified (discovered) as significant but which are not real (are false)**. That is, control the **false discovery rate (FDR)**. In short, find the largest i for which

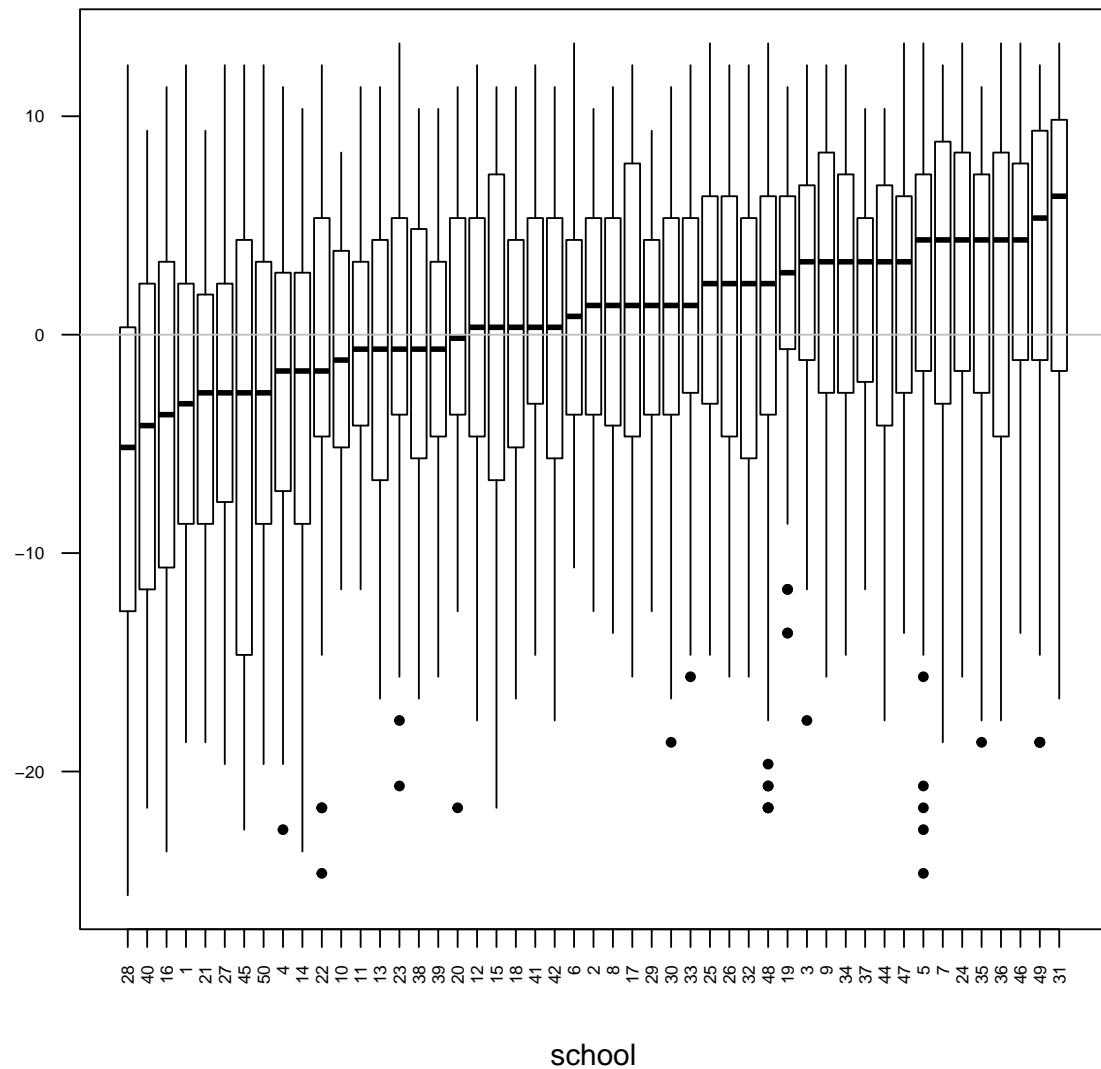
$$p_{(i)} \leq \alpha(i/m),$$

where $p_{(i)}$ is the i th ordered p-value, m is the number of comparisons (tests) and α is a comparison-wise error rate relative to which the p-values are adjusted; declare significance for all tests corresponding to $p_{(i)}$ up to and including this largest i . (Sorry, we switch to m instead of k . ??) We skip many of the details, but notice how FDR is less harsh than Bonferroni by adjusting (with case index i) what would otherwise be the fixed Bonferroni family error rate to allow for the natural range of p-values (small to big) that we would see even if all nulls are true; i.e., if we would “naturally” see range of p-values, it seems somehow unfair to compare a “naturally” occurring small p-value to the same error rate as a naturally occurring big p-value. Thus, we might expect FDR to be less conservative than Bonferroni. See [Far14, §15.5] for a bit more.

```
> ## Math test scores (response) from m=49 elementary schools (factor)
> data(jsp, package="faraway")
> summary(jsp[,c("math","school")])
```

```
      math          school
Min.   : 1.0    48   : 206
1st Qu.:22.0   33   : 131
Median  :28.0   42   : 131
Mean    :26.7   31   : 107
3rd Qu.:33.0   47   : 102
Max.   :40.0   50   : 101
                  (Other):2458

> ## Center math score so comparison to 0 makes sense.
> jsp$mathcent <- jsp$math - mean(jsp$math)
>
> ## Box plots (take that, ggplot2!):
> ## Reorder school factor levels by median math score for plotting
> jsp$schoolord<- with(jsp, reorder(school, math, median))
> tmp<- boxplot(mathcent ~ schoolord, data=jsp, plot=FALSE)
> bxp(tmp, whisklty=1, staplelty=0, outpch=20, las=2,
+       cex.axis=0.6, xlab="school")
> abline(h=0, col="grey")
```



```
> ## Cell means model to get school (centered) mean scores straightforward.
> lmod <- lm(mathcent ~ schoolord - 1, jsp)
> summary(lmod)
```

Call:
`lm(formula = mathcent ~ schoolord - 1, data = jsp)`

Residuals:

	Min	1Q	Median	3Q	Max
	-26.49	-4.90	1.10	5.54	18.17

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
schoolord28	-5.8286	1.0641	-5.48	4.6e-08 ***
schoolord40	-4.9855	1.2643	-3.94	8.2e-05 ***
schoolord16	-3.7374	1.0127	-3.69	0.00023 ***
schoolord1	-3.3685	0.7686	-4.38	1.2e-05 ***
schoolord21	-3.6185	0.7686	-4.71	2.6e-06 ***
schoolord27	-2.3058	0.8629	-2.67	0.00757 **
schoolord45	-4.6392	1.1114	-4.17	3.1e-05 ***
schoolord50	-2.6520	0.7336	-3.62	0.00030 ***
schoolord4	-2.6619	0.8688	-3.06	0.00220 **
schoolord14	-3.2898	1.1243	-2.93	0.00346 **
schoolord22	-1.7271	1.0870	-1.59	0.11217
schoolord10	-1.2453	1.5048	-0.83	0.40802
schoolord11	-0.4619	1.2461	-0.37	0.71089
schoolord13	-1.6474	0.8875	-1.86	0.06351 .
schoolord23	-0.8171	0.9680	-0.84	0.39868
schoolord38	-1.6106	1.1805	-1.36	0.17255
schoolord39	-1.4842	1.0990	-1.35	0.17696
schoolord20	-0.2095	1.1376	-0.18	0.85386
schoolord12	0.2082	0.8401	0.25	0.80429
schoolord15	-0.4921	1.0127	-0.49	0.62702
schoolord18	-0.8619	0.9941	-0.87	0.38597
schoolord41	0.8809	1.2461	0.71	0.47966
schoolord42	-0.3261	0.6441	-0.51	0.61275
schoolord6	0.2381	0.9517	0.25	0.80249
schoolord2	0.6714	1.2287	0.55	0.58481
schoolord8	0.5909	0.7904	0.75	0.45472
schoolord17	1.0881	1.6485	0.66	0.50927
schoolord29	0.2090	0.9363	0.22	0.82334
schoolord30	0.2392	0.7728	0.31	0.75698
schoolord33	1.0633	0.6441	1.65	0.09889 .
schoolord25	1.4232	1.0753	1.32	0.18578
schoolord26	0.4919	0.8347	0.59	0.55570
schoolord32	-0.0268	0.8570	-0.03	0.97506
schoolord48	0.2119	0.5136	0.41	0.68003
schoolord19	1.6108	1.1114	1.45	0.14734
schoolord3	2.2964	1.0641	2.16	0.03099 *
schoolord9	2.2458	0.9144	2.46	0.01410 *

```

schoolord34    1.9359    0.7686    2.52   0.01183 *
schoolord37    1.7618    0.9598    1.84   0.06651 .
schoolord44    0.5881    1.8431    0.32   0.74969
schoolord47    2.3969    0.7300    3.28   0.00104 **
schoolord5     1.8320    0.8092    2.26   0.02364 *
schoolord7     1.7227    1.1805    1.46   0.14459
schoolord24    2.6238    0.9288    2.82   0.00476 **
schoolord35    1.9041    1.0127    1.88   0.06016 .
schoolord36    2.3605    0.7815    3.02   0.00254 **
schoolord46    3.0636    1.0323    2.97   0.00302 **
schoolord49    2.7964    0.8688    3.22   0.00130 **
schoolord31    3.8241    0.7127    5.37   8.6e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.37 on 3187 degrees of freedom
Multiple R-squared:  0.0821, Adjusted R-squared:  0.0679
F-statistic: 5.81 on 49 and 3187 DF,  p-value: <2e-16

> ## Not cell means model (sequential) anova table
> ## (gives correct overall F and R^2):
> anova(lm(mathcent ~ schoolord, jsp))

Analysis of Variance Table

Response: mathcent
          Df Sum Sq Mean Sq F value Pr(>F)
schoolord   48 15484     323     5.94 <2e-16 ***
Residuals 3187 173212      54
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ## Unadjusted (individual- or comparison-wise) p-values
> ## (from cell means model):
> pvals <- summary(lmod)$coef[,4]
>
> ## Bonferroni adjusted p-values. Evidently, those p-values that,
> ## when multiplied by the number of comparisons (m=49 here), are greater
> ## than 1, are set to 1. You might call these family-wise p-values.
> padj <- p.adjust(pvals, method="bonferroni")
>
> ## Manual Bonf. adjusted p-value for comparison
> mypadj<- ifelse(pvals*49 >= 1, 1, pvals*49)
> all(mypadj == padj) ## just curious if equal

```

```
[1] TRUE

> ## Compare adjusted p to 0.05 (or compare unadjusted p to 0.05/49):
> ## These estimated (centered) math score means
> ## are formally significantly different from 0 (after
> ## B adjustment). That is, the family-wise Type I error is that at
> ## least one math score mean, out of 49, is not equal to zero, and
> ## the family-wise error rate (FWER) is guaranteed by Bonferroni
> ## to be at least as small as 0.05, i.e.,
> ## Pr(family-wise Type I) <= 0.05. That is, the probability
> ## of falsely rejecting one or more tests of  $H_0: \mu_i = 0$  is no greater
> ## than 0.05.
> coef(lmod)[padj < 0.05] ## Bonf conservative 8 signif diffs

schoolord28 schoolord40 schoolord16 schoolord1 schoolord21
-5.8286     -4.9855     -3.7374     -3.3685     -3.6185
schoolord45 schoolord50 schoolord31
-4.6392     -2.6520      3.8241

> ## Benjamini and Hochberg control the false discovery rate (FDR)
> ## by sorting p-values (denoted  $p(i)$ ) and declare significance for
> ## all tests corresponding to  $p(i) \leq \alpha * i / m$  up to and
> ## including this largest  $i$  if for which this inequality holds,
> ## where  $m=49$  comparisons here. A more liberal 18 diffs compared to
> ## Bonferroni.
> ## "By hand":
> maxi<- max(which(sort(pvals) < (1:49)*0.05/49))
> coef(lmod)[names(sort(pvals))[1:maxi]]]

schoolord28 schoolord31 schoolord21 schoolord1 schoolord45
-5.8286      3.8241     -3.6185     -3.3685     -4.6392
schoolord40 schoolord16 schoolord50 schoolord47 schoolord49
-4.9855     -3.7374     -2.6520      2.3969      2.7964
schoolord4 schoolord36 schoolord46 schoolord14 schoolord24
-2.6619      2.3605      3.0636     -3.2898      2.6238
schoolord27 schoolord34 schoolord9
-2.3058      1.9359      2.2458

> ## More automatically (?):
> padj <- p.adjust(pvals, method="fdr")
> coef(lmod)[padj < 0.05] ## same schools (different order than above)
```

```
schoolord28 schoolord40 schoolord16 schoolord1 schoolord21  
-5.8286 -4.9855 -3.7374 -3.3685 -3.6185  
schoolord27 schoolord45 schoolord50 schoolord4 schoolord14  
-2.3058 -4.6392 -2.6520 -2.6619 -3.2898  
schoolord9 schoolord34 schoolord47 schoolord24 schoolord36  
2.2458 1.9359 2.3969 2.6238 2.3605  
schoolord46 schoolord49 schoolord31  
3.0636 2.7964 3.8241
```

> ## The Bonferroni FWER is more stringent than FDR.

Lecture 16

Models with Several Factors

Contents

16.1 Initial Concepts and Notation	449
16.2 Example	451
16.3 Cell Means Model of $E(Y_{ijk} \mathbf{x})$	456
16.4 Effects Model: Before Constraints	458
16.5 Effects Model: Sum-to-Zero Constraints/Coding	460
16.5.1 Effects Model: STZ Coding: Parameter Interpretation	462
16.5.2 (Non-)Additive Model & Saturation	464
16.5.3 ANOVA Model Components: Means and Effects	465
16.5.4 Effects Model: STZ: Example: Initial Analysis	472
16.5.5 Effects Model: STZ: Example: Diagnostics	476
16.5.6 Effects Model: STZ: Example: ANOVA For Common $\mathbf{C}\beta$	486
16.5.7 Pit Stop: What is ANOVA?	487
16.5.8 Effects Model: STZ: Example: F v R & $\mathbf{C}\beta$ Approach	490
16.5.9 Effects Model: STZ: Example: Summary So Far	492
16.6 Effects Model: Treatment Constraints/Coding	493
16.6.1 Effects Model: Trmt Coding: Parameter Interpretation	494
16.6.2 (Non-)Additive Model & Saturation	496
16.6.3 Effects Model: Trmt: Example: Initial Analysis	498
16.6.4 Effects Model: Trmt: Example: ANOVA For Common $\mathbf{C}\beta$	502
16.6.5 Effects Model: Trmt: Example: F v R & $\mathbf{C}\beta$ Approach	503
16.6.6 Effects Model: Trmt: Example: Summary	505
16.7 SS Type, Balance & the Marginality Principle	505
16.7.1 Sequential (Type I) SS ANOVA	505

16.7.2 Partial (Type III) SS ANOVA	510
16.7.3 Marginality Principle (aka Hierarchy Principle)	512
16.7.4 Summary & Remarks	516
16.8 Additive Model: Tests for Overall Main Effects	516
16.8.1 F v R Approach	517
16.8.2 C β Approach	519
16.9 Additive Model: More Detailed Inference of Main Effects	521
16.10 Final Remarks	531

Main Objectives:

- Overall F tests and F tests for main/interaction effects using ANOVA tables or using F v R (extra SS) approach or using C β approach
- Inference for (more detailed) linear combinations of β
- Effects model with sum-to-zero constraints or treatment constraints (While these notes offer a detailed treatment of both treatment and sum-to-zero constraints, we will only cover sum-to-zero in class for time. We discussed both treatment constraints and sum-to-zero constraints in the previous chapter.)
- Types of SS in ANOVA tables (sequential aka type I; partial aka type III) (drop1; car package's Anova function)
- Marginality principle aka hierarchy principle
- Balance

 \mathcal{O}

Reading:

- [Far14, Chap. 16]
- [RS13, Chap. 13]
- [KNNL05, Chap. 19, 23 & 24]
- Topics that we do not cover directly but for which the current material is very closely related (FYI only): no treatment replication ([RS13, Chap. 14], [KNNL05, Chap. 20]); RCBD ([KNNL05, Chap. 21]); ANCOVA ([KNNL05, Chap. 22])
- We give more detailed references in the sections below

 \mathcal{R}

16.1 Initial Concepts and Notation

I no longer attempt to align section numbering to any text, though I continue to give references. Still, this note chapter is closely associated with [Far14, Chap. 16], which I suggest you read.

- **Extending One Factor Models.** We repeat and adapt some terminology presented previously in the single factor case (one-way ANOVA). We illustrate mostly with 2 factors. Extension to 3 or more should be fairly obvious. Homework?
- **Number of factor variables:** 2 (or more): Generically factor A, factor B, etc.
- **Number of factor levels** a for factor A, b for factor B, etc.

- **Treatments.** As before, treatments are the set of conditions defined by the **unique combinations of factors' levels** across all factors. **Now that we have two (or more) factors, treatments are no longer synonymous with factor levels.**
- **Sample sizes** (or number of units per treatment level) are denoted $[n_{ij}]$, $i = 1, \dots, a, j = 1, \dots, b$.
 - We can sum over j to get the number of units, $n_{i\cdot}$, for factor A, level i ,
 - or sum over i to get the number of units, $n_{\cdot j}$, for factor B, level j .
 - Notation for, say, a three-way layout?
- **Observation** $[Y_{ijk}]$ is the k th observation (response) at the treatment level defined by the i th level of factor A and the j th level of factor B, $i = 1, \dots, a, j = 1, \dots, b, k = 1, \dots, n_{ij}$. Notation for, say, a three-way layout?
- **Total number of observations**, i.e., total sample size, is denoted as $[n_T]$, i.e., $n_T = n_{11} + n_{12} + \dots + n_{ab} = \sum_{i=1}^a \sum_{j=1}^b n_{ij}$. Three-way layout?
- **Balance.** If we have an **equal number of observations per treatment level**, then we say our treatment design is balanced. Otherwise it is unbalanced. i.e., $n_{11} = n_{12} = \dots = n_{ab} = n$, where n is the common number of observations in each treatment. Thus, in the balanced case, $n_T = nab$. Three-way layout?
- **Treatment Replicates.** If we have $n_{ij} = n$, $i = 1, \dots, a, j = 1, \dots, b$, then we say treatments have been (fully) replicated n times.
- **Observational or experimental treatment (cell) level means** are denoted by $[\mu_{ij}]$, $i = 1, \dots, a, j = 1, \dots, b$. Three-way layout?

- Factor level (marginal) means

$$\mu_{i\cdot} = \sum_{j=1}^b \mu_{ij}/b$$

$$\mu_{\cdot j} = \sum_{i=1}^a \mu_{ij}/a$$

- Overall mean

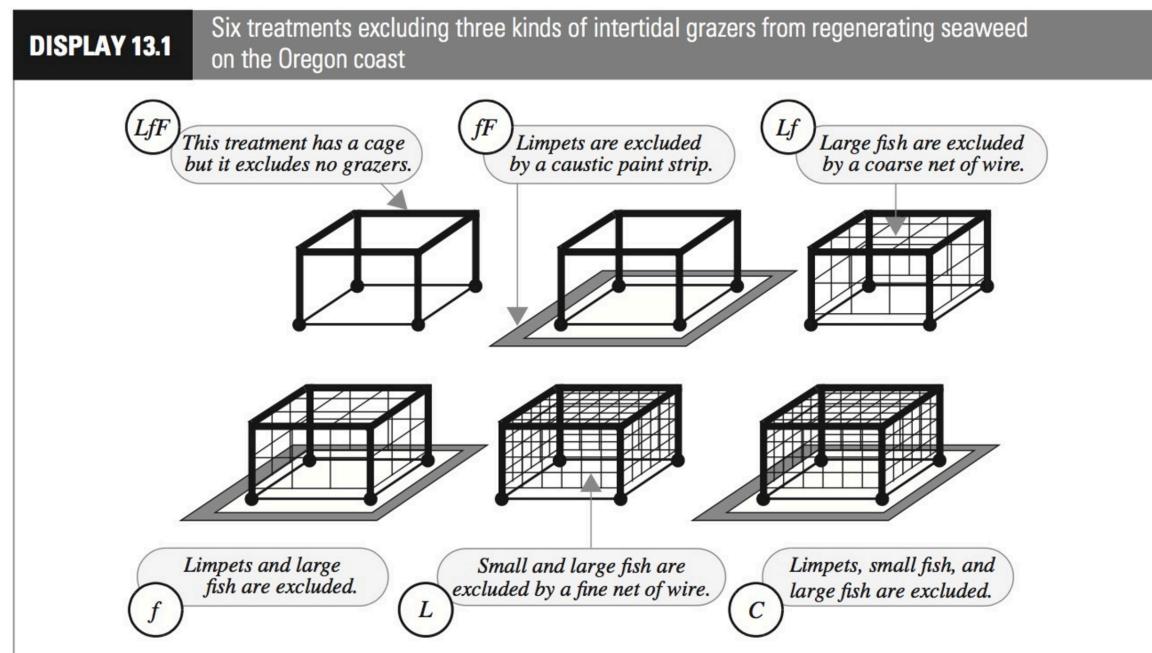
$$\begin{aligned}\mu_{..} &= \sum_{i=1}^a \sum_{j=1}^b \mu_{ij}/(ab) \\ &= \sum_{j=1}^b \mu_{\cdot j}/b \\ &= \sum_{i=1}^a \mu_{i\cdot}/a\end{aligned}$$

16.2 Example

Example 16.1 (Intertidal Seaweed Grazers). (*not used in a pejorative sense, of course*) Researchers designed an experiment to investigate the impacts of grazing on the regeneration rates of seaweed. See [RS13, Sec. 13.1.1] for detailed discussion. The basic treatment design (more in class) is shown in the next figure ([RS13, Display 13.1]).

The intertidal zone is a highly variable environment, which may affect regeneration, but this is (presumably) well known among such researchers, and is not of primary interest. Still, the researchers wish to account for environmental effects in an effort to help elucidate the effects of grazing, their primary interest. The researchers **replicate** their basic treatment design at eight locations, which we call **blocks**, with, in this case, each block (location)

itself containing two replications of the basic “treatment” design. The second figure below illustrates the configuration of the observations by “treatments” and blocks ([RS13, Display 13.2]). Note that it is the combination of the “treatments” factor and the blocks factor that defines what we called “treatments,” i.e., [RS13, Sec. 13.1.1] use the term “treatments” differently than we’ve defined it. More in class.



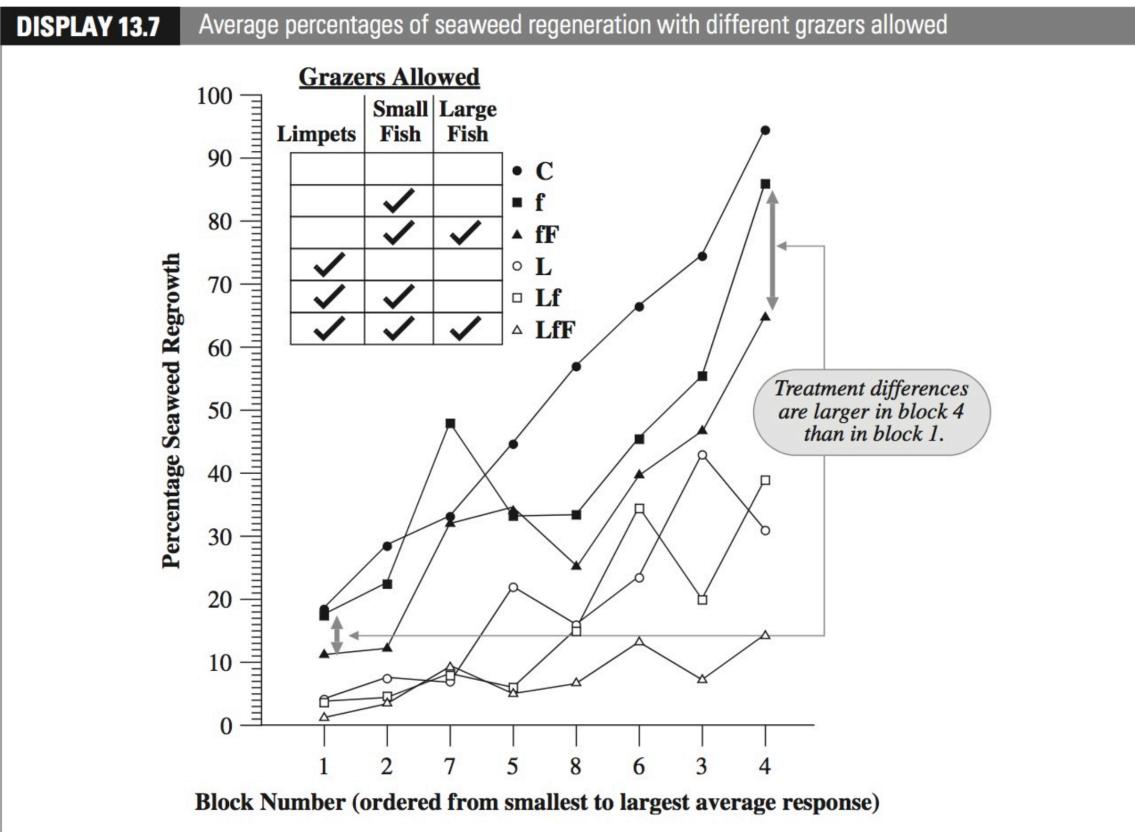
DISPLAY 13.2

Percentage of regenerated seaweed cover on plots with different grazers excluded, in eight blocks of differing tidal situation and exposure

Block #	Treatment: Grazers with access											
	Control		L		f		Lf		fF		LfF	
1	14	23	4	4	11	24	3	5	10	13	1	2
2	22	35	7	8	14	31	3	6	10	15	3	5
3	67	82	28	58	52	59	9	31	44	50	6	9
4	94	95	27	35	83	89	21	57	57	73	7	22
5	34	53	11	33	33	34	5	9	26	42	5	6
6	58	75	16	31	39	52	26	43	38	42	10	17
7	19	47	6	8	43	53	4	12	29	36	5	14
8	53	61	15	17	30	37	12	18	11	40	5	7

Let's reconcile this example with our concepts/definitions for two-way ANOVA before proceeding.

- **Factors?** What are the factors? Factor levels? a ? b ?
- **Treatments?** What are the treatments? (Careful. The term “treatment” in the problem statement is different than we defined it.)
- **Responses?** Y_{ijk} ?
- **Balanced?** n_{ij} ?
- What would an interaction plot look like? See [RS13, Display 13.7], reproduced nearby.



- **Factorial Treatment Design (not):** The example conjures three factors (not including blocks), each at two levels, defining $2^3 = 8$ “treatment” levels—a **three-way ANOVA** or, more particularly a **2^3 factorial design** (3 factors, each at 2 levels) of treatments. Again, not including blocks. See [RS13, Chap. 24] and [KNNL05, Chaps. 15 & 29]. But, consider the two levels defined by the exclusion of small fish and the inclusion of large fish, including or excluding limpets...are you thinking?...Thus, we consider here only $b = 6$ levels for the “treatment” factor, and do not refer to a “factorial” structure for “treatments.” We just have a single “treatment” factor (in addition to the block/location factor). Again, don’t confuse “treatments”!
- **Randomized Complete Block Design (RCBD):** Our example fits the definition of what is called a randomized complete block design (RCBD).

Blocks or blocking is a relatively special experimental design topic; see [RS13, Chap. 24] and [KNNL05, Chaps. 21 & 28] and [Far14, Chap. 17].)

- **Matching or Covariate Adjustment.** Blocking is a generalization of the notion of **pairing** or **match pairs** procedures that you may have seen in an introductory course; “pair” refers to two levels of “treatments” within each level of block, i.e., two treatment levels are somehow matched by each level of block. The rationale for blocking is closely related to that of matching and covariate adjustment discussed in [Far14, §5.5 & 5.6].
- **Simply a Two-Way ANOVA:** For this seaweed grazing example, we simply consider a two-way layout defined by two factors. We do not give much consideration to the special nature of RCBD, factorial designs, or other particular notions of experimental designs.

The next chunk gets the data from the Sleuth3 package and displays them briefly. Cover is the response variable: percent cover of regenerated seaweed in a plot. Note that R already views the Block and Treat variables as factors. We'll continue with these data in subsequent sections.

```
> case1301.df<- Sleuth3::case1301
> ##
> ## How is the data ``structured?:
> str(case1301.df)

'data.frame': 96 obs. of  3 variables:
 $ Cover: int  14 23 22 35 67 82 94 95 34 53 ...
 $ Block: Factor w/ 8 levels "B1","B2","B3",...: 1 1 2 2 3 3 4 4 5 5 ...
 $ Treat: Factor w/ 6 levels "C","L","Lf","LfF",...: 1 1 1 1 1 1 1 1 1 1 ...

> ## First few obs:
> head(case1301.df)
```

```
Cover Block Treat
1     14    B1     C
2     23    B1     C
3     22    B2     C
4     35    B2     C
5     67    B3     C
6     82    B3     C
```

```
> ## Last few:
> tail(case1301.df)
```

```
Cover Block Treat
91    10    B6   LfF
92    17    B6   LfF
93     5    B7   LfF
94    14    B7   LfF
95     5    B8   LfF
96     7    B8   LfF
```

16.3 Cell Means Model of $\mathbf{E}(Y_{ijk} | \mathbf{x})$

- **Like the Single Factor Case.** The cell means model for two-way (or higher-way) ANOVA is essentially the same as in one-way AVOVA, up to an obvious and slight change of notation for, now, multiple factors defining a cell (i.e., treatment). See [KNNL05, Sec. 19.3]. Again, it's about as simple as things get.

$$\mathbf{E}(Y_{ijk} | Bx) = \mu_{ij}$$

- **Etc.** More completely,

$$Y_{ijk} \stackrel{\text{ind}}{\sim} N(\mu_{ij}, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, b \quad k = 1, \dots, n_{ij}$$

or, equivalently (from basic results in INF 511 notes Appendix C)

$$Y_{ijk} = \mu_{ij} + \epsilon_{ijk} \quad \epsilon_{ijk} \stackrel{\text{iid}}{\sim} N(0, \sigma^2) \quad i = 1, \dots, a, \quad j = 1, \dots, b \quad k = 1, \dots, n_{ij}$$

where, as before, ϵ_{ijk} is the **error** of observation k in the treatment level defined by level i of factor A and level j of factor B, and its (error) variance (component), σ^2 , is assumed to be common to all treatment levels.

- **Linear Model.** Again, we can write this cell means model in the form of a general linear model using matrix notation.

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_{n_T}).$$

- **What is X?** As in the one-way (one factor), case, we should be able to write down an example of the \mathbf{X} matrix.
- **Not First Choice.** The cell means model typically is not the first choice for analyzing data in a multi-way layout. This is different than for the one-way layout, where the cell means model is often a very reasonable first choice, though a factor effects model is common, too, as we saw (with either the treatment coding or sum-to-zero coding being common in the one-way case).
- **Usually Effects Model.** For two- and higher-way layouts, a factor effects model is typically employed.
- **Messy Data.** Still, when first choices somehow fail, in some sense, we may return to the cell means model for further analysis. We do not analyze the data now in terms of the cell means model. We could, but we have limited time, we've covered it before, and use of the cell means model here, as we said, is not typical except, perhaps, in "messy data" situations. (Note, we may ignore factors and define means μ_i , $i = 1, \dots, ab$, in which case our one-way cell means model applies directly.)

16.4 Effects Model: Before Constraints

-

$$Y_{ijk} = \mu_{..} + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk} \quad \epsilon_{ijk} \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$$

$$i = 1, \dots, a, \quad j = 1, \dots, b, \quad k = 1, \dots, n_{ij}.$$

where

- **Overall Constant Effect**

$$\mu_{..}$$

is an overall constant effect common to all observations in all treatments; it will be an overall mean, as discussed above, when we impose sum-to-zero constraints (as the .. subscript anticipates), but it will not be an overall mean in the case of treatment constraints, similar to the one-way case;

- **Factor A (Main) Effects**

$$\alpha_i$$

is an effect common to all observations in the i th level of factor A;

- **Factor B (Main) Effects**

$$\beta_j$$

is an effect common to all observations in the j th level of factor B; and

- **Interaction (Not Main) Effects**

$$(\alpha\beta)_{ij}$$

is an effect of factor A level i within factor B level j (or vice-versa), over and above the α_i and β_j main effects.

- **Overparameterized/Non-Identifiable/Redundancy.** As in the one-way layout, without constraints, we have redundancy, i.e., our mean model is currently **overparameterized**. And, again, another way to say this is

that our mean model parameters are **not identifiable** or **not estimable**. Similar to the one-way factor effects parameterization (Section 15.1.3), we can illustrate this non-identifiability by adding zero to our effects model. For example,

$$\begin{aligned} E(Y_{ijk} | \mathbf{x}) &= \mu_{..} + \alpha_i + \beta_j + (\alpha\beta)_{ij} \\ &= (\mu_{..} + 5) + (\alpha_i - 3) + (\beta_j - 1) + ((\alpha\beta)_{ij} - 1) \\ &= \mu_{..}^* + \alpha_i^* + \beta_j^* + (\alpha\beta)_{ij}^* \end{aligned}$$

(Notice we have $p = 1 + a + b + ab$ parameters for only ab means.)

- **'Splain it To Me, Lucy.** That is, we get the same mean value, $E(Y | \mathbf{x})$, (hence same normal distribution) despite different parameter values. In other words, even if we knew this mean model to be true (or knew the entire distribution (normal) with this mean (and variance σ^2)), the mean model cannot help us to identify uniquely a single parameter β , so how can we expect to estimate a unique β when we just have data? (What's β ?) In other words, our model cannot help us to identify particular values for the parameters $\mu_{..}$, the α_i , the β_j or the $(\alpha\beta)_{ij}$, i.e., without a constraint on the parameters, the interpretation of the individual parameters is arbitrary. (Note that their sum $\mu_{..}^* + \alpha_i^* + \beta_j^* + (\alpha\beta)_{ij}^*$ is identified: it's the ij th cell mean, of course, which is the same value above, stars or not.)
- **Redundancy.** Yet another way to see this redundancy is by considering the (non-identifiable) parameter vector

$$\boldsymbol{\beta} = (\mu_{..}, \alpha_1, \dots, \alpha_a, \beta_1, \dots, \beta_b, (\alpha\beta)_{11}, \dots, (\alpha\beta)_{ab})^T$$

and its associated \mathbf{X} matrix, constructed with a column of 1's for the constant, $\mu_{..}$, a (number of factor A levels) columns of 0/1's (indicator/dummy variable columns), one for each α_i , b (number of factor B levels) columns of 0/1's, one for each β_i , and ab columns of 0/1's, one for each $(\alpha\beta)_{ij}$ (the interaction columns are just the element-wise product of the A and B indicator columns). (in class?)

16.5 Effects Model: Sum-to-Zero Constraints/Coding

- One way to resolve the redundancy issue is by constraining sets of parameters to sum to zero, analogous to our discussion of the factor effects model with sum-to-zero constraints for one-way ANOVA in §15.1.3 and 15.1.6:
- Refer to [RS13, 13.5.6] for a very brief discussion on the (additive; no interactions) factor effects model (parameterization) and to [KNNL05, Sec. 19.3] for a two-way presentation and [KNNL05, Sec. 24.1] for a three-way presentation; these presentations discuss the factor effects model with the sum-to-zero constraints/coding, not treatment constraints/coding, which we will discuss, later.

- **Constraints.**

$$\begin{aligned} \sum_{i=1}^a \alpha_i &= 0 \\ \sum_{j=1}^b \beta_j &= 0 \\ \sum_{i=1}^a (\alpha\beta)_{ij} &= 0 \quad j=1, \dots, b, \\ \sum_{j=1}^b (\alpha\beta)_{ij} &= 0 \quad i=1, \dots, a. \end{aligned}$$

- **Can Solve for Some Effects.** These suggest, e.g.,

$$\begin{aligned}\alpha_a &= - \sum_{i=1}^{a-1} \alpha_i, \\ \beta_b &= - \sum_{j=1}^{b-1} \beta_j, \\ (\alpha\beta)_{aj} &= - \sum_{i=1}^{a-1} (\alpha\beta)_{ij}, \quad j=1, \dots, b, \\ (\alpha\beta)_{ib} &= - \sum_{j=1}^{b-1} (\alpha\beta)_{ij}, \quad i=1, \dots, a.\end{aligned}$$

- **Coding.** And, this tells us how to (re-)code the (non-redundant) \mathbf{X} matrix.
- **Now, ab Means, ab Parameters.** In other words, similar to the one-way case, we do not need to include α_a , β_b ($\alpha\beta)_{aj}$ (all j) or ($\alpha\beta)_{ib}$ (all i) in our model because we can solve for them. Thus, we only need $p = 1 + (a-1) + (b-1) + (a-1)(b-1) = (1 + (a-1))(1 + (b-1)) = ab$ parameters for ab means, μ_{ij} (FOIL).

- What does the non-redundant β look like? (more in class)
- What does the non-redundant \mathbf{X} look like? (more in class) (All interaction term (re-coded) columns can be obtained by multiplying (element-wise!) each (re-coded) column associated with one factor by each (re-coded) column associated with another factor.)
- Again, we are simply performing regression with specially coded “ \mathbf{X} ” variables, one for each column in \mathbf{X} , and with associated coefficients having a particular interpretations.

- We'll see how to implement these constraints/coding in R, shortly.
- Higher way effect model? (briefly in class?)
- Non-additive model? (briefly in class?)

16.5.1 Effects Model: STZ Coding: Parameter Interpretation

- Of course, we can relate cell means model and effects models parameters as

$$\mu_{ij} = \mu_{..} + \alpha_i + \beta_j + (\alpha\beta)_{ij}.$$

Indeed, the (non-additive or saturated...more shortly) effects model is equivalent to the cell means model: their respective \mathbf{X} matrices span the same covariate/regressor space, each gives the same predicted values, same residuals, same estimate of σ^2 , same overall F-test; this is similar to the one-way case. Perhaps more on this later or in an assignment.

- **Overall Effect.** Averaging μ_{ij} over i and j, **with** sum-to-zero constraints, gives

$$\mu_{..} = \sum_i \sum_j \mu_{ij} / (ab),$$

so that $\mu_{..}$ now is not merely some arbitrary overall constant effect, but is, in particular, the **overall (unweighted) mean** (average) of the μ_{ij} , similar to the one-way case with sum-to-zero constraint discussed in a previous chapter.

- **Main Effects of Factor A.** With sum-to-zero constraints, averaging μ_{ij} over j gives,

$$\mu_{i.} \equiv \mu_{..} + \alpha_i$$

or

$$\alpha_i = \mu_{i.} - \mu_{..},$$

so that α_i is interpreted as the deviation from the overall mean due to being in the ith level of factor A. It is the **main effect** of the ith level

of A in the sense that it is added to all observations in the ith level of A without regard for the observation's factor B level. We have already discussed (Section 16.5.3) that inferring main effects (lower order terms) in the presence of interactions (see bullet below) (higher order terms) may not be sensible (see Section 16.5.3)

- **Main Effects of Factor B.** With sum-to-zero constraints, averaging over i gives,

$$\mu_{\cdot j} \equiv \mu_{..} + \beta_j$$

or

$$\beta_j = \mu_{\cdot j} - \mu_{..},$$

so that β_j is interpreted as the deviation from the overall mean due to being in the jth level of factor B. It is the **main effect** of the jth level of B in the sense that it is added to all observations in the jth level of B without regard for the observation's factor A level. We have already discussed (Section 16.5.3) that inferring main effects (lower order terms) in the presence of interactions (see bullet below) (higher order terms) may not be sensible (see Section 16.5.3)

- **Interaction Effects.**

$$\begin{aligned} (\alpha\beta)_{ij} &\equiv \mu_{ij} - (\mu_{..} + \alpha_i + \beta_j) \\ &= \mu_{ij} - \mu_{..} + (\mu_i - \mu_{..}) + (\mu_{\cdot j} - \mu_{..}) \\ &= \mu_{ij} - \mu_i - \mu_{\cdot j} + \mu_{..}, \end{aligned}$$

After accounting for an overall effect ($\mu_{..}$) and the main (or global or average factor) effects (α_i , and β_j), $(\alpha\beta)_{ij}$ is the additional effect required to reproduce the mean, μ_{ij} . If we hold, e.g., j constant, we see that we may interpret $(\alpha\beta)_{ij}$ as an additional effect of the ith level of A, due to being at level j of factor B. (Or, If we hold, e.g., i constant, we see that we may interpret $(\alpha\beta)_{ij}$ as an additional effect of the jth level of B, due to being at level i of factor A.) In other words, the effects of factor A (B) depends on the level of factor B (A), and we say that the factors **interact** and call $(\alpha\beta)_{ij}$ an **interaction effect**. Again, we've

discussed interaction (Section 16.5.3), and we have warned about the potential nonsensicality of inferring main effects (lower order terms) in the presence of certain interaction effects (higher order terms).

16.5.2 (Non-)Additive Model & Saturation

- **Non-Additive Model.** The effects model presented here, with (non-zero) interaction effects, $(\alpha\beta)_{ij}$, is often referred to as a non-additive model because we cannot reproduce the treatment means, μ_{ij} , by *adding* main effects (and overall effect) alone; we also need to add $(\alpha\beta)_{ij}$ to reproduce treatment means. (How'd we get to non-additivity by one more addition?...anyway...)
- **Saturated Model.** We also say that the model is saturated in the sense that we cannot specify further (linear) mean model complexity/flexibility; we have reached the “saturation point” of (linear) mean model complexity by allowing each treatment level to have its own value—any value whatsoever. (The non-additive model, which, as we said, is equivalent to the cell means model, is saturated. No more parameters, please.)
- **Over-Saturation.** We might even consider the notion of model over-saturation: the case of having too much complexity in the sense of not being able to identify the model parameter values, as illustrated by the effects model (with interaction) without constraints, discussed above.
- **Additive Model.** We might simplify our model (often supported by an F-test) by omitting, e.g., the interactions, to arrive at a simpler, additive model,

$$Y_{ijk} = \mu_{..} + \alpha_i + \beta_j + \epsilon_{ijk} \quad \epsilon_{ijk} \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$$

$$i = 1, \dots, a, \quad j = 1, \dots, b, \quad k = 1, \dots, n_{ij}.$$

(All quantities are as described previously in the non-additive/saturated model, constrained or not.)

- **Simpler.** As we have discussed (Section 16.5.3), without interactions, we are left with a relatively simple interpretation in terms of main effects.
- **Strategy for Analysis.** We often begin (if data permit) with a non-additive model, then test to see if we can reasonably omit interaction effects to simplify further analysis/interpretation. This process is discussed in [RS13, Sec. 13.5.1] and [KNNL05, Sec. 19.7]. See, in particular, [KNNL05, Fig. 19.11 pg. 848]. In some sense, this process is no more a matter of performing **F v R F-tests**, as we will continue to illustrate below. Thus, we are already in good stead (if you took INF 511).

16.5.3 ANOVA Model Components: Means and Effects

- Here, we summarize/illustrate the **cell means model** and the **effects model with sum-to-zero coding**. We'll add coverage of **treatment coding, later**, for an overall presentation that is similar to our previous presentation of one-way ANOVA (one factor models).
- Much of the material here is adapted from [KNNL05, Sec. 19.2].
- We may not cover this section systematically for time. But, it's here for future reference. If there is any **primary message** here it is this: interactions are indicated by non-parallel lines (non-additive model), and we need to be careful when inferring lower order effects (e.g., main effects) in the presence of interactions that involve the main effects.

Two Way ANOVA (equal sample sizes $n_{ij} = n$)

TABLE 19.1
Age Effect but
No Gender
Effect, with No
Interactions—
Learning
Example.

Theoretical Example
Does (mean) time
to learn a task
depend on gender?
on age?

Recall Cell Means Model

$$Y_{ijk} = \mu_{ij} + \epsilon_{ijk}$$

$i = 1 \dots a$

$j = 1 \dots b$

$k = 1 \dots n_{ij}$

we're just looking
at cell means
and effects for now

		(a) Mean Learning Times (in minutes)			Row Average	Overall mean
		Factor B Age				
Factor A Gender		$j = 1$	$j = 2$	$j = 3$		
		Young	Middle	Old		
$i = 1$ Male		9 (μ_{11})	Cell mean 11 (μ_{12})	Column average 11 (μ_{13})	12 ($\mu_{1..}$)	12 ($\mu_{..}$)
$i = 2$ Female		9 (μ_{21})	mean 11 (μ_{22})	means 16 (μ_{23})	12 ($\mu_{2..}$)	12 ($\mu_{..}$)
Column average		9 ($\mu_{.1}$)	11 ($\mu_{.2}$)	16 ($\mu_{.3}$)	12 ($\mu_{..}$)	12 ($\mu_{..}$)
				marginal means		overall mean

(b) Main Gender Effects (in minutes)		(c) Main Age Effects (in minutes)	
$\alpha_1 = \mu_{1..} - \mu_{..} = 12 - 12 = 0$	$\alpha_2 = \mu_{2..} - \mu_{..} = 12 - 12 = 0$	$\beta_1 = \mu_{.1} - \mu_{..} = 9 - 12 = -3$	$\beta_2 = \mu_{.2} - \mu_{..} = 11 - 12 = -1$
$\alpha_i \equiv \mu_{i..} - \mu_{..}$ Factor A Effects		$\beta_j = \mu_{.j} - \mu_{..}$ Factor B Effects	$\beta_3 = \mu_{.3} - \mu_{..} = 16 - 12 = 4$

Definitions:

Marginal Means

$$A \quad \mu_{i..} \equiv \sum_{j=1}^b \mu_{ij} / b$$

$$B \quad \mu_{.j} \equiv \sum_{i=1}^a \mu_{ij} / a$$

$$\text{Overall Mean} \quad \mu_{..} \equiv \sum_{i=1}^a \sum_{j=1}^b \mu_{ij} / (ab)$$

check for yourself!

This page illustrates additive effects (not always the case!!!)

(*) More on Effects

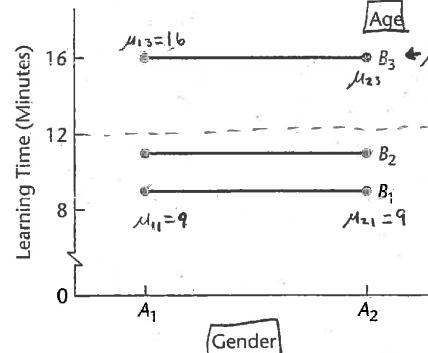
Definitions result in sum to zero:

$$\begin{aligned} \sum_{i=1}^a \alpha_i &= 0 \\ \sum_{j=1}^b \beta_j &= 0 \end{aligned}$$

These will become constraints on our 2-way factor effects model (coming soon)

FIGURE 19.3
Age Effect but
No Gender
Effect, with No
Interactions—
Learning
Example.

"Interaction" plot



$$\begin{aligned} M_{13} &= 16 & B_3 &\leftarrow M_{03} = \frac{M_{13} + M_{23}}{2} = 16 & \beta_3 &= 4 \\ M_{23} & & M_{13} & & & \\ M_{1..} &= 12 & & & & \\ \text{(in this ex.,)} & & & & & \\ M_1 &= M_2 = 12 & & & & \\ \Rightarrow \alpha_i &= 12 - 12 = 0 & & & & \\ i=1,2 & & & & & \end{aligned}$$

Interpretation of effects:

- α_i — on average (over levels of Factor B), the "blip" (effect) up/down from the overall mean $\mu_{..}$ due to being in the i th level of Factor A
- β_j — on average (over levels of Factor A), the "blip" (effect) up/down from the overall mean $\mu_{..}$ due to being in the j th level of Factor B

TABLE 19.2
Age and
Gender Effects,
with No
Interactions—
Learning
Example.

		(a) Mean Learning Times (in minutes)			
		Factor B—Age			
Factor A—Gender		$j = 1$ Young	$j = 2$ Middle	$j = 3$ Old	Row Average
$i = 1$	Male	11 (μ_{11})	13 (μ_{12})	18 (μ_{13})	14 ($\mu_{1..}$)
$i = 2$	Female	7 (μ_{21})	9 (μ_{22})	14 (μ_{23})	10 ($\mu_{2..}$)
Column average		9 ($\mu_{..1}$)	11 ($\mu_{..2}$)	16 ($\mu_{..3}$)	12 ($\mu_{...}$)

(b) Main Gender Effects (in minutes)		(c) Main Age Effects (in minutes)
$\alpha_1 = \mu_{1..} - \mu_{....} = 14 - 12 = 2$		$\beta_1 = \mu_{1..} - \mu_{....} = 9 - 12 = -3$
$\alpha_2 = \mu_{2..} - \mu_{....} = 10 - 12 = -2$		$\beta_2 = \mu_{2..} - \mu_{....} = 11 - 12 = -1$
		$\beta_3 = \mu_{3..} - \mu_{....} = 16 - 12 = 4$

FIGURE 19.4
Age and
Gender Effects,
with No
Interactions—
Learning
Example.

Tell-tale sign
of no interaction:
parallel lines

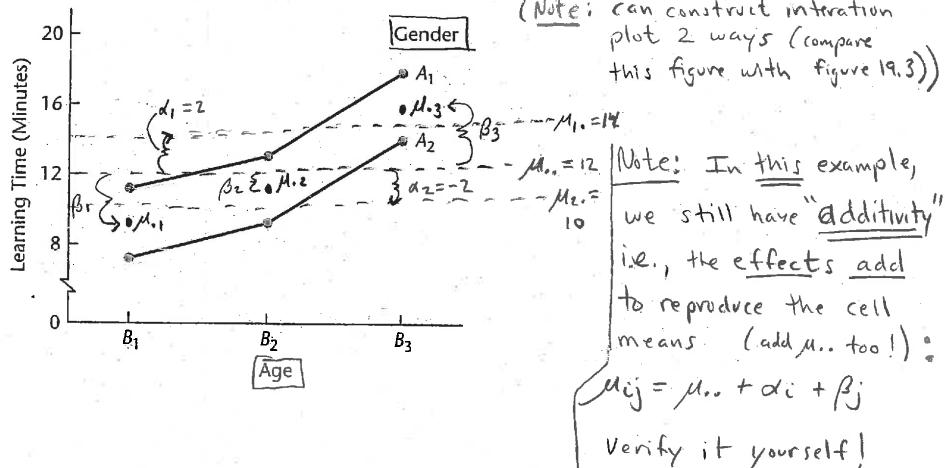


TABLE 19.3
Age and
Gender Effects,
with
Interactions—
Learning
Example.

		(a) Mean Learning Times (in minutes)			Main Gender Effect
		Factor B—Age			
Factor A—Gender		$j = 1$ Young	$j = 2$ Middle	$j = 3$ Old	Row Average
$i = 1$	Male	9 (μ_{11})	12 (μ_{12})	18 (μ_{13})	13 ($\mu_{1..}$)
$i = 2$	Female	9 (μ_{21})	10 (μ_{22})	14 (μ_{23})	11 ($\mu_{2..}$)
Column average		9 ($\mu_{..1}$)	11 ($\mu_{..2}$)	16 ($\mu_{..3}$)	12 ($\mu_{...}$)
Main age effect		-3 (β_1)	-1 (β_2)	4 (β_3)	

		(b) Interactions (in minutes)			Row Average
		$j = 1$	$j = 2$	$j = 3$	
$i = 1$		-1	0	1	0
$i = 2$		1	0	-1	0
Column average		0	0	0	0

All interaction effects

$(\alpha\beta)_{ij}$ must be zero before we say main effects are additive.

We have "non-additivity" of main effects, i.e., there is an interaction effect in this e.g.

Definition

Interaction effects

$$(\text{one symbol!}) (\alpha\beta)_{ij} \equiv \mu_{ij} - (\mu_{..} + \alpha_i + \beta_j)$$

"extent to which the additive model cannot reproduce the cell means"

We will have the additional sum-to-zero constraints:

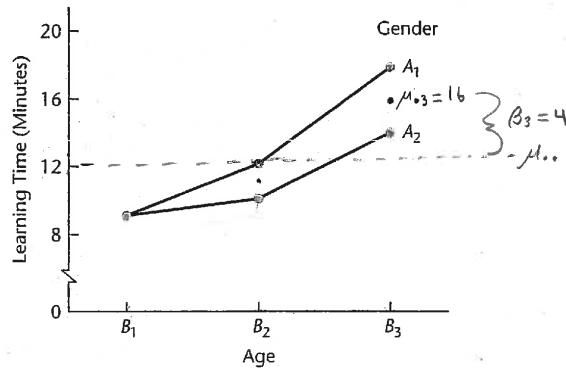
$$\sum_{i=1}^a (\alpha\beta)_{ij} = 0 = \sum_{j=1}^b (\alpha\beta)_{ij}$$

Why are these "important"?

FIGURE 19.5
Age and
Gender Effects,
with Important
Interactions—
Learning
Example.

Tell-tail sign
of interaction:

non-parallel
lines

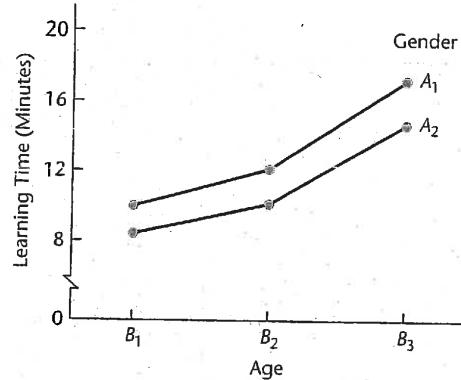


Interpretation of interaction effects: The "blip" up/down due to being in cell i, j after accounting for the average effect of Factor A level i and the average effect of Factor B level j .

TABLE 19.4
Age and
Gender Effects,
with
Unimportant
Interactions—
Learning
Example.

		Factor B—Age			Row Average
Factor A—Gender		$j = 1$ Young	$j = 2$ Middle	$j = 3$ Old	
$i = 1$	Male	9.75	12.00	17.25	13.00
	Female	8.25	10.00	14.75	11.00
Column average		9.00	11.00	16.00	12.00

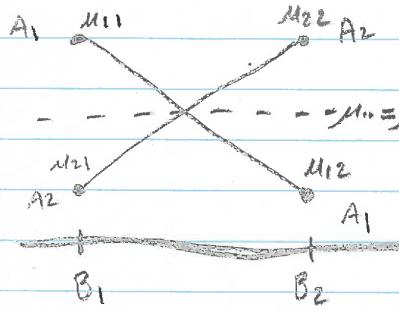
FIGURE 19.6
Age and
Gender Effects,
with
Unimportant
Interactions
(curves almost
parallel)—
Learning
Example.



Try computing
interaction effects.
They're small here.
Moreover, they're "unimportant"
(why?)

An illustrative e.g. of what can happen when main effects are used to characterize differences in factor level means when "bad" interactions exists.

2 Factors A, B each at 2 levels



Obviously, something interesting is going on with the mean response between levels of A within a level of B (simples are interesting), but using main effects to look at differences in mean

response averages the simples to get a misleading answer. Point: Beware of interpreting main effects when interactions are present.

- Simple effect of A at B_1 : $\mu_{21} - \mu_{11}$

- "Simple effect" of A at B_2 : $\mu_{22} - \mu_{12}$

- Using Main Effects:

$$\alpha_2 - \alpha_1 \quad (\text{why do this?})$$

$$= (\mu_{21} - \mu_{11}) - (\mu_{22} - \mu_{12})$$

$$= (\mu_{21} - \mu_{12}) \quad (\text{OK, I see})$$

$$= \frac{1}{2} \{ \mu_{21} + \mu_{22} - (\mu_{11} + \mu_{12}) \}$$

$$= \frac{1}{2} \{ \underbrace{(\mu_{21} - \mu_{11})}_{\text{Simple}} + \underbrace{(\mu_{22} - \mu_{12})}_{\text{Simple}} \}$$

$$= 0 \quad (\text{in this example})$$

828 Part Five Multi-Factor Studies

TABLE 19.6
Examples of
Different Types
of Interactions.

		(a) Productivity of Executives	
		Factor B—Authority	
Factor A—Pay		Small	Great
Low		50	72
High		74	75

		(b) Productivity of Executives	
		Factor B—Authority	
Factor A—Pay		Small	Great
Low		50	52
High		53	75

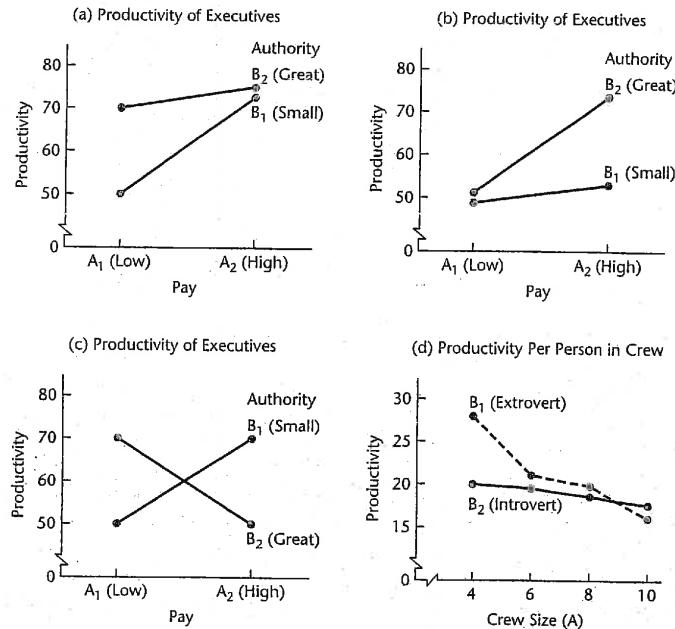
		(c) Productivity of Executives	
		Factor B—Authority	
Factor A—Pay		Small	Great
Low		50	72
High		72	50

		(d) Productivity per Person in Crew	
		Factor B—Personality of Crew Chief	
Factor A—Crew Size		Extrovert	Introvert
4 persons		28	20
6 persons		22	20
8 persons		20	19

More examples for your edification.

Chapter 19 Two-Factor Studies with Equal Sample Sizes 829

FIGURE 19.7
Treatment
Means Plots—
Examples of
Interactions
from
Table 19.6.



16.5.4 Effects Model: STZ: Example: Initial Analysis

We continue to use the **seaweed grazing example**, introduced above, to illustrate analysis in R using the factor effects model with sum-to-zero constraints. If you haven't already recognized our approach, it may be helpful to realize that we are proceeding analogously to our one-way analysis of the factor effects model with sum-to-zero constraint, covered previously in §15.1.6.

```
> ## Default constraint/coding is treatment...:
>getOption("contrasts")

[1] "contr.sum"  "contr.poly"

> ## ...unless factors have a different, overriding
> ## contrasts attribute...nope:
> sapply(case1301.df, attr, which="contrasts")

$Cover
NULL

$Block
NULL

$Treat
NULL

> ## Again, we can set constraints/coding globally...
> options(contrasts = rep("contr.sum",2))
>getOption("contrasts")

[1] "contr.sum"  "contr.sum"

> ## ...or, we can assign (overriding) constraints/coding
> ## (perhaps not the same) to individual factors:
> contrasts(case1301.df$Block)<- contr.sum(levels(case1301.df$Block))
> contrasts(case1301.df$Treat)<- contr.sum(levels(case1301.df$Treat))
> sapply(case1301.df, attr, which="contrasts")
```

```
$Cover
NULL

$Block
 [,1] [,2] [,3] [,4] [,5] [,6] [,7]
B1    1    0    0    0    0    0    0
B2    0    1    0    0    0    0    0
B3    0    0    1    0    0    0    0
B4    0    0    0    1    0    0    0
B5    0    0    0    0    1    0    0
B6    0    0    0    0    0    1    0
B7    0    0    0    0    0    0    1
B8   -1   -1   -1   -1   -1   -1   -1

$Treat
 [,1] [,2] [,3] [,4] [,5]
C     1    0    0    0    0
L     0    1    0    0    0
Lf    0    0    1    0    0
Lff   0    0    0    1    0
f     0    0    0    0    1
ff    -1   -1   -1   -1   -1
```

Now, let's fit our first multi-way ANOVA model. We'll skip inspection of the (non-redundant) \mathbf{X} in R; we've looked at (some illustration of) it, above (perhaps in class again), and have illustrated the `model.matrix` function before, so you should be able to reproduce it by hand or with R.

```
> case1301.lm<- lm(Cover ~ Block + Treat + Block:Treat,
+                      data=case1301.df)
> ## Typical regression summary, often of subsidiary interest
> ## in an ANOVA context:
> summary(case1301.lm)

Call:
lm(formula = Cover ~ Block + Treat + Block:Treat, data = case1301.df)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-18.00	-3.62	0.00	3.62	18.00

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	28.625	0.965	29.67	< 2e-16 ***
Block1	-19.125	2.553	-7.49	1.3e-09 ***
Block2	-15.375	2.553	-6.02	2.3e-07 ***
Block3	12.625	2.553	4.95	9.7e-06 ***
Block4	26.375	2.553	10.33	8.6e-14 ***
Block5	-4.375	2.553	-1.71	0.0930 .
Block6	8.625	2.553	3.38	0.0015 **
Block7	-5.625	2.553	-2.20	0.0324 *
Treat1	23.375	2.157	10.84	1.7e-14 ***
Treat2	-9.375	2.157	-4.35	7.2e-05 ***
Treat3	-12.125	2.157	-5.62	9.5e-07 ***
Treat4	-20.875	2.157	-9.68	7.3e-13 ***
Treat5	14.125	2.157	6.55	3.6e-08 ***
Block1:Treat1	-14.375	5.708	-2.52	0.0152 *
Block2:Treat1	-8.125	5.708	-1.42	0.1610
Block3:Treat1	9.875	5.708	1.73	0.0900 .
Block4:Treat1	16.125	5.708	2.83	0.0069 **
Block5:Treat1	-4.125	5.708	-0.72	0.4734
Block6:Treat1	5.875	5.708	1.03	0.3085
Block7:Treat1	-13.375	5.708	-2.34	0.0233 *
Block1:Treat2	3.875	5.708	0.68	0.5005
Block2:Treat2	3.625	5.708	0.64	0.5284
Block3:Treat2	11.125	5.708	1.95	0.0571 .
Block4:Treat2	-14.625	5.708	-2.56	0.0136 *
Block5:Treat2	7.125	5.708	1.25	0.2180
Block6:Treat2	-4.375	5.708	-0.77	0.4471
Block7:Treat2	-6.625	5.708	-1.16	0.2515
Block1:Treat3	6.625	5.708	1.16	0.2515
Block2:Treat3	3.375	5.708	0.59	0.5571
Block3:Treat3	-9.125	5.708	-1.60	0.1164
Block4:Treat3	-3.875	5.708	-0.68	0.5005
Block5:Treat3	-5.125	5.708	-0.90	0.3737
Block6:Treat3	9.375	5.708	1.64	0.1070
Block7:Treat3	-2.875	5.708	-0.50	0.6168
Block1:Treat4	12.875	5.708	2.26	0.0287 *
Block2:Treat4	11.625	5.708	2.04	0.0472 *

```

Block3:Treat4 -12.875    5.708   -2.26   0.0287 *
Block4:Treat4 -19.625    5.708   -3.44   0.0012 **
Block5:Treat4  2.125     5.708   0.37    0.7113
Block6:Treat4 -2.875     5.708   -0.50   0.6168
Block7:Treat4  7.375     5.708   1.29    0.2025
Block1:Treat5 -6.125     5.708   -1.07   0.2886
Block2:Treat5 -4.875     5.708   -0.85   0.3973
Block3:Treat5  0.125     5.708   0.02    0.9826
Block4:Treat5 16.875     5.708   2.96    0.0048 **
Block5:Treat5 -4.875     5.708   -0.85   0.3973
Block6:Treat5 -5.875     5.708   -1.03   0.3085
Block7:Treat5 10.875     5.708   1.91    0.0627 .
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Residual standard error: 9.45 on 48 degrees of freedom
 Multiple R-squared: 0.919, Adjusted R-squared: 0.84
 F-statistic: 11.6 on 47 and 48 DF, p-value: 1.12e-14

```

> ## Typical (sequential / Type I) ANOVA table, often receives primary
> ## interest in an ANOVA context.
> anova(case1301.lm)

```

Analysis of Variance Table

Response: Cover

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Block	7	19106	2729	30.55	1.3e-15 ***
Treat	5	23045	4609	51.58	< 2e-16 ***
Block:Treat	35	6612	189	2.11	0.0081 **
Residuals	48	4289	89		

```

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

- **X Matrix.** What does the R X matrix look like? (Not in output...too big; we should have a fair idea of what it looks like at this point, given previous discussion.)
- **LS Estimates.** What are the LS estimators/estimates of the parameters?

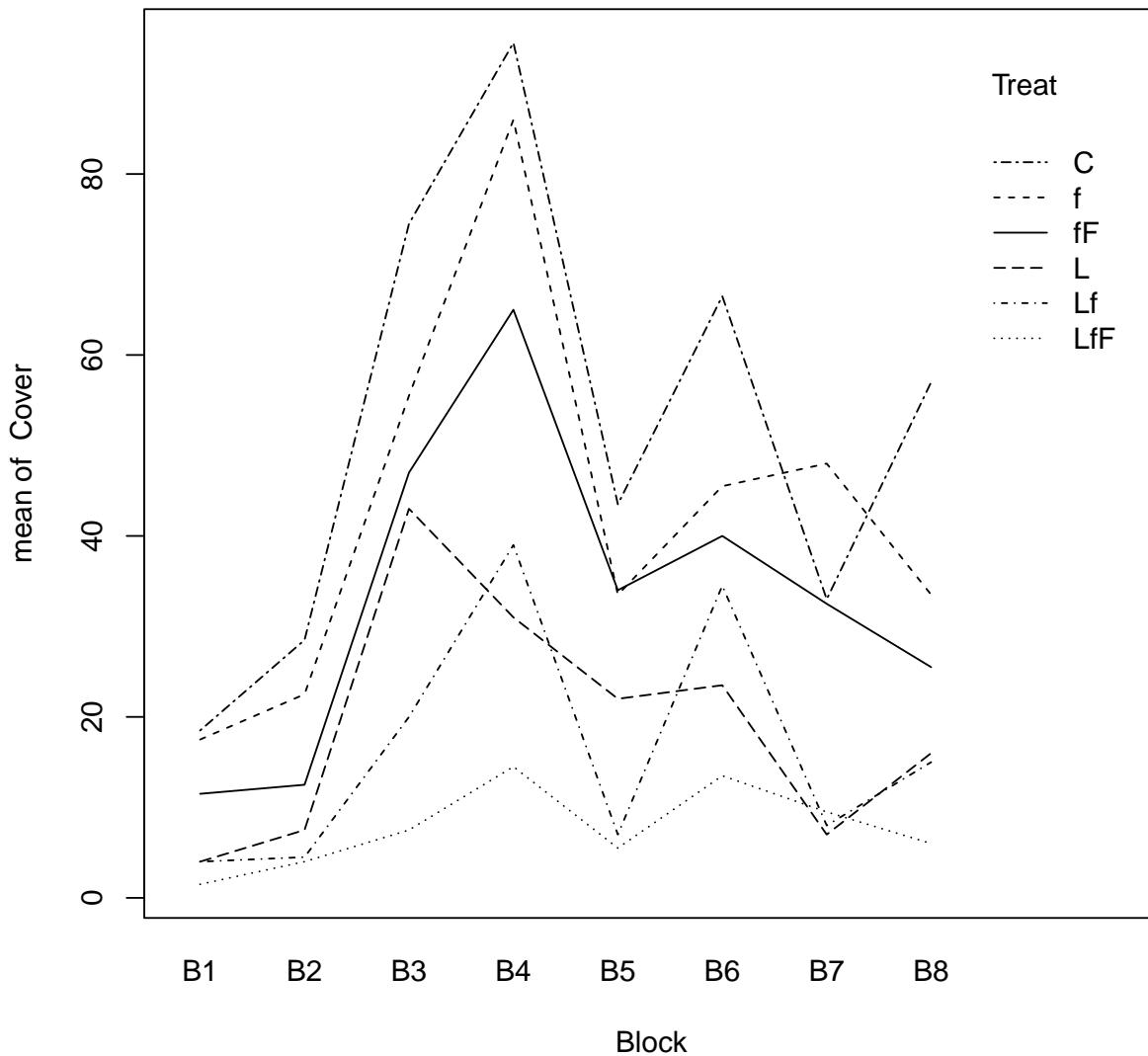
- **Interpretation.** How do we interpret these (estimated) parameters? Understanding parameters may help when making detailed inference, beyond a standard ANOVA table, i.e., for general $\mathbf{C}\boldsymbol{\beta}$.
- **Compare to Previous Model.** How do these estimates compare to those for the cell means model? We did not perform a cell-means model analysis for this two-way case, so we did not directly discuss $\hat{\mu}_{ij}$. Still, it's a fair question for someone who would brandish an ANOVA.
- **Estimated Error Variance.** What is the estimator/estimate of the variance, σ^2 (or of the standard deviation, σ)? How would this compare with the estimate given by the cell means analysis if we were to have performed one? (Same.)
- **Estimated Standard Errors.** What are the estimated standard errors of the estimators of the parameters? (Think in terms of diagonal elements of a certain matrix, as we have discussed before.)
- **Default t-tests.** R gives default t-tests for each of the parameters assuming a null value of zero by default. How are these tests computed? Are these tests interesting?
- **p-values.** How are the p-values for the above tests computed?
- **Etc.** What are the remaining quantities in the output of the `summary` function? (Overall F-test and R^2 okay here.)
- **Scope of Inference?** (See unnumbered section, Scope of Inference: Summary, at the end of INF 511 note chapter 5.)

16.5.5 Effects Model: STZ: Example: Diagnostics

- **Interaction?** We might conclude that there are significant interaction effects as indicated by the default F-test given in the (sequential / Type

- I) anova output, which would dictate the strategy for further analysis, as discuss above.
- **Perhaps Not.** Perhaps we are being a bit hasty. Let's **diagnose** our model first before we proceed. (We follow [RS13, Sec. 13.3].)
 - **Interaction?** Below, we construct **interaction plots** based on averages of Cover at each treatment (not Treat!) level (i.e., $\bar{Y}_{ij\cdot}$, which, incidentally, are the least squares fitted values, \hat{Y}_{ijk} (either $k = 1$ or $k = 2$ of course!) for the cell means model or the saturated (non-additive) effects model. (Recall that our non-additive model is equivalent to the cell means model so that treatment (cell) averages minimize the LS criterion.).
 - **Something Else Going On?** In other words, we plot fitted values versus our covariates (factors) to see if we can diagnose non-additivity or perhaps some departure from model assumptions, i.e., does there appear to be interaction or is there something else going on? You might want to revisit Section 16.5.3 for its discussion of interaction before proceeding.

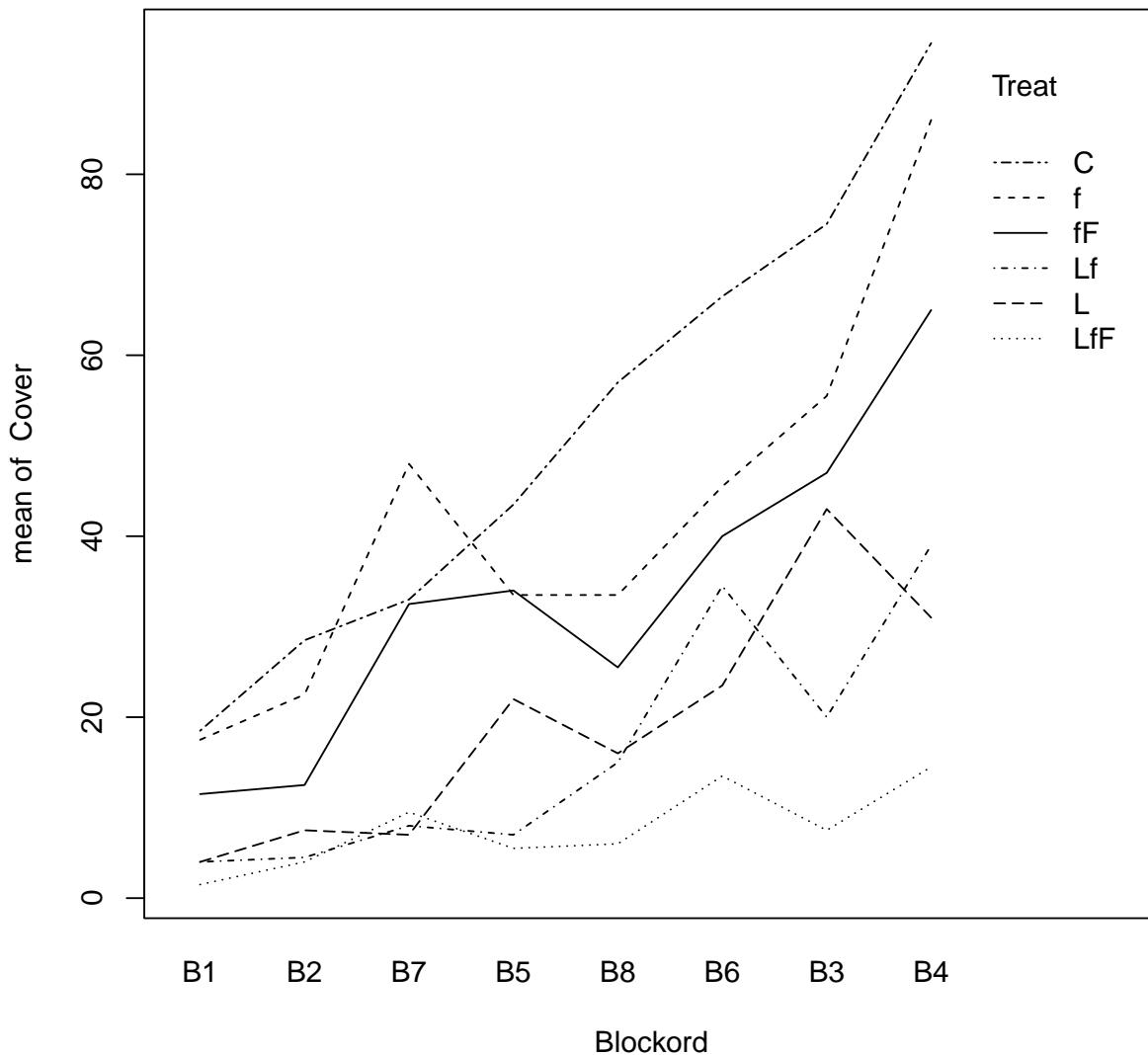
```
> ## Similar to Display 13.7 in R&S:  
> with(case1301.df, {interaction.plot(x.factor=Block,  
+                                         trace.factor=Treat,  
+                                         response=Cover)})
```



Let's edit the interaction plot to be more like [RS13, Display 13.7] (in terms of Block levels, at least).

```
> ## Perhaps re-ordering may help (more like R&S Display 13.7):
> case1301.df$Blockord<- with(case1301.df, reorder(Block, Cover, mean))
```

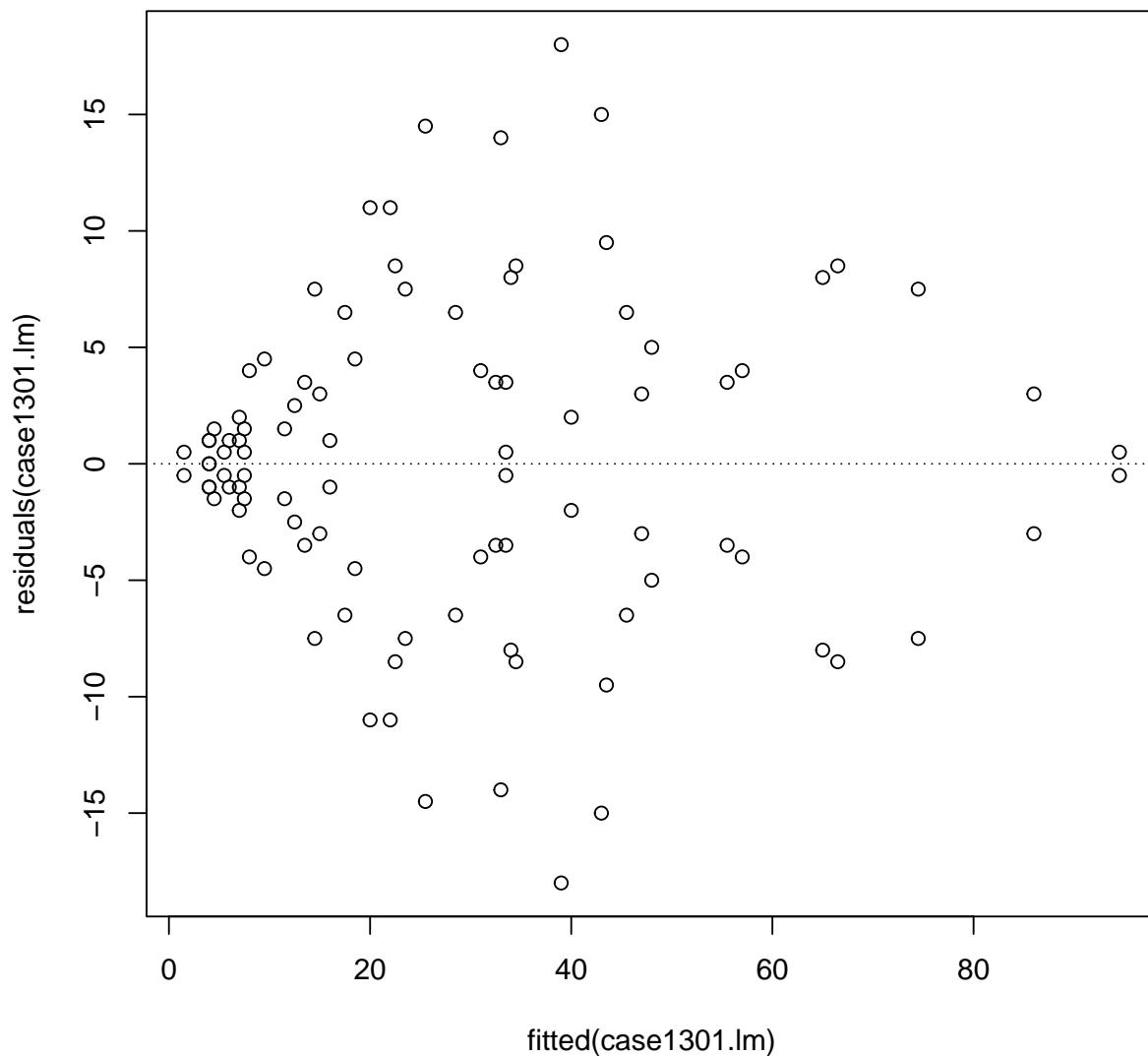
```
> with(case1301.df, {interaction.plot(x.factor=Blockord,
+                                     trace.factor=Treat,
+                                     response=Cover)})
```



- **Non-Constant Variance.** I (and R&S) think that we may not be seeing interaction, but may be seeing non-constant variance, typically seen in

proportion or percent data. The interaction plot may not be the best way to illustrate non-constant variance—it's using averages instead of observations—still, it does hint at non-constant variance. As we may recall, **residuals** are often a good way to diagnose non-constant variance ([RS13, Display 13.8]).

```
> ## Compare to R&S Display 13.8 (classic):
> plot(residuals(case1301.lm) ~ fitted(case1301.lm))
> abline(h=0, lty=3)
```



- **Theory.** Also, theory suggests that proportions or percentages have non-constant variance. Further, the pattern in the residual plot above is entirely consistent with what the theory suggests—classic. Based on this theory, a typical transformation of a proportion, Y , is the **logit (log**

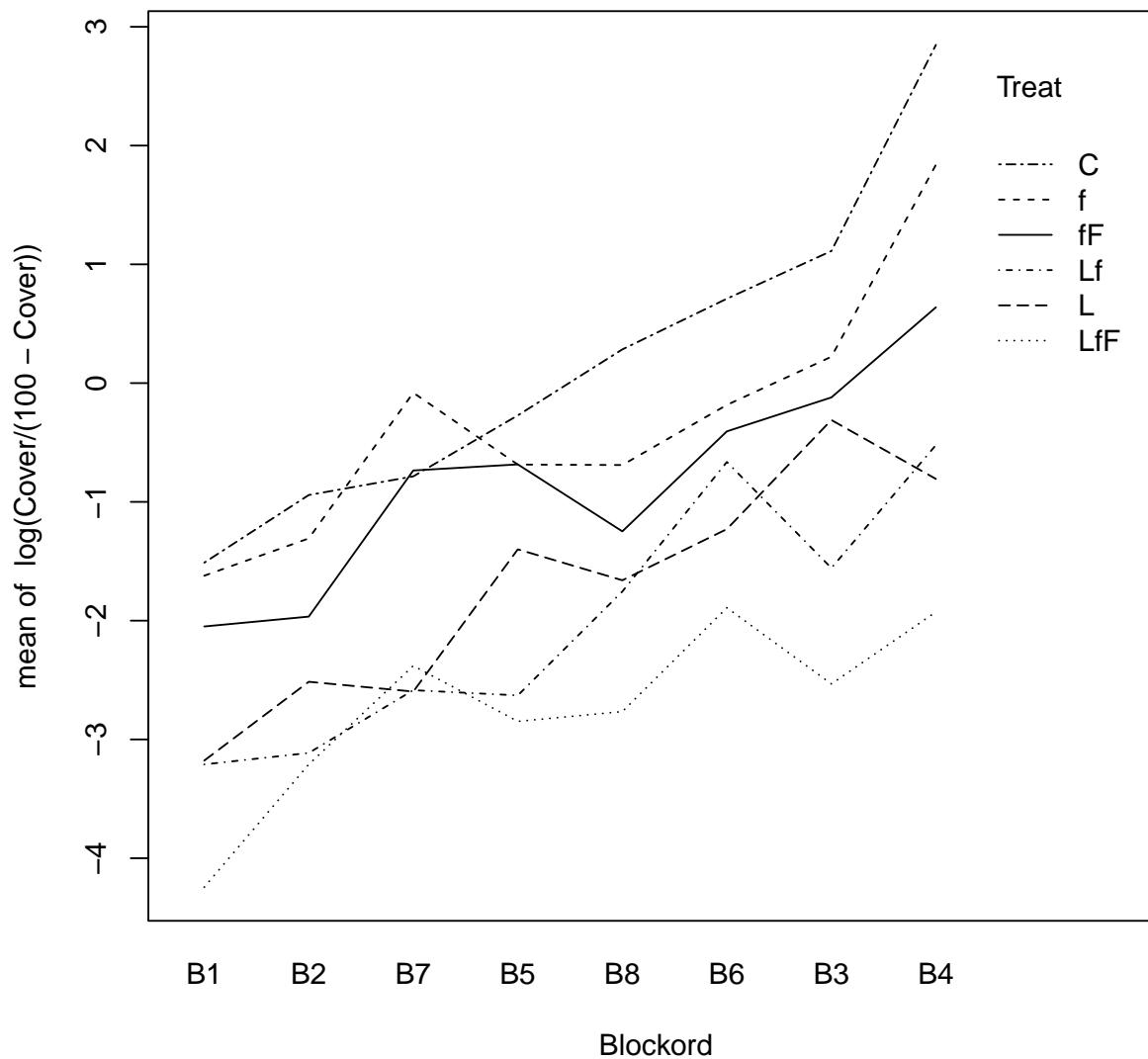
odds) transformation:

$$\text{logit}(Y) = \log(Y/(1 - Y))$$

[KNNL05, Sec. 18.5].

- **Interaction?** The interaction plot, based on the transformed response, suggests that the transformation stabilizes variance. (Recall that the response variable, Cover, is a percentage, not a proportion, hence the 100...). And, the interaction somehow does not look as serious.

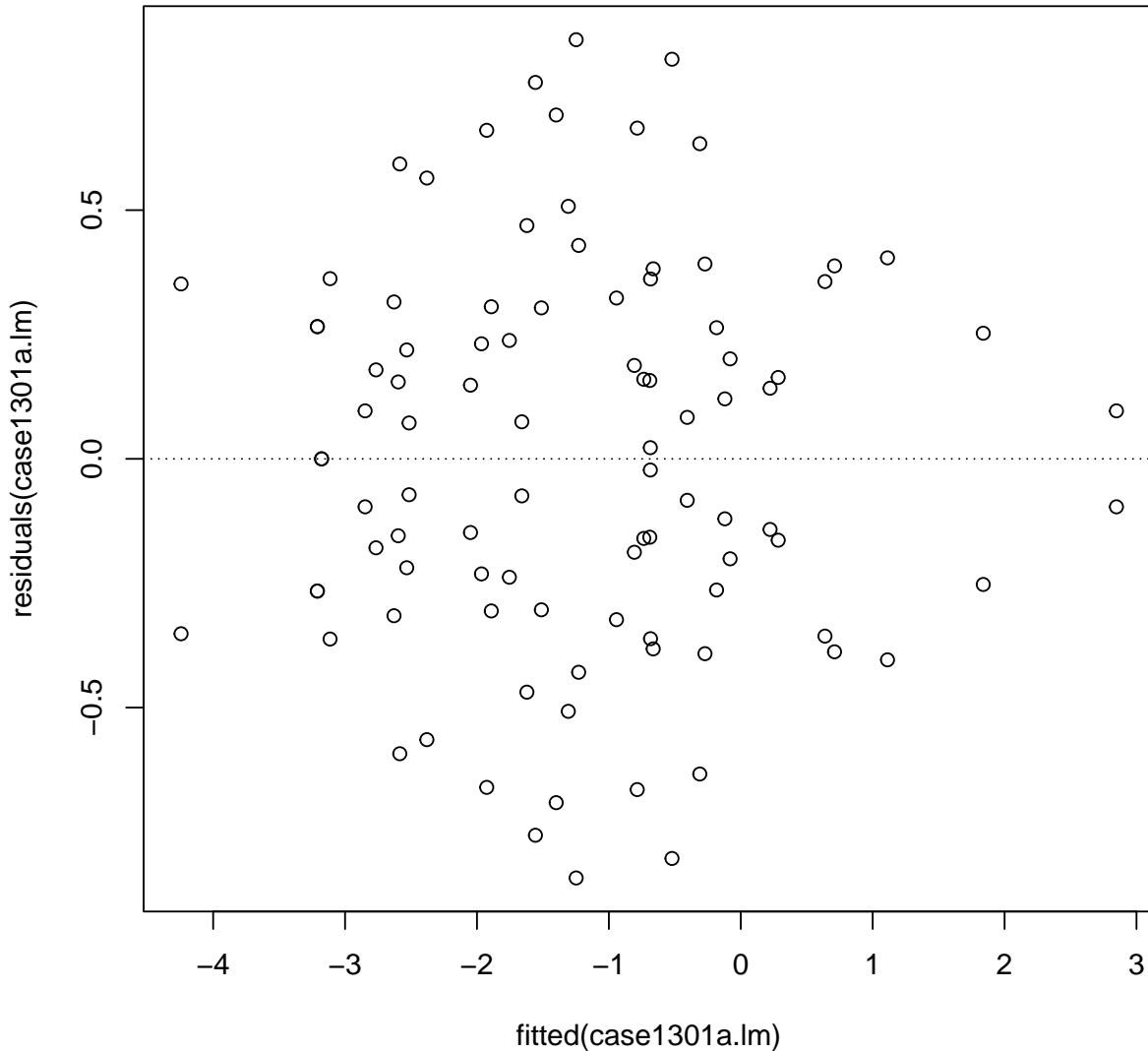
```
> ## Compare to R&S Display 13.9...better?
> with(case1301.df, {interaction.plot(x.factor=Blockord,
+                                         trace.factor=Treat,
+                                         response=log(Cover/(100-Cover))))})
```



- **Variance Stabilized?** Again, residuals (next chunk) may allow a better diagnosis. Your eye may suggest remaining heteroskedasticity, but I think the residual pattern may be attributed more to the concentration of values near mid-level proportions, and less to residual heteroskedasticity. In any case, I am not too concerned about the pattern.

- **Apophenia?** Incidentally, why the peculiar-looking reflection of points about the horizontal zero line? If you turn your head sideways and let your eyes wander a bit, you can see an elderly lady or a young lady, depending on which way you turn your head...just kidding. Relatedly, by the way, this symmetry of residuals can be seen in the brief residual diagnostic of the `summary` function for the non-transformed case, above (and for the transformed case had we used `summary` in this case).

```
> case1301a.lm<- lm(log(Cover/(100-Cover)) ~ Block + Treat +
+                         Block:Treat,
+                         data=case1301.df)
> ## Improved
> plot(residuals(case1301a.lm) ~ fitted(case1301a.lm))
> abline(h=0, lty=3)
```



- **Parallel Up to Chance?** Our primary message of (skipped?) Section 16.5.3) is that additivity (i.e., no interaction) is exhibited (ideally!) by parallel lines in an interaction plot, indicating that mean differences among Factor A (B) levels are the same for each level of Factor B (A) (and vice versa). In the last interaction plot, above, differences of aver-

ages (of transformed values) among the levels of Treat are not *exactly* the same across the levels of Block, but it seems reasonable to argue that the differences are *comparable* across the levels of Block, and we may suggest that the departure from parallel lines may simply be due to sampling error variation (recall that we only have $n_{ij} = 2$ observations to estimate the treatment (cell) means).

16.5.6 Effects Model: STZ: Example: ANOVA For Common $C\beta$

- **Continue on the Logit (Log Odds) Scale.** Following the above analysis, let's continue on the logit scale. The next chunk shows a standard (sequential / Type I) ANOVA table, which verifies the ANOVA table shown in [RS13, Display 13.10].

```
> ## Repeat sequential (type I) anova with fit to transformed
> ## response. Interaction?
> anova(case1301a.lm)

Analysis of Variance Table

Response: log(Cover/(100 - Cover))
          Df  Sum Sq Mean Sq F value Pr(>F)
Block       7    76.2   10.89   35.96 <2e-16 ***
Treat       5    97.0   19.40   64.06 <2e-16 ***
Block:Treat 35    15.2    0.44    1.44    0.12
Residuals   48    14.5    0.30
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **Skip Details.** We skip the estimated parameter printouts, which we covered before the transformation. Interpretation is the same, except it's now in terms of $\text{logit}(\text{proportion cover})$ (log odds of proportion cover). Besides, as I should have said earlier, the parameter estimates in ANOVA are typically of subsidiary interest. Similarly, we skip \mathbf{X} and $\boldsymbol{\beta}$ (or estimate thereof), as we have discussed these previously.

16.5.7 Pit Stop: What is ANOVA?

- **What is ‘ANalysis of VAriance (ANOVA)?** After all, we've been saying it's just a linear model like any other.
- **ANOVA Table Convenience.** The (sequential / Type I) ANOVA table, illustrated above, simply summarizes typical tests of the significance of factor variables or their interactions, as we've seen above. In other words, the ANOVA table summarizes tests of particular linear combinations associated with factors and interactions so that you don't have to figure out $\mathbf{C}\boldsymbol{\beta}$. It is often featured in a course like this. Fine. But, we don't need it, because we have our F vs. R and linear combinations approaches to infer more generally about linear combination of parameters. Still, the tests are likely to be of use, so we should discuss the table, at least for historical reasons and because ANOVA tables are featured in textbooks and computer printouts. (Admittedly, it may be relatively convenient as the numbers of factors/interactions increase.)
- **SS Decomposition.** In any linear model, we have a decomposition of the total sum of squares into a component due to the regression/ANOVA/linear mean model and to residuals (see [Far14, §3.2] and INF 511 notes §3.2.1)

$$\text{TSS} = \text{SS}_{\text{reg}} + \text{RSS}.$$

In the **BALANCED** two-way case, we can further decompose “regression sum of squares” SS_{reg} into components associated with the factor variables and their interaction(s):

$$SS_{reg} = SSA + SSB + SSAB,$$

where

- **SSA** is “factor A sum-of-squares,” a measure of the response variability that is associated with Factor A. For our running example (see the latest output from the `anova` function), this is 76.239 (treating Block as A and Treat as B).
 - **SSB** is “factor B sum-of-squares,” a measure of the response variability that is associated with Factor A. 96.993.
 - **SSAB** is “interaction sum-of-squares,” a measure of the response variability that is associated with the interaction between Factors A and B. 15.230.
 - [RS13, Display 13.10] shows SS_{reg} as “Between groups” sum-of-squares: 188.4622, which is, in this **BALANCED** case, the sum of its constituent SS. Computer printouts may or may not show SS_{reg} .
- **df Decomposition.** Following [Far14, §3.2] and INF 511 notes §3.2.1, we also have a corresponding decomposition of the total degrees of freedom (using our specialized notation here):

$$(n_T - 1) = (ab - 1) + (n_T - ab),$$

where, analogously to the decomposition of SS_{reg} , we can further decompose its degrees of freedom, $(ab - 1)$, as

$$(ab - 1) = (a - 1) + (b - 1) + (a - 1)(b - 1),$$

where

- $(n_T - 1)$ is the “**total degrees of freedom**,” as in the one-way case; 95 in the running example, though not shown directly by anova.
 - $(ab - 1)$ (47) is the “**regression degrees of freedom**” (we have $ab = (8)(6) = 48$ treatment levels, right?)
 - $(n_T - ab)$ is the “**error degrees of freedom**” (or “residual degrees of freedom”); 48.
 - $(a - 1)$ is “**factor A degrees of freedom**”; 7.
 - $(b - 1)$ is “**factor B degrees of freedom**”; 5.
 - $(a - 1)(b - 1)$ “**AB interaction degrees of freedom**”; 35.
 - Note that the degrees of freedom associated with A, B, and AB, are merely the **number of columns** associated with each factor or interaction in the (non-redundant) **X** matrix.
- **Mean Squares, Etc..** The above sum-of-squares and degrees of freedom are used to construct further entries in the standard ANOVA table.
 - **Total, Regression, & Error MS.** As in the one-way case, we divide sums-of-squares by their respective degrees of freedom to get mean squares, MST , MS_{reg} and MSE , although MST is not typically used (it's just the sample variance of the Y values), and these mean squares are **not** additive like the above SS's and df's. Again, [RS13, Display 13.10] shows MS_{reg} as “Between group” mean square and MSE as “Within group” mean square. (We presented one-way ANOVA table examples in the previous lecture chapter 15 on without much discussion; and, again, see [Far14, §3.2].)
 - **Regression Related MS.** In addition, in the two-way case, we have MS_A , MS_B , and MS_{AB} (mean square for Block, mean square for Treat and mean square for Block:Treat interaction in the running example). Look at [RS13, Display 13.10] or the last anova output for values. Again, computer programs may not show SS_{reg} or MS_{reg} and usually do not show TSS or MST.

- **F, p-value, etc..** Also, we obtain various F-statistics and associated p-values, computed under a null distribution, which we will discuss in class.
- **Again, Convenience.** Note that the presentation of the ANOVA table, with the ratio of mean squares as F-statistics, seems somehow “classic” (dated?) to me. (Granted, the tests are often of interest.) As mentioned, we can (have/will) use the linear combinations ($C\beta$) approach (or F v R approach) to get the same F-statistics. In other words, the ANOVA table may be convenient, but is somehow extra, automated output that is not necessary given the alternative approaches that we’ve discussed (F v R or linear combinations). Still, with more complicated, higher-way layouts, standard ANOVA tables may be convenient.
- Note, ANOVA tables may be ambiguous about what is being tested, in either the **BALANCED** or **UNBALANCED** cases; see Section 16.7, below, for more about this.
- **Details.** Note, we skipped some details, which typically accompany a discussion of “hand computations”. Though an argument may be made that discussion of these hand computations may facilitate conceptual understanding, ultimately, I believe they serve little purpose and in some sense seem out of date to me. See [KNNL05, Sec. 19.4] for more details behind *balanced* ANOVA hand computations.

16.5.8 Effects Model: STZ: Example: F v R & $C\beta$ Approach

- **Alternative to (Convenient) ANOVA Table.** Before omitting interaction effects and proceeding to infer about grazing effects, let’s, once again, illustrate the use of our F v R (aka extra sum of squares) and linear combinations ($C\beta$) approaches as alternatives to using the automatic F-tests of the ANOVA table, illustrated in the previous section.

- **Illustrate with Interaction.** In particular, we look at the interaction term. Of course, we already know, from the ANOVA table and our previous discussion, that there does not appear to be serious, significant interaction, on the logit scale. But, again, it doesn't hurt to continue building our familiarity with these alternative, more general approaches to inference. We will use these later, when there is not an automatically generated ANOVA table that happens to summarize interesting inferences for us.

- **F v R Approach Using anova.**

```
> ## Reduced model, without interaction term (using logit transform fit):
> case1301aR.lm<- update(case1301a.lm,. ~ . - Block:Treat)
>
> ##F v R test as usual:
> anova(case1301aR.lm, case1301a.lm)
```

Analysis of Variance Table

Model	log(Cover/(100 - Cover))	Block	Treat			
Model 1:	\sim					
Model 2:	\sim	Block	Treat + Block:Treat			
	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	83	29.8				
2	48	14.5	35	15.2	1.44	0.12

```
> ## which gives the same test for interaction as given
> ## by anova(case1301a.lm), as promised (see previous)
```

- **C β Approach.** We use `glh.test` as we have before (in INF 511 anyway). We skip the corresponding “matrix hand computation,” as we have done in INF 511. You should be able to do these. In any case,

this approach is a bit less convenient because it requires us to recall the non-redundant (coded or “regression”) form of \mathbf{X} . More particularly, it requires us to interpret the parameters in the associated β vector—after all, we want to infer about particular linear combinations of parameters, so we should know what β is! Which elements of β are associated with the interaction, i.e., with the $(\alpha\beta)_{ij}$, i.e., with the Treat:Block model formula term?

```
> ## I build C matrix in stages (not necessary, but seems convenient).
> ##
> a<- 8; b<- 6 ## number of factor levels
> Ca<- matrix(0,nrow=(a-1)*(b-1), ncol=a-1)
> Cb<- matrix(0,nrow=(a-1)*(b-1), ncol=b-1)
> Cab<- diag((a-1)*(b-1))
> Cmat<- cbind(0,Ca,Cb,Cab) ## C matrix
> d<- rep(0,(a-1)*(b-1)) ## null CBeta value
> library(gmodels)
> glh.test(reg=case1301a.lm, cm=Cmat, d=d)
```

```
Test of General Linear Hypothesis
Call:
glh.test(reg = case1301a.lm, cm = Cmat, d = d)
F = 1.4369, df1 = 35, df2 = 48, p-value = 0.1209

> ## Result is same as F v R approach or anova(case1301a.lm)
> ## of course!
```

16.5.9 Effects Model: STZ: Example: Summary So Far

- **Broad Inferences.** We’ve seen how traditional ANOVA tables are used to present tests for certain, commonly interesting, linear combinations of parameters, which we can also infer about using our usual F v R and $C\beta$ approaches.

- **More Detailed Inferences To Come.** However we approached these inferences, results suggest that we feel comfortable proceeding to infer about grazing effects with the additive model, i.e., without the interaction term.
- **But, First, Treatment Coding.** But, before we do that, we mimic our one-way presentation and cover treatment constraints/coding. While, perhaps, not as popular as effects sum-to-zero constraints/coding, treatment constraints/coding is the default coding in R. I think it's worth knowing about both, obviously.

16.6 Effects Model: Treatment Constraints/Coding

We essentially repeat the above sum-to-zero analyses, now in terms of treatment constraints/coding as an alternative way to resolve redundancy among the columns of \mathbf{X} . In particular, we set particular parameters to zero:

$$\begin{aligned}\alpha_1 &= 0 \\ \beta_1 &= 0 \\ (\alpha\beta)_{1j} &= 0 \quad j=1, \dots, b, \\ (\alpha\beta)_{i1} &= 0 \quad i=1, \dots, a.\end{aligned}$$

In other words, similar to the treatment constraint/coding in the one-way case, we work with the remaining α_i , β_j , $(\alpha\beta)_{ij}$ $i = 2, \dots, a$ $j = 2, \dots, b$. Note that R sets the first effect parameter to zero by default; SAS sets the last effect parameter to zero when using treatment coding.

- What does the non-redundant β look like? (more in class)
- What does the non-redundant \mathbf{X} look like? (more in class) (BTW,

all interaction term (re-coded) columns can be obtained by multiplying (element-wise!) each (re-coded) column associated with one factor with each (re-coded) column associated with another factor, just as with the sum-to-zero constraints/coding.

- Again, we are simply performing regression with specially coded “ \mathbf{X} ” variables, one for each column in \mathbf{X} , and with associated coefficients having a particular interpretations. Incidentally, what is the “reference cell”? (Recall treatment coding is aka cell reference coding.)
- We’ll see how to implement this coding/constraints in R, shortly.
- Higher way?
- Non-additive model?

16.6.1 Effects Model: Trmt Coding: Parameter Interpretation

Notice (if we have not already) the interpretation of parameters in this factor effects model with treatment (cell reference) constraints/coding:

- As with sum-to-zero constraints/coding, we can relate the cell means and effects as

$$\mu_{ij} = \mu_{..} + \alpha_i + \beta_j + (\alpha\beta)_{ij}.$$

Indeed, this (non-additive or saturated...more shortly) effects model is equivalent to the cell means model and to the effects model with sum-to-zero constraints/coding: their respective \mathbf{X} matrices span the same covariate/regressor space, each gives the same predicted values, same residuals, same estimate of σ^2 , same overall F-test...perhaps more on this point later.

- **Overall Effect.** If we look at $i = 1$ and $j = 1$, we see that

$$\mu_{11} = \mu_{..},$$

so that, now, with treatment constraints, $\mu_{..}$ may still, of course, be considered an overall effect common to all observations, but we see now it is the mean of the **reference cell or reference treatment** defined by the first level of factor A and the first level of Factor B.

- **Main Effects of Factor A.** With treatment constraints, for $j = 1$, we see

$$\mu_{i1} = \mu_{..} + \alpha_i,$$

or

$$\alpha_i = \mu_{i1} - \mu_{..},$$

or

$$\alpha_i = \mu_{i1} - \mu_{11},$$

so that, α_i may still be interpreted as the deviation from the overall effect (not the overall mean now!) due to being in the i th level of factor A, but, more particularly, it is the deviation from the mean of the reference cell due to being in the i th level of Factor A (because we're not changing the reference level of B, $j = 1$, here). Again, inference for main effects may not be sensible in the presence of interaction.

- **Main Effects of Factor B.** With treatment constraints, for $i = 1$, we see

$$\mu_{1j} = \mu_{..} + \beta_j,$$

or

$$\beta_j = \mu_{1j} - \mu_{..},$$

or

$$\beta_j = \mu_{1j} - \mu_{11},$$

so that, β_j may still be interpreted as the deviation from the overall effect (not the overall mean now!) due to being in the j th level of factor B, but, more particularly, it is the deviation from the mean of the reference cell due to being in the j th level of Factor B (because we're not changing the reference level of A, $i = 1$, here). Again, inference for main effects may not be sensible in the presence of interaction.

- **Interaction Effects.** For $i, j > 1$, we have

$$\begin{aligned} (\alpha\beta)_{ij} &= \mu_{ij} - (\mu_{..} + \alpha_i + \beta_j) \\ &= \mu_{ij} - (\mu_{11} + \alpha_i + \beta_j) \end{aligned}$$

so that $(\alpha\beta)_{ij}$, again, is the additional effect, beyond the main effects, necessary to reproduce the ij th mean, now starting from the reference cell mean, not from the overall mean. Similar to the interpretation for interaction effects in the sum-to-zero constraints/coding case, if we hold, e.g., j constant, we see that we may interpret $(\alpha\beta)_{ij}$ as an additional effect, beyond main effects, of the i th level of A, due to being at level j of factor B. (Or, If we hold, e.g., i constant, we see that we may interpret $(\alpha\beta)_{ij}$ as an additional effect of the j th level of B, due to being at level i of factor A.) In other words, the effects of factor A (B) depends on the level of factor B (A), and we say that the factors **interact** and call $(\alpha\beta)_{ij}$ an **interaction effect**. Again, we've discussed interaction (Section 16.5.3), and we have warned about the potential nonsensicality of inferring main effects in the presence of certain interaction effects.

16.6.2 (Non-)Additive Model & Saturation

- **Additivity, Non-Additivity, Saturation.** The same discussion as we gave with the sum-to-zero effects model applies equally here: The effects model presented here, with (non-zero) interaction effects, $(\alpha\beta)_{ij}$, is often referred to as a **non-additive** model because we cannot reproduce the treatment means, μ_{ij} , by *adding* main effects (and overall effect...now

not the overall mean but the reference cell mean) alone; we also need $(\alpha\beta)_{ij}$ to reproduce treatment means. We also say that the model is **saturated** in the sense that we cannot specify further (linear) mean model complexity/flexibility; we have reached the “saturation point” of (linear) mean model complexity by allowing each treatment level to have any value whatsoever. We might even consider the notion of model **oversaturation**: the case of having too much complexity in the sense of not being able to identify it, i.e., not being able to identify parameter values, as illustrated by the effects model (with interaction) without constraints, discussed above. Again, we may consider parsimony to be the flip-side of flexibility; we like to explain things as simply as possible. We might simplify our model by omitting, e.g., the interactions, to arrive at a simpler (**additive**) model,

$$Y_{ijk} = \mu_{..} + \alpha_i + \beta_j + \epsilon_{ijk} \quad \epsilon_{ijk} \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$$

$$i = 1, \dots, a, \quad j = 1, \dots, b, \quad k = 1, \dots, n_{ij}.$$

All quantities are as described previously in the non-additive/saturated model. As we have discussed (Section 16.5.3), without interactions, we are left with a relatively simple interpretation in terms of main effects. (NOTE: In an attempt to avoid unnecessary confusion, our discussion of ANOVA components in Section 16.5.3 used the sum-to-zero interpretation of effects, to allow the somehow more intuitive interpretation of the overall effect as the overall mean, not the reference cell mean.)

- **Strategy for Analysis.** Same as given with the sum-to-zero effects model: As the above discussion may suggest, we often begin (if data permit) with a non-additive model, then test to see if we can reasonably omit interaction effects, which can simplify further analysis/interpretation.

This process is discussed in [RS13, Sec. 13.5.1] and [KNNL05, Sec. 19.7]. See, in particular, [KNNL05, Fig. 19.11 pg. 848]. In some sense, this process is no more a matter of performing F v R F-tests, as we will continue to illustrate below. Thus, we are already in good stead.

16.6.3 Effects Model: Trmt: Example: Initial Analysis

As mentioned before, treatment constraints are the default in R for (unordered) factors, but, we've been tinkering with such matters, so, first, we ensure treatment constraints before proceeding.

```
> ## Default constraint/coding is treatment, but may have changed:
> getOption("contrasts")

[1] "contr.sum" "contr.sum"

> ## Do factors have an overriding contrasts attribute?:
> sapply(case1301.df, attr, which="contrasts")

$Cover
NULL

$Block
 [,1] [,2] [,3] [,4] [,5] [,6] [,7]
B1    1    0    0    0    0    0    0
B2    0    1    0    0    0    0    0
B3    0    0    1    0    0    0    0
B4    0    0    0    1    0    0    0
B5    0    0    0    0    1    0    0
B6    0    0    0    0    0    1    0
B7    0    0    0    0    0    0    1
B8   -1   -1   -1   -1   -1   -1   -1

$Treat
 [,1] [,2] [,3] [,4] [,5]
C     1    0    0    0    0
L     0    1    0    0    0
```

```

Lf      0      0      1      0      0
LfF     0      0      0      1      0
f       0      0      0      0      1
ff     -1     -1     -1     -1     -1

$Blockord
NULL

> ## Again, we can set constraints/coding globally...
> options(contrasts = rep("contr.treatment",2))
>getOption("contrasts")
[1] "contr.treatment" "contr.treatment"

> ## ...or, we can assign (overriding) constraints/coding
> ## (perhaps not the same) to individual factors:
> contrasts(case1301.df$Block)<- contr.treatment(levels(case1301.df$Block))
> contrasts(case1301.df$Treat)<- contr.treatment(levels(case1301.df$Treat))
> sapply(case1301.df, attr, which="contrasts")

$Cover
NULL

$Block
  B2 B3 B4 B5 B6 B7 B8
B1  0  0  0  0  0  0  0
B2  1  0  0  0  0  0  0
B3  0  1  0  0  0  0  0
B4  0  0  1  0  0  0  0
B5  0  0  0  1  0  0  0
B6  0  0  0  0  1  0  0
B7  0  0  0  0  0  1  0
B8  0  0  0  0  0  0  1

$Treat
  L Lf LfF f ff
C   0  0    0 0  0
L   1  0    0 0  0
Lf  0  1    0 0  0
LfF 0  0    1 0  0
f   0  0    0 1  0
ff  0  0    0 0  1

$Blockord
NULL

```

- Now, let's repeat the fit of our first multi-way ANOVA model—using the logit transform—now using treatment constraints.
- We'll skip inspection of the (non-redundant) \mathbf{X} in R; we've looked at it, above, and have illustrated the `model.matrix` function before, so you should be able to reproduce it by hand or with R.
- Whether using sum-to-zero constraints or treatment constraints, nothing changes for the R model formula in our seaweed grazing example.

```
> case1301TC.lm<- lm(log(Cover/(100-Cover)) ~ Block + Treat +
+                         Block:Treat,
+                         data=case1301.df)
> ## Typical regression summary, often of subsidiary interest
> ## in an ANOVA context:
> summary(case1301TC.lm)
```

Call:

```
lm(formula = log(Cover/(100 - Cover)) ~ Block + Treat + Block:Treat,
   data = case1301.df)
```

Residuals:

Min	1Q	Median	3Q	Max
-0.843	-0.275	0.000	0.275	0.843

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-1.5118	0.3891	-3.89	0.00031 ***
BlockB2	0.5694	0.5503	1.03	0.30596
BlockB3	2.6241	0.5503	4.77	1.8e-05 ***
BlockB4	4.3598	0.5503	7.92	2.9e-10 ***
BlockB5	1.2402	0.5503	2.25	0.02882 *
BlockB6	2.2225	0.5503	4.04	0.00019 ***
BlockB7	0.7267	0.5503	1.32	0.19291
BlockB8	1.7955	0.5503	3.26	0.00204 **
TreatL	-1.6663	0.5503	-3.03	0.00396 **
TreatLf	-1.6985	0.5503	-3.09	0.00336 **

TreatLfF	-2.7317	0.5503	-4.96	9.1e-06	***
Treatf	-0.1099	0.5503	-0.20	0.84254	
TreatffF	-0.5373	0.5503	-0.98	0.33379	
BlockB2:TreatL	0.0941	0.7783	0.12	0.90428	
BlockB3:TreatL	0.2431	0.7783	0.31	0.75607	
BlockB4:TreatL	-1.9886	0.7783	-2.56	0.01384	*
BlockB5:TreatL	0.5384	0.7783	0.69	0.49242	
BlockB6:TreatL	-0.2736	0.7783	-0.35	0.72670	
BlockB7:TreatL	-0.1456	0.7783	-0.19	0.85237	
BlockB8:TreatL	-0.2776	0.7783	-0.36	0.72289	
BlockB2:TreatLf	-0.4730	0.7783	-0.61	0.54621	
BlockB3:TreatLf	-0.9707	0.7783	-1.25	0.21836	
BlockB4:TreatLf	-1.6711	0.7783	-2.15	0.03686	*
BlockB5:TreatLf	-0.6590	0.7783	-0.85	0.40133	
BlockB6:TreatLf	0.3239	0.7783	0.42	0.67916	
BlockB7:TreatLf	-0.1017	0.7783	-0.13	0.89658	
BlockB8:TreatLf	-0.3396	0.7783	-0.44	0.66449	
BlockB2:TreatLfF	0.4638	0.7783	0.60	0.55405	
BlockB3:TreatLfF	-0.9132	0.7783	-1.17	0.24644	
BlockB4:TreatLfF	-2.0425	0.7783	-2.62	0.01160	*
BlockB5:TreatLfF	0.1553	0.7783	0.20	0.84272	
BlockB6:TreatLfF	0.1296	0.7783	0.17	0.86849	
BlockB7:TreatLfF	1.1369	0.7783	1.46	0.15058	
BlockB8:TreatLfF	-0.3176	0.7783	-0.41	0.68500	
BlockB2:Treatf	-0.2554	0.7783	-0.33	0.74417	
BlockB3:Treatf	-0.7804	0.7783	-1.00	0.32104	
BlockB4:Treatf	-0.8999	0.7783	-1.16	0.25328	
BlockB5:Treatf	-0.3043	0.7783	-0.39	0.69757	
BlockB6:Treatf	-0.7844	0.7783	-1.01	0.31855	
BlockB7:Treatf	0.8141	0.7783	1.05	0.30076	
BlockB8:Treatf	-0.8636	0.7783	-1.11	0.27269	
BlockB2:TreatffF	-0.4863	0.7783	-0.62	0.53505	
BlockB3:TreatffF	-0.6956	0.7783	-0.89	0.37592	
BlockB4:TreatffF	-1.6725	0.7783	-2.15	0.03671	*
BlockB5:TreatffF	0.1245	0.7783	0.16	0.87358	
BlockB6:TreatffF	-0.5796	0.7783	-0.74	0.46009	
BlockB7:TreatffF	0.5870	0.7783	0.75	0.45439	
BlockB8:TreatffF	-0.9945	0.7783	-1.28	0.20743	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.55 on 48 degrees of freedom

```
Multiple R-squared:  0.928, Adjusted R-squared:  0.858
F-statistic: 13.2 on 47 and 48 DF,  p-value: 7.54e-16
```

```
> ## Typical ANOVA table, often receives primary interest
> ## in an ANOVA context. Note it's same as in stz case.
> anova(case1301TC.lm)
```

Analysis of Variance Table

```
Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Block       7   76.2   10.89   35.96 <2e-16 ***
Treat       5   97.0   19.40   64.06 <2e-16 ***
Block:Treat 35   15.2    0.44    1.44   0.12
Residuals   48   14.5    0.30
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notice above that, aside from the parameter estimates and associated default tests, essentially all of the remaining output, above, is identical to that given for our sum-to-zero effects analysis: again, the cell means model and the saturated, non-additive effects models—either coding—are equivalent models, up to parameter interpretation, i.e., up to reparameterization.

16.6.4 Effects Model: Trmt: Example: ANOVA For Common C β

We do not repeat the discussion of the ANOVA table that we gave in the context of the sum-to-zero analysis, above. The table is the same (compare `anova` output of the current treatment constraint analysis, above, to that of the sum-to-zero analysis, given previously, or to [RS13, Display 13.10]). Still, as in the analogous section, above, using sum-to-zero constraints, we warn that ANOVA tables may be ambiguous about what is being tested, in either the **BALANCED** or **UNBALANCED** cases; see Section 16.7, below, for more about this.

16.6.5 Effects Model: Trmt: Example: F v R & C β Approach

As we did for the sum-to-zero case, above, we illustrate the F v R (aka extra sums of squares) and C β approaches. Of course, results are the same: no interaction. Incidentally, the linear combinations implementation appears to be same as in the sum-to-zero case, above, despite the different parameter interpretations. Why? More discussion in class.

First: F v R code. Once again, now with treatment constraints, notice the reduced model ANOVA table given by `anova` shows that the sum-of-squares for the remaining terms, Block and Treat, are the same as those given by the full model. Also, compare [RS13, Displays 13.10 & 13.11]. Again, that the sum-of-squares, etc., do not change between F and R models generally only happens in **BALANCED** cases. Do you see more clearly now a potential ambiguity of tests as presented in ANOVA tables? More about this in Section 16.7, below.

```
> ## Next is essentially identical to the sum-to-zero analysis above:
> ##
> ## Reduced model, without interaction term:
> case1301TCR.lm<- update(case1301TC.lm,. ~ . - Block:Treat)
> ## Just to illustrate effect of balance, otherwise...
> anova(case1301TCR.lm)
```

Analysis of Variance Table

```
Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Block      7   76.2   10.89   30.4 <2e-16 ***
Treat      5   97.0   19.40   54.1 <2e-16 ***
Residuals 83   29.8    0.36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ##...just go to F v R test:
> anova(case1301TCR.lm, case1301TC.lm)
```

Analysis of Variance Table

```

Model 1: log(Cover/(100 - Cover)) ~ Block + Treat
Model 2: log(Cover/(100 - Cover)) ~ Block + Treat + Block:Treat
  Res.Df RSS Df Sum of Sq    F Pr(>F)
1      83 29.8
2      48 14.5 35      15.2 1.44   0.12

> ## which gives the same test for interaction as given
> ## by anova(case1301TC.lm), as promised.

```

Next, we consider the **linear combinations approach**. Which elements of β are associated with the interaction, i.e., with the $(\alpha\beta)_{ij}$, i.e., with the Treat:Block model formula term? Why is the implementation the same as in the sum-to-zero analysis?

```

> ## I build C matrix in stages (not necessary).
> ##
> a<- 8; b<- 6 ## number of factor levels
> Ca<- matrix(0,nrow=(a-1)*(b-1), ncol=a-1)
> Cb<- matrix(0,nrow=(a-1)*(b-1), ncol=b-1)
> Cab<- diag((a-1)*(b-1))
> Cmat<- cbind(0,Ca,Cb,Cab) ## C matrix
> d<- rep(0,(a-1)*(b-1)) ## null CBeta value
> glh.test(reg=case1301TC.lm, cm=Cmat, d=d)

Test of General Linear Hypothesis
Call:
glh.test(reg = case1301TC.lm, cm = Cmat, d = d)
F = 1.4369, df1 = 35, df2 = 48, p-value = 0.1209

> ## Result is same as F v R approach or anova(case1301TC.lm)
> ## of course!

```

16.6.6 Effects Model: Trmt: Example: Summary

No matter how we approached the analysis—sum-to-zero or treatment constraints; ANOVA table vs. F v R vs. linear combinations—there is little evidence for serious, significant interaction. Before moving on with inferences about the main effects of grazing in the additive model—see **Strategy for Analysis** bullets above—we discuss some potential ambiguity that may arise when using standard ANOVA tables.

16.7 SS Type, Balance & the Marginality Principle

16.7.1 Sequential (Type I) SS ANOVA

- **Always The (relatively uninteresting) Overall Decomposition.** As we mentioned, above, when discussing the factor effects model with sum-to-zero constraints (§16.5.6), we always have

$$\text{TSS} = \text{SS}_{\text{reg}} + \text{RSS},$$

i.e., TSS always retains an additive decomposition into SS_{reg} (“among groups” SS or “regression” SS) and RSS (residual SS).

- There is no issue with this ‘overall’ decomposition. On the other hand, there is not much at risk, either: the overall F-test, which is often not of primary interest. (As we know, R typically does report an overall F-test, but it does not usually report TSS, but some ANOVA tables may show it.)
- **After Previous but Before Subsequent.** For **type I SS** (sequential) ANOVA tables, the overall SS_{reg} , however, is further decomposed into SS values that are associated with model formula terms, after all **previous** (but not subsequent) terms in the table are in the model.
- **Adjustment.** In other words, these SS values are adjusted for previous terms but not for subsequent terms. (I use ‘adjustment’ in the same

sense as ‘covariate adjustment’ in [Far14, Chap. 5] (INF 511); to be sure, covariate adjustment adjusts for all other terms in the model, not just previous terms, so, generally speaking, sequential (type I) ANOVA tables may deviate from our notion of covariate adjustment.)

- Thus, if FactorA is reported first then Factor B, SS values may be different than when the order is reversed in the table.
- That is, with suggestive (“extra SS”) notation, we might have an ANOVA table reporting SS as

$$\begin{aligned}SS(A) \\SS(B|A) \\SS(AB|A, B)\end{aligned}$$

or as

$$\begin{aligned}SS(B) \\SS(A|B) \\SS(BA|B, A)\end{aligned}$$

so that

$$\begin{aligned}SS_{reg} &= SS(A) + SS(B|A) + SS(AB|A, B) \\&= SS(B) + SS(A|B) + SS(BA|B, A),\end{aligned}$$

but, generally,

$$\begin{aligned}SS(A) &\neq SS(A|B) \quad \text{and} \\SS(B) &\neq SS(B|A)\end{aligned}$$

- **Last is the Same.** The last SS components are the same if they are for the same term. For example, the interaction term enters last in both cases, above, so that $SS(BA|B, A) = SS(AB|A, B)$.

- **No Difference with Balanced Designs.** In the **balanced** case, there is no such dependence of SS values on the term order in an ANOVA table (sequential (type I) or otherwise). That's why we didn't have to talk about this dependence on order of appearance in the table (type I or otherwise) in our running example of the *balanced* intertidal seaweed grazing experiment (§16.5.6). (Still, there seems to be an ambiguity about what test is being done: after all others are in the model? after previous before others? what?...)
- **Unbalanced Illustration.** To illustrate the different decompositions and the ensuing ambiguity about which SS/test to use, I create an **unbalanced version of the data set** from our running example for seaweed grazers.

```
> ## Balanced: nij = n = 2
> with(case1301.df, table(interaction(Block, Treat)))
```

B1.C	B2.C	B3.C	B4.C	B5.C	B6.C	B7.C	B8.C	B1.L	B2.L
2	2	2	2	2	2	2	2	2	2
B3.L	B4.L	B5.L	B6.L	B7.L	B8.L	B1.Lf	B2.Lf	B3.Lf	B4.Lf
2	2	2	2	2	2	2	2	2	2
B5.Lf	B6.Lf	B7.Lf	B8.Lf	B1.LfF	B2.LfF	B3.LfF	B4.LfF	B5.LfF	B6.LfF
2	2	2	2	2	2	2	2	2	2
B7.LfF	B8.LfF	B1.f	B2.f	B3.f	B4.f	B5.f	B6.f	B7.f	B8.f
2	2	2	2	2	2	2	2	2	2
B1.fF	B2.fF	B3.fF	B4.fF	B5.fF	B6.fF	B7.fF	B8.fF		
2	2	2	2	2	2	2	2		

```
> ## To illustrate imbalance, randomly omit 1 observation from each
> ## of half of the treatment levels to create imbalance
> ## (a*b = 8*6 = 48 treatments: randomly choose 24 treatments from
> ## which to omit 1 of the 2 cell observations but not both)
> set.seed(24601)
> indx<- sample(seq(1,95,2), size=24) + sample(0:1,size=24, repl=TRUE)
> dim(unbal.df<- case1301.df[-indx,])
```

```
[1] 72 4
```

```
> with(unbal.df, table(interaction(Block, Treat)))
```

B1.C	B2.C	B3.C	B4.C	B5.C	B6.C	B7.C	B8.C	B1.L	B2.L
2	2	1	2	1	2	1	2	2	1
B3.L	B4.L	B5.L	B6.L	B7.L	B8.L	B1.Lf	B2.Lf	B3.Lf	B4.Lf
2	1	1	1	2	2	1	2	2	2
B5.Lf	B6.Lf	B7.Lf	B8.Lf	B1.LfF	B2.LfF	B3.LfF	B4.LfF	B5.LfF	B6.LfF
1	1	2	1	1	2	1	2	1	1
B7.LfF	B8.LfF	B1.f	B2.f	B3.f	B4.f	B5.f	B6.f	B7.f	B8.f
2	2	2	2	1	1	1	2	2	2
B1.ff	B2.ff	B3.ff	B4.ff	B5.ff	B6.ff	B7.ff	B8.ff		
1	1	2	2	1	1	1	1		

- Now, for example, I can specify the same model in R as

$$R \sim A + B + A:B,$$

implying $SS_{reg} = SS(A) + SS(B|A) + SS(AB|A, B)$, or as

$$R \sim B + A + B:A,$$

implying $SS_{reg} = SS(B) + SS(A|B) + SS(BA|B, A)$.

- The differences (see code/output below) between $SS(A)$ and $SS(A|B)$ or between $SS(B)$ and $SS(B|A)$ are not dramatic in this particular case, but you can see the change; and, as mentioned, $SS(AB|A, B) = SS(BA|B, A)$, as these enter the type I sequential table last in each case. (The different specifications result in different permutations of the β vector elements, but this is not relevant in the current context as we are not explicitly considering “picking-off” parameters with C matrices; again, ANOVA tables can be convenient...)

```
> ##
> ## Unbalanced: Block ('`Factor A'') first, then Treat ('`B''):
> ##
> anova(unbal1.lm<- lm(log(Cover/(100-Cover)) ~
+                      Block + Treat + Block:Treat,
+                      data=unbal.df))
```

Analysis of Variance Table

```
Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Block      7   54.3    7.76   22.56 4.2e-09 ***
Treat      5   79.9   15.98   46.47 1.4e-11 ***
Block:Treat 35   11.9    0.34    0.99    0.52
Residuals  24    8.3    0.34
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ##
> ## Unbalanced: Treat ('`Factor B'') first, then Block ('`A''):
> ##
> anova(unbal2.lm<- lm(log(Cover/(100-Cover)) ~
+                      Treat + Block + Treat:Block,
+                      data=unbal.df))
```

Analysis of Variance Table

```
Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Treat      5   77.3   15.47   44.99 2.0e-11 ***
Block      7   56.9    8.12   23.62 2.7e-09 ***
Treat:Block 35   11.9    0.34    0.99    0.52
Residuals  24    8.3    0.34
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- So, the question is, **which test(s) do we use?**

16.7.2 Partial (Type III) SS ANOVA

- Perhaps in an attempt to remove any ambiguity about which decomposition to use for tests, **type III SS (partial)** are the SS values that result for (main effect or interaction) terms **after all other** terms are in the model.
- **Partial SS and Tests.** Generally speaking, unlike the sequential case, these SS values are *not* a decomposition of SS_{reg} (except in balanced cases), but these SS values and associated tests are **what we want** (and what we have been doing in regression and ANOVA all along): test for the significance of a term(s) after all others are in the model: partial F tests (or partial t tests).
- **Covariate Adjustment.** That is, we now ‘adjust’ our inference of a particular term in the model formula after all other terms have been accounted for, with, again, ‘adjustment’ used in the same sense as in [Far14, Chap. 5].
- **Marginality Principle.** But, likely, we do not want to violate the **marginality** principle. More below.
- **No Order Ambiguity.** So, no matter what order the terms/SS occur in a type III (partial) SS ANOVA table, the SS and tests are (numerically) the same for each term; we might write suggestively, e.g. (compare to sequential notation, above),

$$\begin{aligned} SS(A|B, AB) \\ SS(B|A, AB) \\ SS(AB|A, B) \end{aligned}$$

- We use the `Anova` function (capital A) the the `car` library to illustrate. As mentioned, the type III ANOVAs are the same, regardless of the order of terms in the model formula (see previous code), e.g.,

```
> library(car)

Loading required package: carData

> Anova(unbal1.lm, type="III")

Anova Table (Type III tests)

Response: log(Cover/(100 - Cover))
            Sum Sq Df F value    Pr(>F)
(Intercept) 4.57   1 13.29    0.0013 ***
Block        24.14  7 10.03 0.0000081 ***
Treat        7.39   5  4.30    0.0062 **
Block:Treat 11.87  35  0.99    0.5235
Residuals    8.25  24
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> Anova(unbal2.lm, type="III")

Anova Table (Type III tests)

Response: log(Cover/(100 - Cover))
            Sum Sq Df F value    Pr(>F)
(Intercept) 4.57   1 13.29    0.0013 ***
Treat        7.39   5  4.30    0.0062 **
Block        24.14  7 10.03 0.0000081 ***
Treat:Block 11.87  35  0.99    0.5235
Residuals    8.25  24
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Again, $SS(AB|A, B)$ (or $SS(BA|B, A)$) is the same in either the type I or type III tables because the interaction term enters last.
- **Temptation?** But now, the type III table somehow encourages us (well, perhaps a novice) to test a main effect before testing the higher order interaction that contains the main effect term. (Granted, a novice might

also mistakenly use a sequential table to test mains before interactions.)
A bit more in class.

16.7.3 Marginality Principle (aka Hierarchy Principle)

- **As in Regression.** We considered the same issue with the typical regression summary (from the `summary` function) that reports partial regression coefficients and partial t-tests: we warned that we almost always want to obey the **marginality principle: briefly, don't test lower order terms that exist in higher order terms that are still in the model.**
- **R Tries to Help?** The fact that R's "default" `anova` function presents only type I (sequential) SS *may* be seen as an (deliberate?) attempt to discourage violation of the marginality principle: we would somehow have to get `anova` to put the desired term to be tested as the last entry in the table (which is the only entry that presents a type III (partial) F/t test in a sequential table as given by `anova`.)
- **An Attempt to Violate Marginality.** For illustration, in an attempt to violate the marginality principle by testing main effects before interaction effects (and, we pretend to be unaware of cars `Anova` function), we might try something like

$$R \sim A:B + A + B ,$$

or

$$R \sim B:A + B + A .$$

- **Trying to Help (?) R Refuses.** Notice, however, in the output, below, while `anova` does change the table entry order of the similar (main effects) terms, A and B, it does not allow the higher order term to come before the lower order terms!

- Thus, `anova` (not `Anova`) seems to discourage attempts to test lower order terms before higher order terms that contain them.
- **F v R Okay.** To be sure, we can still use `anova` with two or more model fits to implement our F v R model approach without issue. No problems. (Not illustrated here.)

```
> ##
> ## Try to violate marginality principle...nope
> anova(unbal1.lm<- lm(log(Cover/(100-Cover)) ~
+           Block:Treat + Block + Treat,
+           data=unbal.df))

Analysis of Variance Table

Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Block      7   54.3    7.76   22.56 4.2e-09 ***
Treat      5   79.9   15.98   46.47 1.4e-11 ***
Block:Treat 35   11.9    0.34     0.99    0.52
Residuals  24    8.3    0.34
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ##
> ## Try again...nope
> anova(unbal2.lm<- lm(log(Cover/(100-Cover)) ~
+           Treat:Block + Treat + Block,
+           data=unbal.df))

Analysis of Variance Table

Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Treat      5   77.3   15.47   44.99 2.0e-11 ***
Block      7   56.9    8.12   23.62 2.7e-09 ***
Treat:Block 35   11.9    0.34     0.99    0.52
Residuals  24    8.3    0.34
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **drop1.** R also has the drop1 function, which gives partial F tests (when asked) without, evidently, violating the marginality principle.
- Notice, in the output, below, we get tests that result by “dropping” each term, individually, while leaving all other terms in the model, only if the test does not violate marginality; there are no main effects tests reported because there is an interaction term in the model.

```
> ## no Block or Treat test: respects marginality principle
> drop1(unbal1.lm, test="F")
```

Single term deletions

Model:

```
log(Cover/(100 - Cover)) ~ Block:Treat + Block + Treat
      Df Sum of Sq   RSS   AIC F value Pr(>F)
<none>           8.25 -60.0
Block:Treat 35     11.9 20.13 -65.8    0.99   0.52
```

```
> ## no Block or Treat test: respects marginality principle
> drop1(unbal2.lm, test="F")
```

Single term deletions

Model:

```
log(Cover/(100 - Cover)) ~ Treat:Block + Treat + Block
      Df Sum of Sq   RSS   AIC F value Pr(>F)
<none>           8.25 -60.0
Treat:Block 35     11.9 20.13 -65.8    0.99   0.52
```

- **You Are in Command.** Still, you can get around R’s default behavior and force drop1 to ignore (higher order) terms by using the scope option to risk violating the marginality principle.

- Notice the tests for main effects below are the same as given in the type III (partial) ANOVAs, above, given by car's Anova function, which does not seem to fight against violation of the marginality principle.
- To be sure, we are violating the marginality principle here by testing main effects with a relevant higher order interaction in the model. (We have an illustration of a particularly bad violation in §16.5.3.)

```
> drop1(unbal1.lm, scope= ~ Block + Treat, test="F")

Single term deletions

Model:
log(Cover/(100 - Cover)) ~ Block:Treat + Block + Treat
      Df Sum of Sq   RSS   AIC F value    Pr(>F)
<none>           8.3 -60.0
Block    7     24.14 32.4  24.5     10.0 0.0000081 ***
Treat    5      7.39 15.6 -23.9      4.3   0.0062 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> drop1(unbal2.lm, scope= ~ Treat + Block, test="F")

Single term deletions

Model:
log(Cover/(100 - Cover)) ~ Treat:Block + Treat + Block
      Df Sum of Sq   RSS   AIC F value    Pr(>F)
<none>           8.3 -60.0
Treat    5      7.39 15.6 -23.9      4.3   0.0062 **
Block    7     24.14 32.4  24.5     10.0 0.0000081 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

16.7.4 Summary & Remarks

- Be mindful of the marginality principle, especially in particularly bad interaction cases as illustrated in §16.5.3.
- Though we may not have discussed it much, the marginality principle is applicable in regression, too: if, e.g., xy is in the model then x and y (and 1) should be in the model; similarly for x^2 and x (and 1).
- See [KNNL05, pg. 299] for their “Hierarchical Approach to Fitting,” which is essentially a description of the marginality principle in practice for regression models.
- Use Anova (car package) for the partial tests that we want.
- Or, use anova (stats package) if we have balance.
- Or, use anova with two or more models for the typical F v R approach. Okay, balanced or not.
- Or, use the $\mathbf{C}\boldsymbol{\beta}$ approach. Okay, balanced or not. (A bit messy; not common.)
- Recall previous **Strategy for Analysis** bullets: try to simplify your model by omitting (testing for) interaction terms (without violating the marginality principle) (if data permit).

16.8 Additive Model: Tests for Overall Main Effects

- **Additive Model.** After investigating interaction and deciding non-significance, a typical course of action is to proceed with the additive model to make further inference about main effects, e.g., grazing effects or block (location) effects in our running example.

- **Main Effects.** In short, both the Treat and Block (location) terms are significant, as reported in any (transformed response) ANOVA table so far in this example (except the deliberately imbalanced illustrations). (Why?)
- **Our Usual Approaches.** Still, in keeping with our pattern of presentation, in addition to the ANOVA tables, we can, of course, use the F v R or linear combinations approach. Now, however, our “F”ull model is now the additive model, without interaction effects.

16.8.1 F v R Approach

First, we present the F v R approach to testing for significant block (location) effects.

```
> #####
> ## Sum-to-zero analysis:
> #####
> ## New reduced model (case1301aR.lm omits interaction):
> case1301aR2.lm<- update(case1301aR.lm, . ~ . - Block)
> ## F v R test:
> anova(case1301aR2.lm, case1301aR.lm)
```

Analysis of Variance Table

```
Model 1: log(Cover/(100 - Cover)) ~ Treat
Model 2: log(Cover/(100 - Cover)) ~ Block + Treat
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1      90 106.0
2      83  29.8  7     76.2 30.4 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> ## Or, why not this (note use of fullest model MSE)?:
> anova(case1301aR2.lm, case1301aR.lm, case1301a.lm)
```

Analysis of Variance Table

```

Model 1: log(Cover/(100 - Cover)) ~ Treat
Model 2: log(Cover/(100 - Cover)) ~ Block + Treat
Model 3: log(Cover/(100 - Cover)) ~ Block + Treat + Block:Treat
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     90 106.0
2     83 29.8  7      76.2 35.96 <2e-16 ***
3     48 14.5 35      15.2  1.44   0.12
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

> ## BTW, just to illustrate the non-effect of balance
 > ## on SS, we see Treat SS and test is unchanged:
 > **anova**(case1301aR2.lm)

Analysis of Variance Table

```

Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Treat      5     97   19.40   16.5 1.6e-11 ***
Residuals 90    106    1.18
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

> #####
 > ## Treatment analysis:
 > #####
 > ## New reduced model:
 > case1301TCR2.lm<- **update**(case1301TCR.lm, . ~ . - Block)
 > ## F v R test:
 > **anova**(case1301TCR2.lm, case1301TCR.lm)

Analysis of Variance Table

```

Model 1: log(Cover/(100 - Cover)) ~ Treat
Model 2: log(Cover/(100 - Cover)) ~ Block + Treat
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     90 106.0
2     83 29.8  7      76.2 30.4 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
> ## or, why not this (note use of fullest model MSE)?:
> anova(case1301TCR2.lm, case1301TCR.lm, case1301TC.lm)
```

Analysis of Variance Table

```
Model 1: log(Cover/(100 - Cover)) ~ Treat
Model 2: log(Cover/(100 - Cover)) ~ Block + Treat
Model 3: log(Cover/(100 - Cover)) ~ Block + Treat + Block:Treat
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     90 106.0
2     83  29.8  7      76.2 35.96 <2e-16 ***
3     48  14.5 35      15.2  1.44    0.12
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> ## BTW, just to illustrate the non-effect of balance
> ## on SS, we see Treat SS and test is unchanged:
> anova(case1301TCR2.lm)
```

Analysis of Variance Table

```
Response: log(Cover/(100 - Cover))
          Df Sum Sq Mean Sq F value Pr(>F)
Treat      5     97   19.40   16.5 1.6e-11 ***
Residuals 90    106    1.18
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As we already knew, blocks appear to account for significant response variability, something that the seaweed grazing researchers (!) suspected and were, presumably, not surprised by. We keep blocks in our additive model. See [RS13, pg 397].

16.8.2 C β Approach

Next, we consider the linear combinations approach. Despite the different parameter interpretation between the sum-to-zero analysis and the treatment

analysis, the implementation appears to be the same in either case. (Recall that implementations for testing interaction effects also appeared to be the same between these analyses.) Why?

```
> ## Linear combinations approach:
> ##
> a<- 8; b<- 6 ## number of factor levels
> Ca<- diag(a-1)
> Cb<- matrix(0, nrow=(a-1), ncol=b-1)
> Cmat<- cbind(0,Ca,Cb) ## C matrix
> d<- rep(0,(a-1)) ## null CBeta value
> ##
> ## Sum-to-zero analysis:
> ##
> glh.test(reg=case1301aR.lm, cm=Cmat, d=d)
```

```
Test of General Linear Hypothesis
Call:
glh.test(reg = case1301aR.lm, cm = Cmat, d = d)
F = 30.368, df1 = 7, df2 = 83, p-value = < 2.2e-16
```

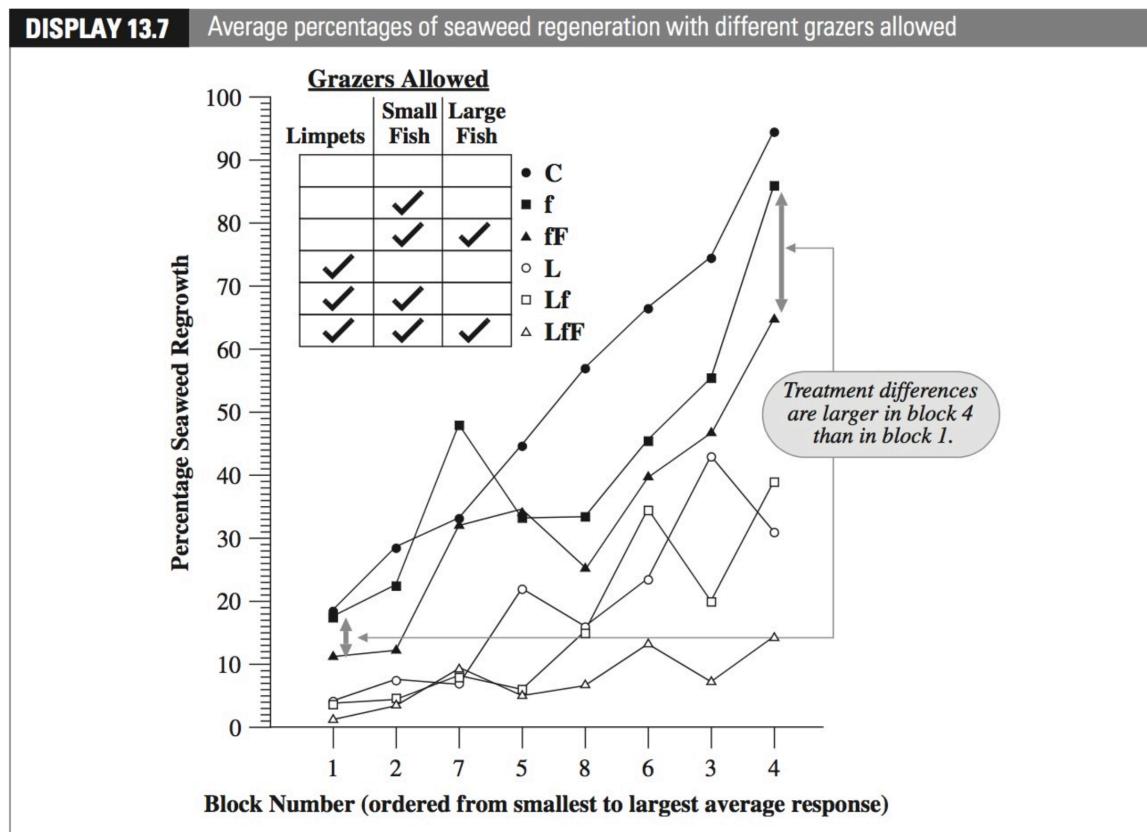
```
> ##
> ## Treatment analysis:
> ##
> glh.test(reg=case1301TCR.lm, cm=Cmat, d=d)
```

```
Test of General Linear Hypothesis
Call:
glh.test(reg = case1301TCR.lm, cm = Cmat, d = d)
F = 30.368, df1 = 7, df2 = 83, p-value = < 2.2e-16
```

Now, we move on to **more specific questions about grazing effects** using the additive model with Block and Treat terms, as indicated by our analysis so far.

16.9 Additive Model: More Detailed Inference of Main Effects

- **Marginality Again.** Again, as with the tests of main effects (Block or Treat), in the previous section, even more detailed inferences about main effects generally only make sense in the absence of (bad) **interaction**, which we dropped from our model, of course.
- **No Interaction: Relatively Straightforward.** We will see that, without interaction, the (additive) effects model (either set of constraints) make for relatively straightforward investigation of detailed questions about main effects: such questions typically involve, wonder of wonders, only the main effects parameters of the effect in question.
- Again, we mentioned that we do not typically have interest in making further, more detailed inference about block effects, but we typically would want to investigate more detailed inferences about grazing effects, the “treatment” (factor!) of interest.
- We follow [RS13, Sections 13.3.4 & 13.3.5] with the R code to implement (some of) the first two of their inferences, numbered 1-5 in [RS13, Section 13.3.4]. We will have some discussion in class.
- You may be asked about the remaining inferences on a homework. See [RS13, Sections 13.3.4 & 13.3.5] for more.
- To help us keep track of factor levels, let’s refer to the “Grazers Allowed” table in [RS13, Display 13.7], which we reproduced, above, and reproduce again, nearby, for convenience.
- This table is merely helping us to interpret the levels of our factor B, the levels of Treat, the grazing “treatment” (technically, factor) to enable further analysis among these levels.
- In particular, we will re-order our factor B (Treat) levels to match the order shown in that table; see code below.



- (1) Do large fish have an effect on the regeneration ratio? If so, how much?
- First, we express a linear combination using notation suggested by the table. We will convert it to our general effects notation shortly. The following linear combination (specifically, contrast) of (marginal factor) means embodies our question (γ_1 notation is just the [RS13] denote their first linear combination of interest):

$$\gamma_1 = \frac{1}{2}(\mu_{fF} - \mu_f) + \frac{1}{2}(\mu_{LfF} - \mu_{Lf}).$$

- Note how this linear combination is reasonable: the question and table suggest two effects of large fish to be considered: (1) an effect of large fish (F) in the presence of small fish (f), limpet (L) absent; and (2) the

effect large fish (F) in the presence of small fish (f), limpets (L) present. Because the question does not distinguish (1) or (2), it seems reasonable to average these effects to arrive at a linear combination of parameters that gets at our question. (Again, we'll convert to our effects notation, shortly.) See the summary of linear combination coefficients in [RS13, Display 13.13] (not shown here).

- We will order the levels of the Treat factor to match the order of the rows in the table in [RS13, Display 13.7] (reproduced nearby), first row = 1, etc. Thus, we translate the above linear combination to our marginal mean notation (see Factor level (marginal) means bullet in §16.1, above),

$$\gamma_1 = \frac{1}{2}(\mu_{.3} - \mu_{.2}) + \frac{1}{2}(\mu_{.6} - \mu_{.5}),$$

which is more explicit about the fact that we are averaging over blocks (and is 'our' notation).

- While using (marginal) means to specify detailed linear combinations may be convenient, especially when interactions exist (not here), we translate the γ_1 linear combination to our effects model(s) notation. In either the sum-to-zero or treatment constraints cases, we have

$$\beta_j = \mu_{.j} - \mu_{..},$$

- This allows us to write the above linear combination in terms of the treatment effects, β_j :

$$\gamma_1 = \frac{1}{2}(\beta_3 - \beta_2) + \frac{1}{2}(\beta_6 - \beta_5).$$

- Or, using the suggestive subscripts again, we may write,

$$\gamma_1 = \frac{1}{2}(\beta_{fF} - \beta_f) + \frac{1}{2}(\beta_{LfF} - \beta_{Lf}).$$

- To proceed further, we must consider which constraint(s) we are using, because the interpretation of the β_j and the (non-redundant) β vector depend on the constraints used, as we know.
- First, let's consider the treatment constraints/coding, which is the default in R. It turns out that this is the more convenient coding, in some sense, for question 1. Why? More in class (maybe).
- For the treatment constraint (in R, at least), recall $\beta_1 = 0$ so that β contains (among others) all β_j , β_2 to β_b , but not β_1 . Notice that this requires no further changes to γ_1 , i.e., we have γ_1 in terms of the elements of the treatment constraint's β vector:

$$\gamma_1 = \frac{1}{2}(\beta_3 - \beta_2) + \frac{1}{2}(\beta_6 - \beta_5) \quad \text{treatment.}$$

- For the sum-to-zero constraint, recall that β contains β_1 to β_{b-1} (in R, at least), because we can recover $\beta_b = -\sum_{j=1}^{b-1} \beta_j$.
- Plugging in this constraint definition of β_6 ($b = 6$) in the above expression for γ_1 gives

$$-\frac{1}{2}\beta_1 - \beta_2 - \frac{1}{2}\beta_4 - \beta_5 \quad \text{sum-to-zero.}$$

```
> ## Are we sure we know how R orders factor levels? help(factor) WARNS:
> ## The levels of a factor are by default sorted, but the sort order
> ## may well depend on the locale at the time of creation, and should
> ## not be assumed to be ASCII.
> levels(case1301.df$Treat)
[1] "C"     "L"     "Lf"    "LfF"   "f"     "fF"
> ## Let's reorder/refit to match Display 13.7 in R&S (nearby)
> ## to help avoid confusion:
> case1301.df$Treat<- factor(case1301.df$Treat,
+                                 levels=levels(case1301.df$Treat)[c(1,5,6,2,3,4)])
> levels(case1301.df$Treat) ## now matches Display 13.7 (nearby)
```

```
[1] "C"     "f"     "fF"    "L"     "Lf"    "LfF"

> ## Are we sure which constraint(s) is/are in effect?:
> sapply(case1301.df, attr, which="contrasts")

$Cover
NULL

$Block
   B2 B3 B4 B5 B6 B7 B8
B1  0  0  0  0  0  0  0
B2  1  0  0  0  0  0  0
B3  0  1  0  0  0  0  0
B4  0  0  1  0  0  0  0
B5  0  0  0  1  0  0  0
B6  0  0  0  0  1  0  0
B7  0  0  0  0  0  1  0
B8  0  0  0  0  0  0  1

$Treat
NULL

$Blockord
NULL

>getOption("contrasts") ## ...okay to proceed?

[1] "contr.treatment" "contr.treatment"

> ##
> #####
> ## Treatment analysis:
> #####
> ## Overwrites previous fit of same name (levels okay now?):
> case1301TCR.lm<- lm(log(Cover/(100-Cover)) ~ Block + Treat,
+                         data=case1301.df)
> a<- 8; b<- 6 ## number of factor levels
> Ca<- rep(0,a-1) ## blocks
> Cb<- 1/2 * c(-1, 1, 0, -1, 1) ## grazing
> Cmat<- c(0,Ca,Cb) ## C matrix (vector)
> d<- 0 ## null CBeta value
> glh.test(reg=case1301TCR.lm, cm=Cmat, d=d)
```

```

Test of General Linear Hypothesis
Call:
glh.test(reg = case1301TCR.lm, cm = Cmat, d = d)
F = 16.82, df1 = 1, df2 = 83, p-value = 0.0000954

> ## Or, somewhat ugly, but readable and perhaps instructive
> ## (compare to ``Contrast Summary'' in Display 13.13 in R&S):
> ##
> estimable(obj=case1301TCR.lm, cm=Cmat, beta0=d, conf.int=0.95)

          beta0 Estimate Std. Error
(0 0 0 0 0 0 0 -0.5 0.5 0 -0.5 0.5)    0 -0.61403   0.14972
                                         t value DF  Pr(>|t|) Lower.CI
(0 0 0 0 0 0 0 -0.5 0.5 0 -0.5 0.5) -4.1013 83 0.000095399 -0.9118
                                         Upper.CI
(0 0 0 0 0 0 0 -0.5 0.5 0 -0.5 0.5) -0.31625

> ## (Alas, R&S reorder the ``Treatment'' (factor B) levels in
> ## their Display 13.13 of results! Our results match theirs
> ## of course, despite the difference in order of presentation.)
> ##
> #####
> ## Sum-to-zero analysis:
> #####
> contrasts(case1301.df$Block)<- contr.sum(levels(case1301.df$Block))
> contrasts(case1301.df$Treat)<- contr.sum(levels(case1301.df$Treat))
> ## Overwrites previous fit of same name (levels okay now?):
> case1301aR.lm<- lm(log(Cover/(100-Cover)) ~ Block + Treat,
+                         data=case1301.df)
> Cb<- -c(1/2, 1, 0, 1/2, 1) ## grazing
> Cmat<- c(0,Ca,Cb) ## C matrix (vector)
> glh.test(reg=case1301aR.lm, cm=Cmat, d=d)

```

```

Test of General Linear Hypothesis
Call:
glh.test(reg = case1301aR.lm, cm = Cmat, d = d)
F = 16.82, df1 = 1, df2 = 83, p-value = 0.0000954

> estimable(obj=case1301aR.lm, cm=Cmat, beta0=d, conf.int=0.95)

          beta0 Estimate Std. Error t value
(0 0 0 0 0 0 0 -0.5 -1 0 -0.5 -1)    0 -0.61403   0.14972 -4.1013
                                         DF  Pr(>|t|) Lower.CI Upper.CI
(0 0 0 0 0 0 0 -0.5 -1 0 -0.5 -1) 83 0.000095399 -0.9118 -0.31625

```

> ## Results are the same for either coding, of course.

- **(4) Do limpets have a different effect when small fish are present than when small fish are absent?** (This is question 4 in [RS13, Chap. 13].)
- If we compare the question to the table in [RS13, Display 13.7] (above), we can see two effects of limpets in the presence of small fish (following R&S suggestive subscripts for the moment, as before):

$$\mu_{LfF} - \mu_{fF}$$

and

$$\mu_{Lf} - \mu_f.$$

- Similar to our approach to our first question, we might average these two effects (i.e., average over large fish(F)) to get an average effect of limpets when small fish are present.
- We also see the (single) effect of limpets in the absence of small fish:

$$\mu_L - \mu_C.$$

- Thus, to get at our question, it seems reasonable, to consider the difference of the above (average and single) effects to arrive at the linear combination,

$$\gamma_4 = \frac{1}{2}(\mu_{LfF} - \mu_{fF}) + \frac{1}{2}(\mu_{Lf} - \mu_f) - (\mu_L - \mu_C).$$

- You may be able to see how this translates to β_j effects notation already. But, we go through maringal means first, as with question 1, to be careful (again, see Factor level (marginal) means bullet in §16.1, above).

$$\gamma_4 = \frac{1}{2}(\mu_{.6} - \mu_{.3}) + \frac{1}{2}(\mu_{.5} - \mu_{.2}) - (\mu_{.4} - \mu_{.1}).$$

- As we did for γ_1 , we plug in the definition of the β_j effect (any constraints),

$$\beta_j = \mu_{\cdot j} - \mu_{\cdot \cdot},$$

to get

$$\gamma_4 = \frac{1}{2}(\beta_6 - \beta_3) + \frac{1}{2}(\beta_5 - \beta_2) - (\beta_4 - \beta_1).$$

- Or, using the suggestive subscripts again, we may write,

$$\gamma_4 = \frac{1}{2}(\beta_{LfF} - \beta_{fF}) + \frac{1}{2}(\beta_{Lf} - \beta_f) - (\beta_L - \beta_C).$$

- Now, as in question 1, to proceed further, we must consider the particular constraints/coding of the effects model.
- For the treatment constraint, recall $\beta_1 = 0$ so that β contains (among others) all β_j , β_2 to β_b , but not β_1 .
- Unlike the treatment constraint case for question 1, we have some work to do, but not much:

$$\gamma_4 = \frac{1}{2}(\beta_6 - \beta_3) + \frac{1}{2}(\beta_5 - \beta_2) - \beta_4 \quad \text{treatment.}$$

- The sum-to-zero constraints' β contains β_1 to β_{b-1} , because $\beta_b = -\sum_{j=1}^{b-1} \beta_j$ ($b = 6$).
- Thus,

$$\gamma_4 = \frac{1}{2}\beta_1 - \beta_2 - \beta_3 - \frac{3}{2}\beta_4 \quad \text{sum-to-zero.}$$

```
> ## Are we sure we know how R orders factor levels?
> ## Careful that the ordering is in the fitted object!
> levels(case1301.df$Treat) ## reflects changes above
[1] "C"     "f"     "fF"    "L"     "Lf"    "LfF"
> #####
```

```

> ## Treatment analysis:
> #####
> case1301TCR.lm ## ok, level order looks good

Call:
lm(formula = log(Cover/(100 - Cover)) ~ Block + Treat, data = case1301.df)

Coefficients:
(Intercept)      BlockB2      BlockB3      BlockB4      BlockB5
              -1.223       0.460       2.105       2.981       1.216
BlockB6      BlockB7      BlockB8   Treatf Treatff
              2.025       1.109       1.330      -0.494      -1.002
TreatL     TreatLf TreatLfF
             -1.892      -2.185      -2.905

> case1301TCR.lm$contrasts ## 1st time using this (?)

$Block
  B2 B3 B4 B5 B6 B7 B8
B1  0  0  0  0  0  0  0
B2  1  0  0  0  0  0  0
B3  0  1  0  0  0  0  0
B4  0  0  1  0  0  0  0
B5  0  0  0  1  0  0  0
B6  0  0  0  0  1  0  0
B7  0  0  0  0  0  1  0
B8  0  0  0  0  0  0  1

$Treat
[1] "contr.treatment"

> a<- 8; b<- 6 ## number of factor levels
> Ca<- rep(0,a-1) ## blocks
> Cb<- 1/2 * c(-1, -1, -2, 1, 1) ## grazing
> Cmat<- c(0,Ca,Cb) ## C matrix (vector)
> d<- 0 ## null CBeta value
> glh.test(reg=case1301TCR.lm, cm=Cmat, d=d)

Test of General Linear Hypothesis
Call:
glh.test(reg = case1301TCR.lm, cm = Cmat, d = d)
F = 0.1356, df1 = 1, df2 = 83, p-value = 0.7136

```

```

> ## Or, somewhat ugly, but readable and perhaps instructive
> ## (compare to ``Contrast Summary'' in Display 13.13 in R&S):
> ##
> estimable(obj=case1301TCR.lm, cm=Cmat, beta0=d, conf.int=0.95)

                                beta0 Estimate Std. Error
(0 0 0 0 0 0 0 -0.5 -0.5 -1 0.5 0.5)      0 0.095485  0.25932
                                         t value DF Pr(>|t|) Lower.CI
(0 0 0 0 0 0 0 -0.5 -0.5 -1 0.5 0.5) 0.36822 83  0.71365 -0.42028
                                         Upper.CI
(0 0 0 0 0 0 0 -0.5 -0.5 -1 0.5 0.5)  0.61125

> ## (Alas, R&S reorder the ``Treatment'' (factor) levels in
> ## Display 13.13!)
> #####
> ## Sum-to-zero analysis:
> #####
> case1301aR.lm$contrasts

$Block
 [,1] [,2] [,3] [,4] [,5] [,6] [,7]
B1    1    0    0    0    0    0    0
B2    0    1    0    0    0    0    0
B3    0    0    1    0    0    0    0
B4    0    0    0    1    0    0    0
B5    0    0    0    0    1    0    0
B6    0    0    0    0    0    1    0
B7    0    0    0    0    0    0    1
B8   -1   -1   -1   -1   -1   -1   -1

$Treat
 [,1] [,2] [,3] [,4] [,5]
C     1    0    0    0    0
f     0    1    0    0    0
fF    0    0    1    0    0
L     0    0    0    1    0
Lf    0    0    0    0    1
Lff   -1   -1   -1   -1   -1

> ## level order looks good
> ##
> Cb<- c(1/2, -1, -1, -3/2, 0) ## grazing
> Cmat<- c(0,Ca,Cb) ## C matrix (vector)
> glh.test(reg=case1301aR.lm, cm=Cmat, d=d)

```

```
Test of General Linear Hypothesis
Call:
glh.test(reg = case1301aR.lm, cm = Cmat, d = d)
F = 0.1356, df1 = 1, df2 = 83, p-value = 0.7136

> estimable(obj=case1301aR.lm, cm=Cmat, beta0=d, conf.int=0.95)

              beta0 Estimate Std. Error t value
(0 0 0 0 0 0 0 0.5 -1 -1 -1.5 0)      0 0.095485  0.25932 0.36822
                                         DF Pr(>|t|) Lower.CI Upper.CI
(0 0 0 0 0 0 0 0.5 -1 -1 -1.5 0) 83  0.71365 -0.42028  0.61125

> ## Again, results are the same for either coding, of course
```

16.10 Final Remarks

- The implementation of the above questions, as linear combinations of parameters, may seem a bit complicated at first.
- After all, [RS13, Sec. 13.3.4] simply use corresponding averages of observations, Y_{ijk} , to answer the questions above (though they omit much detail)—relatively easy to compute “by hand,” which was relatively important when computational resources were more limited (think: adding machines!).
- But, their approach of using simple averages generally only works in the **BALANCED** case, as in the current running example. Their presentation is a hold-over, in my opinion, of the classic balanced presentation of ANOVA that we still see in many textbooks. Our approach, however, works also in **UNBALANCED** cases (assuming there are not other complicating factors, like empty cells, or, more generally, estimability issues).
- [KNNL05, Chaps. 19-22], too, devote much of their presentation to the balanced case, wherein easy “hand” computations (simple averages) and

the classic ANOVA decomposition, which we discussed above, hold, before they move explicitly to unbalanced cases in [KNNL05, Chap 23], often referring to “the regression approach” to ANOVA in such unbalanced cases. [RS13] discuss the regression approach in various sections of their book ([RS13, pg 390, Sec. 13.3.5, 13.4.2, 13.4.3]). I did not check their constraint/coding scheme closely, but it appears to be more like R’s treatment coding than sum-to-zero coding.

- In this so-called “regression approach,” the concepts are the same between the balanced case, where special hand computations and non-ambiguous additive sum-of-squares ANOVA decomposition apply—as in the seaweed grazing example—and the unbalanced case, where these “features” do not apply in general. (Again, see Section 16.7.)
- I have made an attempt in these notes—indeed, in INF 511 and so far in INF 512—to feature the regression approach all along. It seems more modern to me and is more generally applicable.
- Essentially, this so-called “regression approach” is the F v R approach (aka extra sum-of-squares approach), where we had to fit a reduced model, explicitly, in addition to full model. As we’ve seen, the fitting of the reduced model is (almost!...ask me to explain in class) not necessary for standard interaction/main effects tests with the standard ANOVA printout in the *balanced* case.
- This regression approach may also be called something like “the general linear test” approach, as implemented in `glh.test` or `estimable`, which we’ve tended to call the $C\beta$ approach.
- Whichever implementation to choose—ANOVA table, F v R, or $C\beta$ —is mostly a matter of preference.
- We can imagine how the hand computations and additive decomposition of the balanced case were favored when computational resources were relatively limited some time ago. But, nowadays, perhaps with some special exceptions (e.g., big data), it seems to me that we should focus

more on the more generally useful “regression approach,” for which, in the vast majority of cases, computational burden is not an issue.

- Besides, the regression approach to ANOVA makes explicit the connection to regression—it’s just regression on specially coded covariates, and our inferences typically involve some linear combination(s) of the β vector. Moreover, it tends to force us to have a better understanding of software implementation, at least in R, with little risk to conceptual understanding.
- In any case, I remind you that it is probably good practice to heed the marginality (hierarchy) principle, mentioned above, except in rare, special circumstances, which will likely be dictated by (your) science, not by (some gap in your knowledge about) statistics. In other words, don’t worry about these special circumstances creating problems by accident.

Appendix A

Basic Results in Probability and Statistics

Contents

A.1	Review	537
A.2	Summation Operator	538
	A.2.1 Properties of Summation Operator	539
A.3	Double Summation Operator	540
A.4	Product Operator	540
A.5	R Example	540
A.6	Logarithms and Exponentiation	542
A.7	Random Variables	545
	A.7.1 Modeling Reality	547
	A.7.2 Discrete Random Variables	550
	A.7.3 Continuous Random Variables	558
	A.7.4 Some Remarks	565
A.8	Characteristics of Random Variables	566
	A.8.1 Expected (Mean) Value	566
	A.8.2 Variance Operator	570
A.9	Random Vectors	573
	A.9.1 Covariance Operator & Its Properties	573
	A.9.2 Independence	576
A.10	Linear Combinations of RVs	578
A.11	Central Limit Theorem	579
A.12	pdqr Functions in R	580

A.12.1 Example	582
--------------------------	-----

Main Objectives:

- Familiarize ourselves with basic results and notation to be used throughout the semester.
 - Don't worry, these results are not the main focus of the course; we use them, not vice-versa.
 - Learn more R.

0

Reading:

- Much of the material in this appendix is based on [KNNL05, Appendix A], but I don't expect you to read this unless you really want to.

-R

A.1 Review

- **Introductory Concepts & Methods.** I strongly suggest that you avail yourself of one or more fine instances of the internet meme of introductory statistical concepts and methods, including hypothesis testing & confidence intervals. For example, you may find
 - the (less technical) Statistical Reasoning (<https://oli.cmu.edu/courses/statistical-reasoning-copy/>) or
 - the (more technical) Probability & Statistics <https://oli.cmu.edu/courses/probability-statistics-open-free/>

Open & Free courses offered by the Open Learning Initiative to be helpful. These links are also in our INF 511 BbLearn course shell.

- **Things You Should Know.** If you do not know or are unsure of the any of the following, then please spend time visiting the above (or similar) sites: null hypothesis, alternative hypothesis, type I error, type I error probability, type II error, type II error probability, power, p-value, confidence interval, confidence level or confidence coefficient, average of a set of numbers, standard deviation of a set of numbers, correlation between two (equal size) sets of numbers, median, mode, histogram, one-sample t-test or z-test, two-sample t-test or z-test, paired t-test, z-tables, t-tables, probabilities are numbers between 0 and 1.
- **Going Way Back?** Properties of logs and exponents (e.g., log of a product is sum of logs, add exponents for same base in a product) and general high school / college algebra.

A.2 Summation Operator

$$\sum_{i=1}^n Y_i = Y_1 + Y_2 + \cdots + Y_n$$

- **Up, Random, Down, Fixed.** Often, statisticians strive to use **uppercase** letters toward the end of the alphabet to denote random variables (quantities that are uncertain before being observed; see definition, below) and **lowercase** letters to denote fixed values of the random variables. E.g., Y_1 and y_1 . We **may deviate** from this practice!

A.2.1 Properties of Summation Operator

$$\sum_{i=1}^n k = \overbrace{k + k + \cdots + k}^{\text{n times}} = nk \quad k \text{ some constant}$$

$$\begin{aligned}\sum_{i=1}^n (Y_i + Z_i) &= Y_1 + Z_1 + \\ &\quad Y_2 + Z_2 + \\ &\quad \vdots \\ &\quad Y_n + Z_n + \\ &= \sum_{i=1}^n Y_i + \sum_{i=1}^n Z_i\end{aligned}$$

$$\begin{aligned}\sum_{i=1}^n aY_i &= aY_1 + aY_2 + \cdots + aY_n \\ &= a(Y_1 + Y_2 + \cdots + Y_n) \\ &= a \sum_{i=1}^n Y_i\end{aligned}$$

Above results imply $\sum_{i=1}^n (a + bY_i) = na + b \sum_{i=1}^n Y_i$

A.3 Double Summation Operator

$$\begin{aligned}\sum_{i=1}^n \sum_{j=1}^m Y_{ij} &= \sum_{i=1}^n (Y_{i1} + Y_{i2} + \cdots + Y_{im}) \\&= Y_{11} + Y_{12} + \cdots + Y_{1m} + \\&\quad Y_{21} + Y_{22} + \cdots + Y_{2m} + \\&\quad \vdots \\&\quad Y_{n1} + Y_{n2} + \cdots + Y_{nm} + \\&= \sum_{j=1}^m (Y_{1j} + Y_{2j} + \cdots + Y_{nj}) \\&= \sum_{j=1}^m \sum_{i=1}^n Y_{ij}\end{aligned}$$

A.4 Product Operator

$$\prod_{i=1}^n Y_i = Y_1 Y_2 \cdots Y_n$$

A.5 R Example

- The following code chunk explores some of R's capabilities for computing sums and products.

```
> ## Concatenate numbers into an R vector:  
> myvector<- c(5,6,9,2,9,4,8,2,1,4)  
> (n<- length(myvector))  
  
[1] 10  
  
> myvector  
  
[1] 5 6 9 2 9 4 8 2 1 4  
  
> ## Name the elements because we care  
> names(myvector)<- paste("y[",1:n,"]",sep="")  
> myvector  
  
y[1]   y[2]   y[3]   y[4]   y[5]   y[6]   y[7]   y[8]   y[9]   y[10]  
      5       6       9       2       9       4       8       2       1       4  
  
> ## Alternatively:  
> names(myvector)<- paste0("y[",1:n,"]")  
> myvector  
  
y[1]   y[2]   y[3]   y[4]   y[5]   y[6]   y[7]   y[8]   y[9]   y[10]  
      5       6       9       2       9       4       8       2       1       4  
  
> ## What's the 5th element myvector?  
> ## ("[]" is an extraction operator.)  
> myvector[5]  
  
y[5]  
 9  
  
> ## Sum and product of elements in myvector  
> sum(myvector)  
  
[1] 50  
  
> prod(myvector)  
  
[1] 1244160
```

A.6 Logarithms and Exponentiation

This material is fundamental to INF 512 Modern Regression II (logistic (binomial) and log-linear (Poisson) models) and to INF 504 Data Mining and Machine Learning (classification (multinomial)).

- **Exponentiation** of a **base**, b , and **exponent**, a , is defined by

$$b^a.$$

A very popular, special case is base

$$b = e \approx 2.71828,$$

Euler's number, and we use the notation

$$\exp(a) \equiv e^a.$$

We can **change to base** e ,

$$b^x = e^{x \log_e(b)}.$$

- **The product of exponentials with the same base, b , is the exponential of the sum of the exponents,**

$$b^{a_1} b^{a_2} \dots b^{a_n} = b^{a_1 + a_2 + \dots + a_n},$$

e.g., as we might see in log-linear (Poisson) modeling (INF 512),

$$\exp(\beta_0) \exp(\beta_1 x_1) \dots \exp(\beta_k x_k) = \exp(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k).$$

```

> a<- c(1,3,5)
> b<- 2
> ## (Exponentiation is `vectorized', applied to each element of vector
> ## a.)
> b^a ## vectorized operation

[1] 2 8 32

> prod(b^a) ## product of exponentials

[1] 512

> b^(sum(a)) ## exponential of sum of exponents

[1] 512

```

- **Exponentiation of a negative exponent is the multiplicative inverse of the exponentiation,** (using base e to illustrate)

$$\exp(-a) = \frac{1}{\exp(a)},$$

i.e.,

$$\exp(-a) = (\exp(a))^{-1},$$

```

> a <- 2
> exp(-a) ## exponentiation of neg. exponent

[1] 0.13534

> 1/exp(a) ## inverse of exponentiation

[1] 0.13534

> exp(a)^{-1}

[1] 0.13534

```

- **The logarithm of a product (of (positive) multiplicands) is the sum of the logarithm of the multiplicands,**

$$\log(a_1, a_2, \dots, a_k) = \log(a_1) + \log(a_2) + \dots + \log(a_n).$$

```
> a<- c(1,3,5)
> log(prod(a)) ## log of product
[1] 2.7081

> sum(log(a)) ## sum of logs
[1] 2.7081

> ## NOTE: The log function in R is *NATURAL LOG*. (log10 is base 10).
```

- **The log of a ratio is the difference of the logs,**

$$\log(a/b) = \log(a) - \log(b),$$

a, b positive.

```
> a<-2; b<-3
> log(a/b) ## log of ratio
[1] -0.40547

> log(a) - log(b) ## difference of logs
[1] -0.40547
```

A.7 Random Variables

Definition A.1 (Experiment).

- *A process of obtaining an observation / measurement / outcome / response from an object.*
- *Note that this definition is very broad relative to its more typical meaning in, say, “randomized experiments,” for which we will have more to say, later.*

Definition A.2 (Unit).

- *An object from which an observation / measurement / outcome / response is obtained in an experiment.*
- *Context will dictate use of other, relatively common and synonymous terms such as observational units, experimental units, sampling units, subjects, participants (and then some!).*

Definition A.3 (Variable).

- *An observation / measurement / outcome / response obtained from a unit in an experiment.*
- *A variable may be (i) categorical (non-numerical) or (ii) numerical.*
- *Because a categorical variable may be coded numerically, variables are often treated as numerical, although the meaning of the numerical codes for categorical variables is usually not the same as the numbers associated with numerical variables.*

- E.g., hair color may be treated as a categorical variable with different values: black, blonde, brown, red, and other, which may be coded numerically as, e.g., 0,1,2,3,4. But, in this case, the numbers are not meant to imply any order or any meaning to differences or ratios; what would $2-1=1$ or $2/1=2$ mean here? Nothing.
- More on categorical variables, below.

Definition A.4 (Random Variable).

- A variable that **cannot be predicted with certainty before it is observed**.
- In other words, a variable whose **value is uncertain before it is observed**.
- **Notation (again)**: We often use uppercase letters, e.g., X , Y , Z , to denote a random variable, before it is observed, and lowercase letters, e.g., x , y , z , to denote a particular value, e.g., 5.7
- There are **more technical definitions** of a random variable, but I do not see how such definitions serve us.
- E.g., the weight of your cat, your blood serum cholesterol level, the next president of the United States, the diameters of trees in a tract of land, etc.

Definition A.5 (Population).

- **The complete set of units of interest**. We may denote the total number of units in a population (**population size**) with N , though we do not consider population size much in this class.

- Ideally, the population is well defined.
- In most practical situations, this set of units is almost always beyond us, in some sense, so, instead, we focus on a **subset of the population (sample)**.
- Most often (not always), we use a **mathematical model** as an approximation to the distribution of values from the actual population of N units. In such cases, we sometimes refer to the mathematical approximation as the **superpopulation** because, for example, a normal distribution approximation represents an uncountable infinity of possible outcomes, which no real population has, as far as I know.

Remark A.1 (Population of Units or Population of Values?). Note that some people (e.g., statisticians) may prefer to discuss sample and population in terms of the values of the (random) variables (outcomes) that may be observed from units—a sample/population of values rather than sample/population of units. Context usually clarifies.

Definition A.6 (Sample).

- A **subset of units (or their values) from a population** (if well defined). Data set. We often denote the total number of units/values in a sample (**sample size**) as n .

A.7.1 Modeling Reality

With the above definitions, figure A.1 depicts a very general, perhaps simplistic, view of how we do statistics.

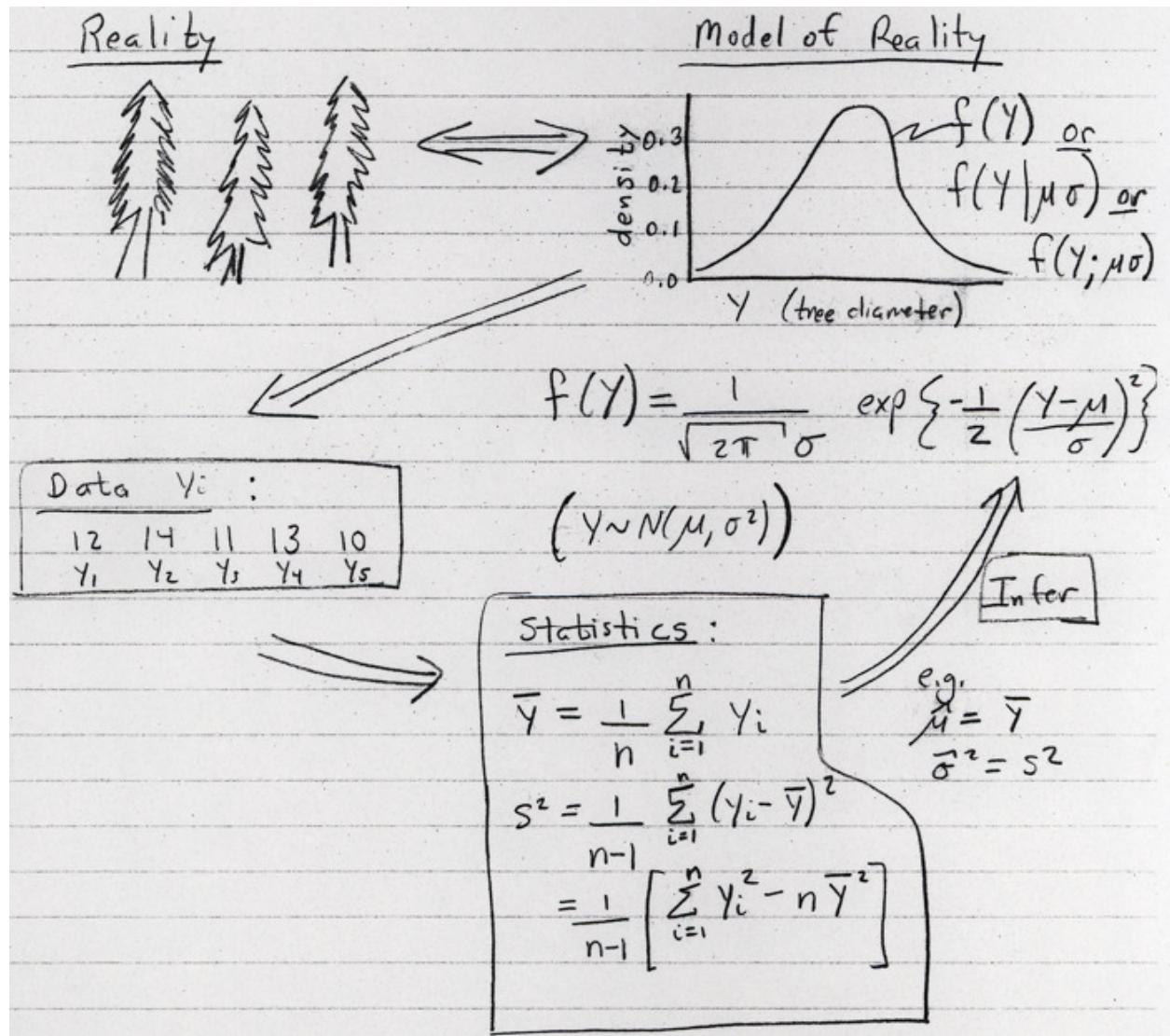


Figure A.1: A view of the process of doing statistics.

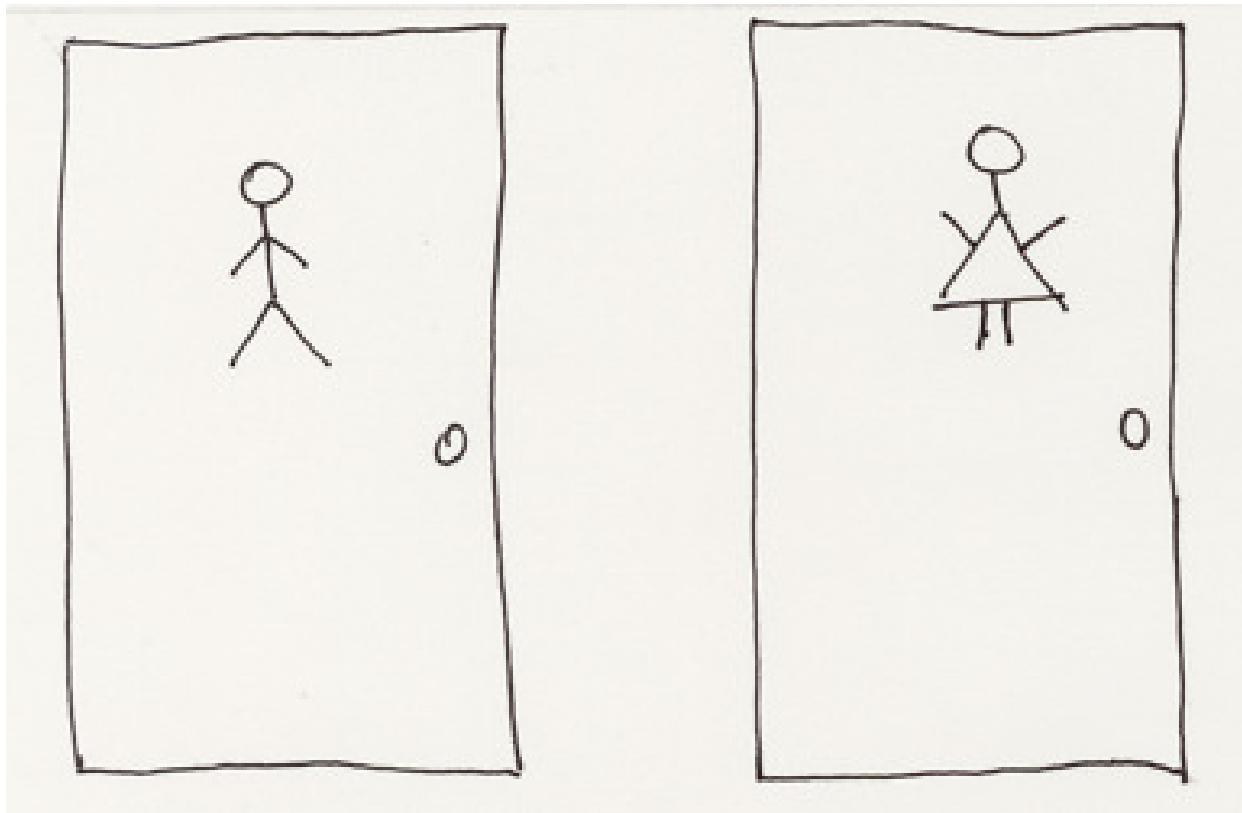


Figure A.2: A useful model of reality?

Notice in the figure the abstraction of reality via a **model**. To paraphrase a famous statistician:

“All models are wrong...some are useful” [Box76].

What did Box mean? Figure A.2 is a stick-figure model of two people. If you identify with one of the stick figures, and you were to walk into a door displaying the other figure, you might be embarrassed, or worse. Sure, the model isn't even close to being real in almost any sense of the word, but it is real enough and very useful.

In Other Words. We typically abstract the reality of a population (of values) into a distributional model (often called a **super-population**). More importantly, we hope the simplification leads us to some understanding, to infer something

about reality. As Figure A.1 suggests, we collect data on reality, treat the data *as if* it has arisen from our (hopefully useful) model of reality, and use the data to make inference about some aspect of the model, hence of reality.

- **Random But Structured.** While we cannot predict values of random variables with certainty, the collection of possible values of a random variable **behave in some structured manner** as often formalized by (i.e., as abstracted into a mathematical model by) a **probability density function (pdf)** (continuous random variables, below), **probability mass function (pmf)** (discrete random variables, below) or **cumulative (probability) distribution function (cdf)** (continuous or discrete) (or by other, related functions, such as moment generated functions (mgfs) or characteristic functions (we won't deal with these latter two functions)). This structure is often manifested from observations of the random variable in, e.g., histograms.
- **E.g.**, though we may not know our blood cholesterol levels before we observed them, histograms of such measurements tend to have a typical form, perhaps "normal-looking", which suggests use of a normal pdf or "bell curve" as a model for such data.
- A "large" sample size may give us good reason to assume that the (a) **CLT** is operating to give approximate normality. We give a definition of CLT, later.

A.7.2 Discrete Random Variables

Figure A.3 captures such a (probability) structure of a discrete random variable in a nutshell.

Definition A.7 (Probability Mass Function (pmf)). *Figure A.3 (left side) may*

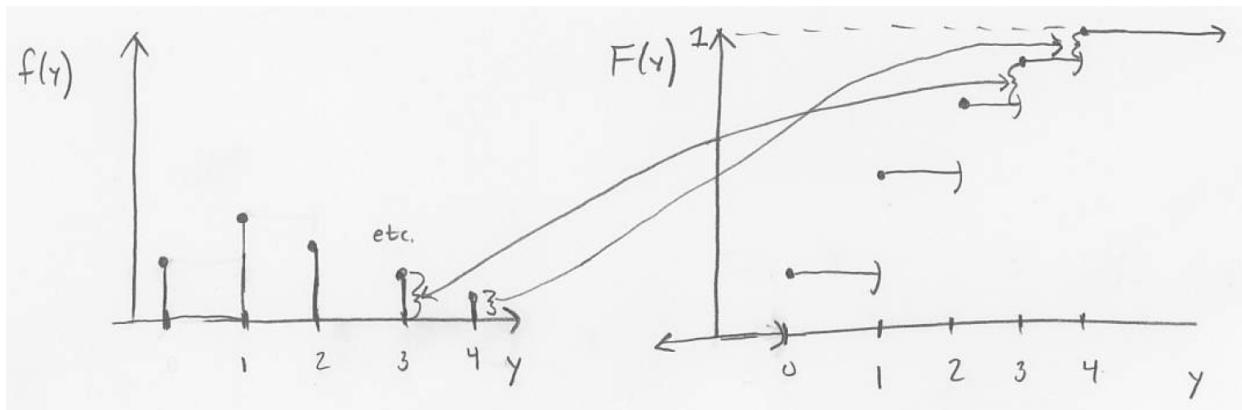


Figure A.3: Relationship between pmf and cdf.

be formalized mathematically: If

$$f(y) = P(Y = y),$$

then $f(y)$ is called the **probability mass function (pmf)**, where $P(Y = y)$ is shorthand for “the probability that random variable Y equals the particular value, y ”.

Definition A.8 (cdf).

- The **cumulative (probability) distribution function** of a random variable Y is given by

$$F(y) = P(Y \leq y),$$

where $P(Y \leq y)$ is interpreted as the “probability” of the random variable Y being **less than or equal to** some number y .

- This generic definition suffices for discrete and continuous rvs, discussed below.

Definition A.9 (Discrete Random Variable).

- *Intuitively, of course, a discrete random variable can assume only a countable, perhaps finite, number of values (with positive probability).*
- *More formally, Y is said to be a **discrete random variable** if its cdf is a step function.*

Example A.1 (Binomial Random Variable).

- **Popular.** *Perhaps the most popular discrete random variable (distributional model) is the binomial random variable, fundamental to **logistic regression** ([Wak13, Chap. 7]); INF 512.) It generalizes to the **multinomial**, used a lot in **classification problems** in machine learning; INF 504).*
- **Success Counts.** *Let Y be an rv that counts the number of 'successes' in n independent Bernoulli trials with constant probability of 'success' from trial to trial. A Bernoulli trial is an experiment (by our simple definition above) that results in one of only two possible outcomes (0/1, failure/success, absence/presence, negative/positive, no disease / disease, etc.).*
- **Binomial RV.** *In this case, Y is a binomial random variable, which we denote as*

$$Y \sim \text{binom}(n, p),$$

where n is the (integer greater than zero) number of trials (a (usually) known parameter), and $0 < p < 1$ is the probability of success (usually unknown, to be estimated with data).

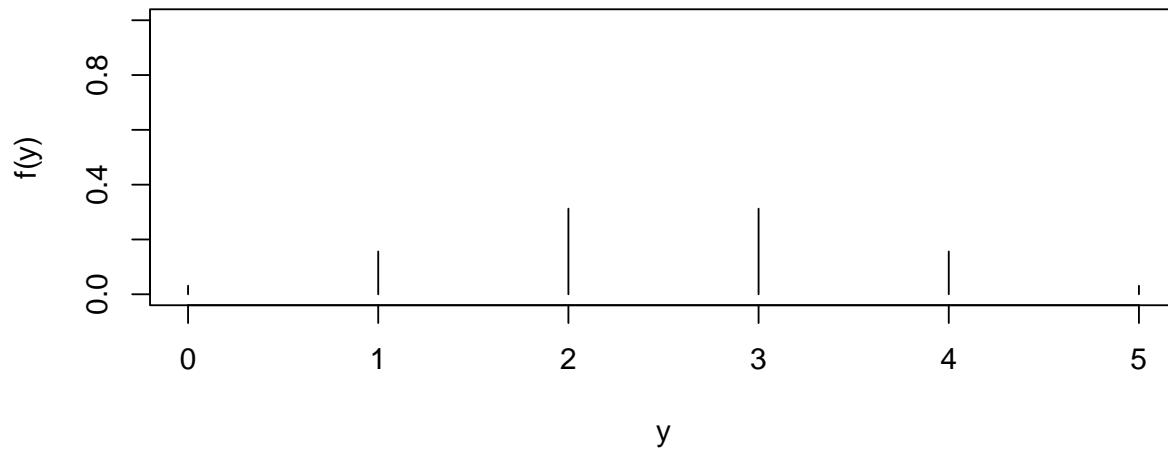
- **E.g. Head Counts.** *For example, we may model the toss of a coin as a Bernoulli trial, and we may reasonably assume that successive tosses are*

independent—the outcome of one toss telling us nothing of the outcomes of other tosses—each toss having the same probability of resulting in a head. (Our modeling assumes, of course, only tail/head outcomes, no sides, etc!) See continued example, below.

```
> par(mfrow=c(2,1))
> ## pmf for binomial(n=5,p=0.5).
> plot(0:5, binprobs<- dbinom(0:5,size=5,p=0.5),
+       ylab="f(y)", xlab="y", type="h",
+       pch=20, main="f(y)=P(Y=y)\n binom(n=5,p=0.5)",
+       ylim=c(0,1))
> ## cdf for binomial(n=5,p=0.5).
> plot(0:5, binprobs<- pbinom(0:5,size=5,p=0.5),
+       ylab="F(y)", xlab="y", type="h",
+       pch=20, main="F(y) = P(Y<=y)\nbinom(n=5,p=0.5)",
+       ylim=c(0,1))
```

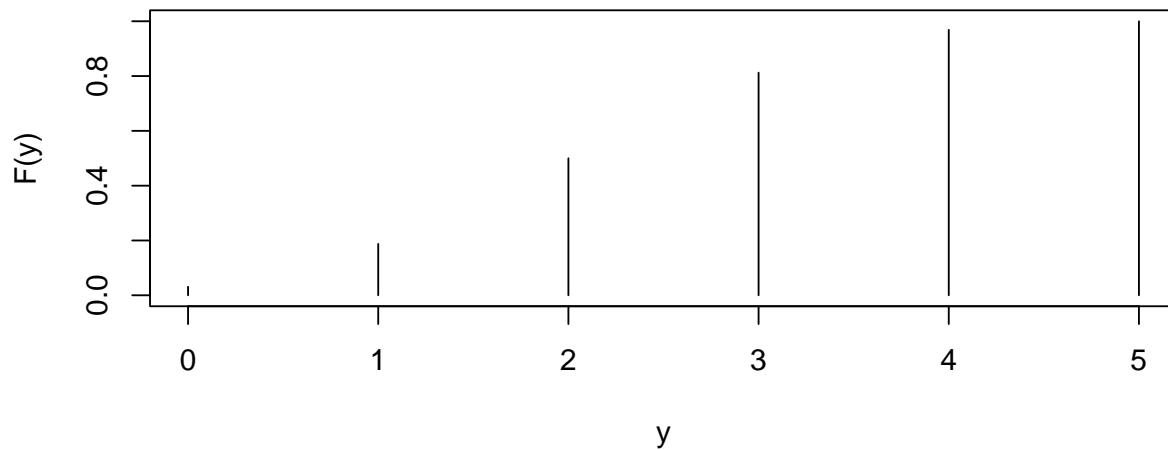
$$f(y) = P(Y=y)$$

$$\text{binom}(n=5, p=0.5)$$



$$F(y) = P(Y \leq y)$$

$$\text{binom}(n=5, p=0.5)$$



```
> par(mfcol=c(1, 1))
```

Definition A.10 (Binomial pmf).

- The pmf of a binomial random variable $\text{binom}(n, p)$ is given by

$$f(y) = P(Y = y) = \begin{cases} \binom{n}{y} p^y (1-p)^{(n-y)} & y = 0, 1, 2, \dots, n \\ 0 & \text{otherwise} \end{cases}$$

Definition A.11 (Binomial cdf).

-

$$\begin{aligned} F(y) &= P(Y \leq y) = \sum_{i \leq y} P(Y = i) \\ &= \sum_{i \leq y} \binom{n}{i} p^i (1-p)^{(n-i)}, \end{aligned}$$

where summation occurs over integers in $[0, y]$ and n and $0 < p < 1$ are the same as the parameters of the binomial pmf, with n being an integer greater than zero.

- In this discrete rv case, $F(y)$ actually sums probability to the left of (and including) y . (Don't let recycled notation confuse you.)

Example A.2 (Tossing a Coin).

- What's the probability of getting two heads in five (independent) tosses of a fair coin? (Model: $Y \sim \text{binom}(n = 5, p = 0.5)$, if not obvious.)
- Use the "binomial table" given nearby, or use R.
- (Incidentally, look at $P(Y \leq 4)$ for $Y \sim \text{binom}(5, 0.5)$ in the table; this could be intentional...more in class.)
- The following chunk explores some of Rs capabilities for computing binomial probabilities.

Table II Cumulative Binomial Probabilities $P(X \leq x)$

n	x	P										
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.95	0.99
1	0	0.9000	0.8000	0.7000	0.6000	0.5000	0.4000	0.3000	0.2000	0.1000	0.0500	0.0100
	2	0	0.8100	0.6400	0.4900	0.3600	0.2500	0.1600	0.0900	0.0400	0.0100	0.0025 0.0001
3	1	0.9900	0.9600	0.9100	0.8400	0.7500	0.6400	0.5100	0.3600	0.1900	0.0975	0.0199
	0	0.7290	0.5120	0.3430	0.2160	0.1250	0.0640	0.0270	0.0080	0.0010	0.0001	0.0000
4	1	0.9720	0.8960	0.7840	0.6480	0.5000	0.3520	0.2160	0.1040	0.0280	0.0073	0.0003
	2	0.9990	0.9920	0.9730	0.9360	0.8750	0.7840	0.6570	0.4880	0.2710	0.1426	0.0297
5	0	0.6561	0.4096	0.2401	0.1296	0.0625	0.0256	0.0081	0.0016	0.0001	0.0000	0.0000
	1	0.9477	0.8192	0.6517	0.4752	0.3125	0.1792	0.0837	0.0272	0.0037	0.0005	0.0000
6	2	0.9963	0.9728	0.9163	0.8208	0.6875	0.5248	0.3483	0.1808	0.0523	0.0140	0.0006
	3	0.9999	0.9984	0.9919	0.9744	0.9375	0.8704	0.7599	0.5904	0.3439	0.1855	0.0394
7	0	0.5905	0.3277	0.1681	0.0778	0.0313	0.0102	0.0024	0.0003	0.0000	0.0000	0.0000
	1	0.9185	0.7373	0.5282	0.3370	0.1875	0.0870	0.0308	0.0067	0.0005	0.0000	0.0000
8	2	0.9914	0.9421	0.8369	0.6826	0.5000	0.3174	0.1631	0.0579	0.0086	0.0012	0.0000
	3	0.9995	0.9933	0.9692	0.9130	0.8125	0.6630	0.4718	0.2627	0.0815	0.0226	0.0010
9	4	1.0000	0.9997	0.9976	0.9898	0.6988	0.9222	0.8319	0.6723	0.4095	0.2262	0.0490
	0	0.5314	0.2621	0.1176	0.0467	0.0156	0.0041	0.0007	0.0001	0.0000	0.0000	0.0000
10	1	0.8857	0.6554	0.4202	0.2333	0.1094	0.0410	0.0109	0.0016	0.0001	0.0000	0.0000
	2	0.9842	0.9011	0.7443	0.5443	0.3438	0.1792	0.0705	0.0170	0.0013	0.0001	0.0000
11	3	0.9987	0.9830	0.9295	0.8208	0.6563	0.4557	0.2557	0.0989	0.0159	0.0022	0.0000
	4	0.9999	0.9984	0.9891	0.9590	0.9806	0.7667	0.5798	0.3446	0.1143	0.0328	0.0015
12	5	1.0000	0.9999	0.9993	0.9959	0.9844	0.9533	0.8824	0.7379	0.4686	0.2649	0.0585
	0	0.4783	0.2097	0.0824	0.0280	0.0078	0.0016	0.0002	0.0000	0.0000	0.0000	0.0000
13	1	0.8503	0.5767	0.3294	0.1586	0.0625	0.0188	0.0038	0.0004	0.0000	0.0000	0.0000
	2	0.9743	0.8520	0.6471	0.4199	0.2266	0.0963	0.0288	0.0047	0.0002	0.0000	0.0000
14	3	0.9973	0.9667	0.8740	0.7102	0.5000	0.2898	0.1260	0.0333	0.0027	0.0002	0.0000
	4	0.9998	0.9953	0.9712	0.9037	0.7734	0.5801	0.3529	0.1480	0.0257	0.0038	0.0000
15	5	1.0000	0.9996	0.9962	0.9812	0.9375	0.8414	0.6706	0.4233	0.1497	0.0444	0.0020
	6	1.0000	1.0000	0.9998	0.9984	0.9922	0.9720	0.9176	0.7903	0.5217	0.3017	0.0679
16	0	0.4305	0.1678	0.0576	0.0168	0.0039	0.0007	0.0001	0.0000	0.0000	0.0000	0.0000
	1	0.8131	0.5033	0.2553	0.1064	0.0352	0.0085	0.0013	0.0001	0.0000	0.0000	0.0000
17	2	0.9619	0.7969	0.5518	0.3154	0.1445	0.0498	0.0113	0.0012	0.0000	0.0000	0.0000
	3	0.9950	0.9437	0.8059	0.5941	0.3633	0.1737	0.0580	0.0104	0.0004	0.0000	0.0000
18	4	0.9996	0.9896	0.9420	0.8263	0.6367	0.4059	0.1941	0.0563	0.0050	0.0004	0.0000
	5	1.0000	0.9988	0.9887	0.9502	0.8555	0.6846	0.4482	0.2031	0.0381	0.0058	0.0001
19	6	1.0000	0.9999	0.9987	0.9915	0.9648	0.8936	0.7447	0.4967	0.1869	0.0572	0.0027
	7	1.0000	1.0000	0.9999	0.9993	0.9961	0.9832	0.9424	0.8322	0.5695	0.3366	0.0773
20	8	0	0.3874	0.1342	0.0404	0.0101	0.0020	0.0003	0.0000	0.0000	0.0000	0.0000
	1	0.7748	0.4362	0.1960	0.0705	0.0195	0.0038	0.0004	0.0000	0.0000	0.0000	0.0000
21	2	0.9470	0.7382	0.4628	0.2318	0.0889	0.0250	0.0043	0.0003	0.0000	0.0000	0.0000
	3	0.9917	0.9144	0.7297	0.4826	0.2539	0.0994	0.0253	0.0031	0.0001	0.0000	0.0000
22	4	0.9991	0.9804	0.9012	0.7334	0.5000	0.2666	0.0988	0.0196	0.0009	0.0000	0.0000
	5	0.9999	0.9969	0.9747	0.9006	0.7461	0.5174	0.2703	0.0856	0.0083	0.0006	0.0000
23	6	1.0000	0.9997	0.9957	0.9750	0.9102	0.7682	0.5372	0.2618	0.0530	0.0084	0.0001
	7	1.0000	1.0000	0.9996	0.9962	0.9805	0.9295	0.8040	0.5638	0.2252	0.0712	0.0034
24	8	1.0000	1.0000	1.0000	0.9997	0.9980	0.9899	0.9596	0.8658	0.6126	0.3698	0.0865

Figure A.4: Table of cumulative binomial probabilities (Source: forgotten!).

```
> ## cdf F(2) (for tossing fair coin 5 times (binom(n=5,p=0.5))
> pbinom(2,size=5,prob=0.5)

[1] 0.5

> ## Incidentally (see the nearby table)
> pbinom(4,5,0.5)

[1] 0.96875
```

Definition A.12 (Poisson RV).

- Quickly, another every popular model for counts is the Poisson. It is fundamental to Poisson regression (usually **log-linear regression**) (INF 512).
- $Y \sim \text{Pois}(\lambda)$, $\lambda > 0$, with pmf

$$f(y) = \begin{cases} \frac{\lambda^y \exp(-\lambda)}{y!} & y = 0, 1, 2, \dots \\ 0 & \text{otherwise} \end{cases}$$

- Take INF 512!

```
> ## pois(lambda = 1) pmf cheap example
> dpois(x=1, lambda=1)

[1] 0.36788
```

Definition A.13 (Categorical or Qualitative Variable). A categorical (or qualitative) variable assumes values that are labels or categories that describe (non-quantitatively) a unit. If the labels have some natural order, then the variable is **ordinal**, otherwise it's **nominal** (just names, without an order). a.k.a,

classification variable whose possible value are called *classes*, or **grouping variable** whose values are called *groups*, or **factor (variable)** whose values are called *levels*. (We mentioned *categorical variables*, previously.)

- For example, we may record the presence or absence of a nest of a certain bird species in forest trees.
- Or, we may want to classify images as a car, motorcycle, bicycle, pedestrian, or ‘other’.
- We may numerically encode categorical variables as a 0/1, for absent/present or 1,2,3,4,5 for car, etc. Further, we may model the variable as a Bernoulli random variable ($\text{binomial}(n=1, p)$), in the first case, or multinomial in latter case.
- A two-category variable (or a variable that counts outcomes in two categories) is usually modeled by a **binomial**. This extends to multi-category variables (our counters thereof) and the multinomial (INF 504/512).

A.7.3 Continuous Random Variables

Similar to the previous Figure A.3, for discrete random variables, Figure A.5 captures the structure of a continuous random variable.

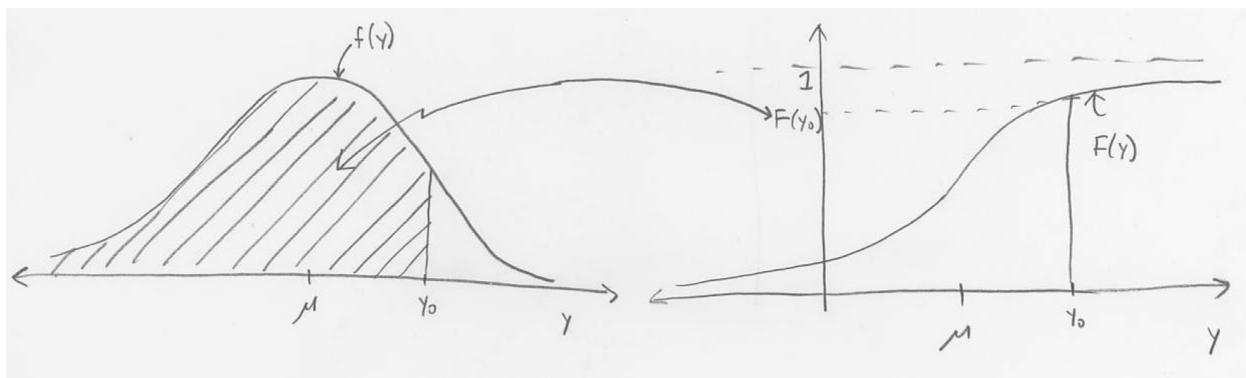


Figure A.5: Relationship between pdf and cdf.

Definition A.14 (Probability Density Function (pdf)).

- If there exists a function $f(y)$ such that

$$F(y) = \int_{-\infty}^y f(t) dt$$

is a **cdf**, then $f(y)$ is called a **probability density function (pdf)**.

- Note that this definition is for a generic continuous random variable. (We do not discuss integration wrt a counting measure in which case an integral definition would suffice for discrete rvs, too.)
- **cdf** As we said, above, our previous definition of cdf applies here, too. Here, if we have such a pdf function, f , then the cdf, F , can be expressed as in the above integral.

Definition A.15 (Continuous Random Variable). • Intuitively, but somewhat loosely speaking, a continuous random variable can (conceptually) assume any **values in an interval**.

- More formally, Y is said to be a **continuous random variable** if its cdf is a continuous function.

- We may use continuous rvs to model measurements such as **height**, **weight**, **area**, **volume**, etc.
- Also, intuitively, a continuous random variable has positive probability of being in an interval (of positive length), no matter how small (except in cases such as negative height, etc.).

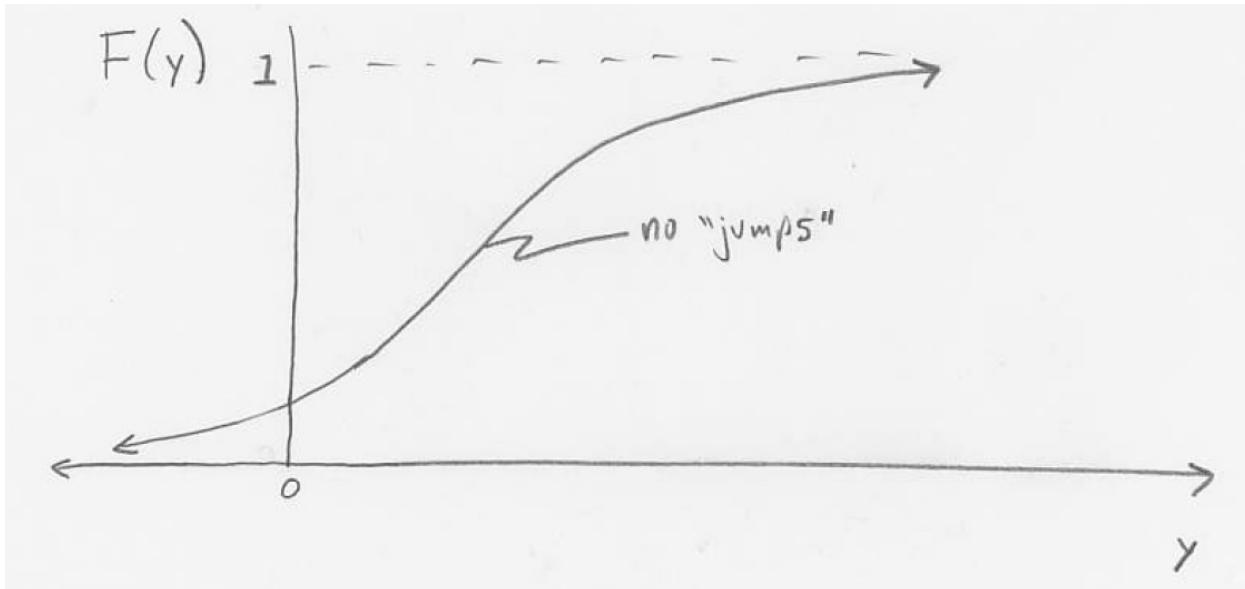


Figure A.6: Continuous cdf.

- Note that $P(Y = y) = 0$ for a continuous random variable, Y , no matter the value y , unlike discrete random variables.

Example A.3 (Normal (Gaussian) Random Variable pdf and cdf).

- We write

$$Y \sim N(\mu, \sigma^2)$$

as short-hand notation for a random variable, Y , with a normal distribution with parameters $-\infty < \mu < \infty$ and $\sigma > 0$.

- pdf.

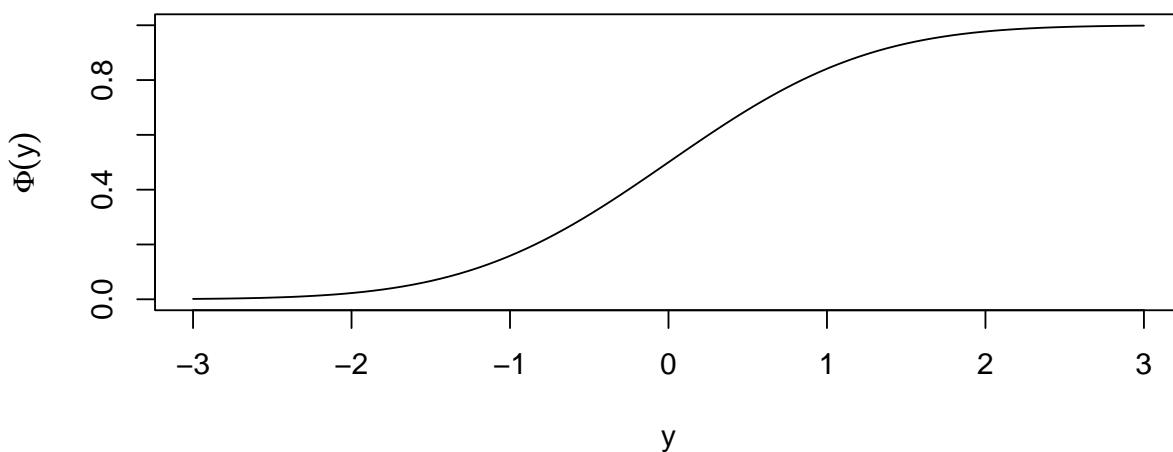
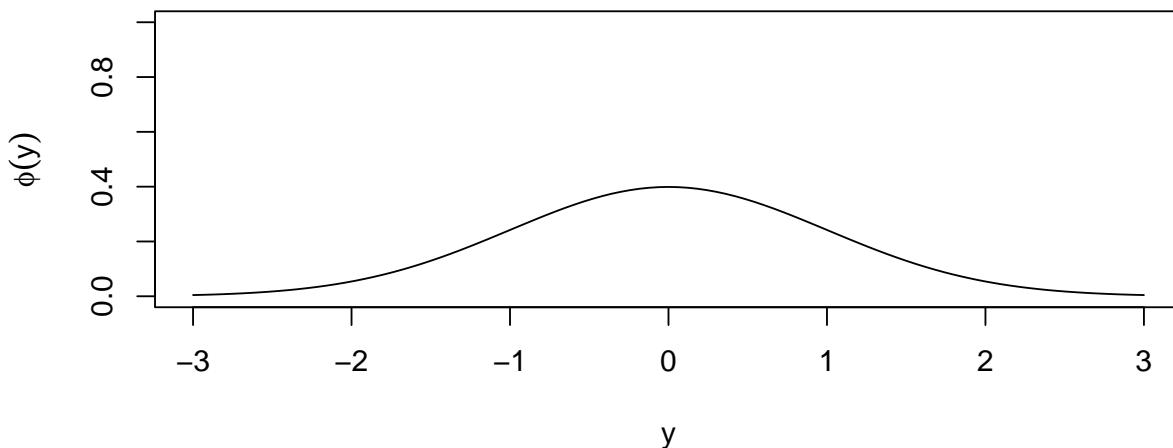
$$f(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2} \left(\frac{y - \mu}{\sigma}\right)^2\right) \quad -\infty < y < \infty.$$

- cdf.

$$F(y) = \int_{-\infty}^y f(t) dt.$$

- We can loosely think of $F(y)$ as “summing” probability to the left of some value, y , analogous to but technically different from cdf of a discrete rv, which does actually sum probabilities (perhaps an infinite sum).
- ϕ and Φ are often reserved to denote the pdf and cdf, respectively, of the standard normal distribution.
- See Figure A.7 (from [KNNL05, Table B.1]) for a standard normal (i.e., $N(0, 1)$) “cdf table,” i.e., a “z-table.” You should already know how to use a z-table or the digital version thereof!!!
- We will use R, however, to compute normal (or other) probabilities for us.

```
> par(mfrow=c(2,1))
> ## N(0,1) pdf (often denoted lowercase phi)
> curve(dnorm(x,mean=0,sd=1), from=-3, to = 3,
+         ylab=expression(phi(y)), xlab="y", ylim=c(0,1))
> ## N(0,1) cdf (often denoted uppercase phi)
> curve(pnorm(x,mean=0,sd=1), from=-3, to = 3,
+         ylab=expression(Phi(y)), xlab="y", ylim=c(0,1))
```



```
> par(mfcol=c(1,1))
```

Example A.4 (Z-Table). Use Figure A.7 ([KNNL05, Table B.1]) to obtain $P(Z \leq 1.96)$ where Z is a standard normal rv (my shorthand for “random variable”).

TABLE B.1 Cumulative Probabilities of the Standard Normal Distribution.

z	Entry is area A under the standard normal curve from $-\infty$ to $z(A)$									
	.00	.01	.02	.03	.04	.05	.06	.07	.08	.09
.0	.5000	.5040	.5080	.5120	.5160	.5199	.5239	.5279	.5319	.5359
.1	.5398	.5438	.5478	.5517	.5557	.5596	.5636	.5675	.5714	.5753
.2	.5793	.5832	.5871	.5910	.5948	.5987	.6026	.6064	.6103	.6141
.3	.6179	.6217	.6255	.6293	.6331	.6368	.6406	.6443	.6480	.6517
.4	.6554	.6591	.6628	.6664	.6700	.6736	.6772	.6808	.6844	.6879
.5	.6915	.6950	.6985	.7019	.7054	.7088	.7123	.7157	.7190	.7224
.6	.7257	.7291	.7324	.7357	.7389	.7422	.7454	.7486	.7517	.7549
.7	.7580	.7611	.7642	.7673	.7704	.7734	.7764	.7794	.7823	.7852
.8	.7881	.7910	.7939	.7967	.7995	.8023	.8051	.8078	.8106	.8133
.9	.8159	.8186	.8212	.8238	.8264	.8289	.8315	.8340	.8365	.8389
1.0	.8413	.8438	.8461	.8485	.8508	.8531	.8554	.8577	.8599	.8621
1.1	.8643	.8665	.8686	.8708	.8729	.8749	.8770	.8790	.8810	.8830
1.2	.8849	.8869	.8888	.8907	.8925	.8944	.8962	.8980	.8997	.9015
1.3	.9032	.9049	.9066	.9082	.9099	.9115	.9131	.9147	.9162	.9177
1.4	.9192	.9207	.9222	.9236	.9251	.9265	.9279	.9292	.9306	.9319
1.5	.9332	.9345	.9357	.9370	.9382	.9394	.9406	.9418	.9429	.9441
1.6	.9452	.9463	.9474	.9484	.9495	.9505	.9515	.9525	.9535	.9545
1.7	.9554	.9564	.9573	.9582	.9591	.9599	.9608	.9616	.9625	.9633
1.8	.9641	.9649	.9656	.9664	.9671	.9678	.9686	.9693	.9699	.9706
1.9	.9713	.9719	.9726	.9732	.9738	.9744	.9750	.9756	.9761	.9767
2.0	.9772	.9778	.9783	.9788	.9793	.9798	.9803	.9808	.9812	.9817
2.1	.9821	.9826	.9830	.9834	.9838	.9842	.9846	.9850	.9854	.9857
2.2	.9861	.9864	.9868	.9871	.9875	.9878	.9881	.9884	.9887	.9890
2.3	.9893	.9896	.9898	.9901	.9904	.9906	.9909	.9911	.9913	.9916
2.4	.9918	.9920	.9922	.9925	.9927	.9929	.9931	.9932	.9934	.9936
2.5	.9938	.9940	.9941	.9943	.9945	.9946	.9948	.9949	.9951	.9952
2.6	.9953	.9955	.9956	.9957	.9959	.9960	.9961	.9962	.9963	.9964
2.7	.9965	.9966	.9967	.9968	.9969	.9970	.9971	.9972	.9973	.9974
2.8	.9974	.9975	.9976	.9977	.9977	.9978	.9979	.9979	.9980	.9981
2.9	.9981	.9982	.9982	.9983	.9984	.9984	.9985	.9985	.9986	.9986
3.0	.9987	.9987	.9987	.9988	.9988	.9989	.9989	.9989	.9990	.9990
3.1	.9990	.9991	.9991	.9991	.9992	.9992	.9992	.9992	.9993	.9993
3.2	.9993	.9993	.9994	.9994	.9994	.9994	.9994	.9995	.9995	.9995
3.3	.9995	.9995	.9995	.9996	.9996	.9996	.9996	.9996	.9996	.9997
3.4	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9997	.9998

Selected Percentiles										
Cumulative probability A :	.90	.95	.975	.98	.99	.995	.999			
$z(A)$:	1.282	1.645	1.960	2.054	2.326	2.576	3.090			

Figure A.7: Z-table (source: [KNNL05, Table B.1]).

- The following chunk explores very briefly some of R's capabilities for computing normal probabilities. Compare to Figure A.7. More later.

```
> ## cdf Phi(1.96) the hard way
> integrate(dnorm, lower=-Inf, upper=1.96, mean=0, sd=1)

0.975 with absolute error < 0.0000013

> ## cdf Phi(1.96) the easy way
> pnorm(1.96)

[1] 0.975

> ## Phi(-infinity) = 0 Phi(infinity) = 1
> pnorm(-Inf)

[1] 0

> pnorm(Inf)

[1] 1
```

- We will use normal random variables/distributions as models for observed data for much of this course.
- We will also use several other distributions, including some that are related to the normal distribution (χ^2 , t , F). For Bayesian analyses, we will consider (prior/posterior) distributions for the parameters of distributions (e.g., normal, t , gamma, inverse gamma (or scaled inverse χ^2), beta, Dirichlet, Wishart).
- Binomial and Poisson distributions will be used in INF 512 for logistic or Poisson regression, respectively

Definition A.16 (Support of an RV).

- *The support of an rv is the set of values for y where $f(y) > 0$, where f is a pdf, for a continuous rv, or pmf, for a discrete rv.*
- *We'll sometimes use S to denote support or \mathcal{X} for the support of a random variable X or \mathcal{Y} for the support of a random variable Y , etc.*

- **E.g.** What's the support of a normal random variable? $\text{binom}(n = 5, p = 0.2)$? $\text{binom}(1, 0.5)$? $\text{Pois}(\lambda)$?

A.7.4 Some Remarks

- **pmf/pdf Non-Negative, Summing/Integrating to 1.** $0 \leq f(y) < \infty$ and summing/integrating to 1 across the set of values of y for which $f(y) > 0$ (its 'support' set). If $0 < g(y) < \infty$ and $g(y)$ sums to some constant, $c > 0$, then we may create a valid pmf as $f(y) = g(y)/c$. (For discrete random variables (pmf), we must have $0 \leq f(y) \leq 1$.)
- **cdf Between 0 and 1 and Monotonic.** $0 \leq F(y) \leq 1$, (not necessarily strictly) **monotonic**, i.e., $y_1 < y_2$ implies $F(y_1) \leq F(y_2)$ and (avoiding limit notation) $F(-\infty) = 0$, $F(\infty) = 1$.
- **Between 0 and 1 (inclusive)!** We do not give a formal definition of probability and its properties, except to say that **probabilities are between 0 and 1**, inclusive; if you obtain a negative probability or a probability greater than 1, there's a mistake somewhere. Perhaps more as we go.

A.8 Characteristics of Random Variables

- **Models of Distributional Reality.** According to our previous discussion, above, we will use mathematical models (e.g., normal, binomial, Poisson) as approximations of distributions of quantities, which we refer to as random variables.
- **Summarizing Functions (of Parameters).** Often, we do not deal with an rv's (theoretical) entire distribution but with some summarizing—and easier to understand—property of the distribution, like the **mean, variance, or covariance**. These properties are typically encapsulated as (functions of usually unknown) **parameters**, e.g., μ or p for the normal or binomial distributions, introduced above.

Models of Summarizing Functions. We will specify models for the mean (regression function) and, perhaps, (co-)variance, and these models will be functions of parameters (and data) that we will estimate/infer about using observed values of random variables (data) from the (so-modeled) distributions.

A.8.1 Expected (Mean) Value

Definition A.17 (Expected (Mean) Value of a Discrete RV).

-

$$\mu(Y) = E(Y) = \sum_{y \in S} yP(Y = y) = \sum_{y \in S} yf(y),$$

where S is the support of the rv Y .

- *The mean is a measure of center or central tendency of an rv in the sense of being the balance point of the probability masses of the pmf.*
- *Appealing for unimodal and symmetric distributions.*

- A **weighted average** of y values with weights given by the pmf, $f(y)$.
- Use of the **expectation (mean) operation notation**, E , is common.

Example A.5 (Mean of Discrete Uniform RV).

$$f(y) = \begin{cases} \frac{1}{3} & y \in \overbrace{\{1, 2, 3\}}^S \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} E(Y) &= \sum_{y \in \{1, 2, 3\}} y f(y) = 1(1/3) + 2(1/3) + 3(1/3) \\ &= 1/3 \sum_{y=1}^3 y = \frac{6}{3} = 2 \end{aligned}$$

Example A.6 (Mean of a Bernoulli (i.e., $\text{bin}(n = 1, p)$)).

$$\begin{aligned} E(Y) &= \sum_{y \in \{0, 1\}} y f(y) = 0f(0) + 1f(1) \\ &= 0p^0(1-p)^{1-0} + 1p^1(1-p)^{1-1} = 0 + 1p = p \end{aligned}$$

Definition A.18 (Expected (Mean) Value of a Continuous RV).

$$E(Y) = \int_{-\infty}^{\infty} y f(y) dy$$

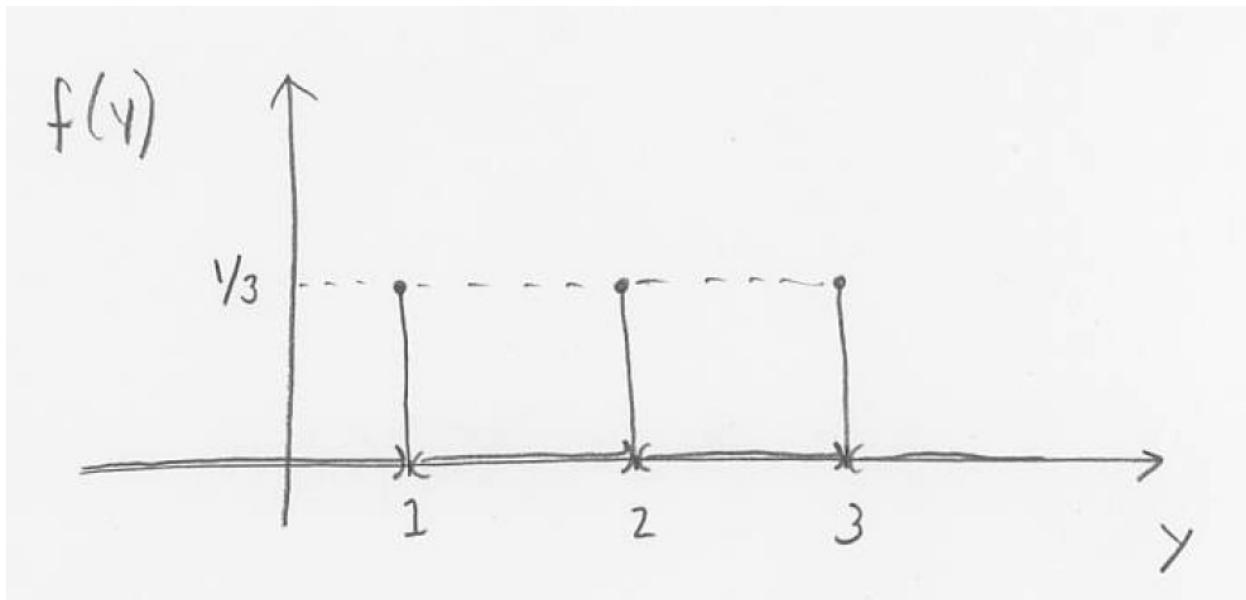


Figure A.8: Discrete uniform pmf supported on 1, 2, 3.

Example A.7 (Mean of $N(\mu, \sigma^2)$). If $Y \sim N(\mu, \sigma^2)$, then

$$E(Y) = \mu$$

(details omitted).

Remark A.2 (Linearity Property of the Expectation Operator).

- Let a , b , and c be arbitrary constants.
- Let X and Y be arbitrary rvs whose expectations (expected values, means) exist.
- Then

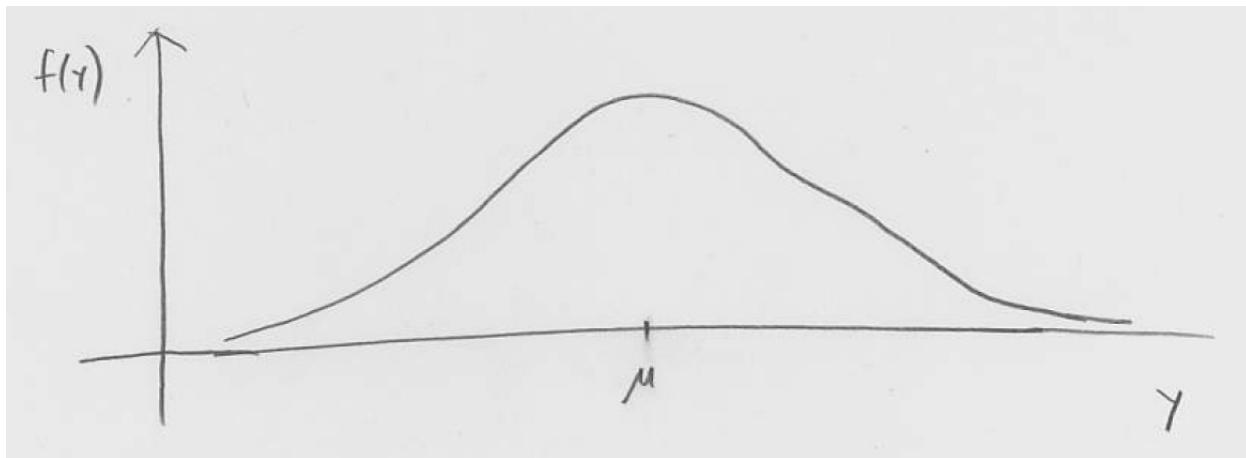


Figure A.9: pdf is balanced on mean (μ).

$$\begin{aligned} E(a + bX + cY) &= E(a) + E(bX) + E(cY) \\ &= a + bE(X) + cE(Y) \end{aligned}$$

- In words, “the expectation (mean) of the linear combination is the linear combination of the expectations (means).”
- This property follows from properties of (convergent) sums/series (for discrete rvs) or integrals (for continuous rvs). Details omitted.
- This is a particular example of a linear combination of rvs with coefficients a , b , and c . (1 (multiplying a) may be thought of as a degenerate rv). We will talk a bit more about linear combinations later in the course when comparing parameters in regression or ANOVA models.

Example A.8 (Mean of a Difference).

- Let $X \sim N(\mu_X, \sigma_X^2)$ $Y \sim N(\mu_Y, \sigma_Y^2)$.

- Then, applying the above linearity property,

$$E(X - Y) = \mu_X - \mu_Y.$$

- Note $a = 0$, $b = 1$ and $c = -1$ in the linearity property just presented, and we have introduced subscripts on our μ notation for expectations.
- “The mean of the difference is the difference of the means.”

Example A.9 (Additive Error Model).

- Let $\epsilon \sim N(0, \sigma^2)$ and $Y = \mu + \epsilon$ where μ and σ^2 are some constants.
- Then,

$$E(Y) = \mu.$$

- Note: let $a = \mu$, $b = 1$, $X = \epsilon$, $c = 0$.

A.8.2 Variance Operator

Definition A.19 (Variance of an RV).

-

$$\begin{aligned} \text{Var}(Y) &= E(Y - E(Y))^2 \\ &= E(Y^2) - (E(Y))^2 \end{aligned}$$

- The variance of an rv is a **measure of the spread or dispersion** of the possible values of an rv.
- May be thought of as the (weighted) **average squared deviation** of a rv from its mean.

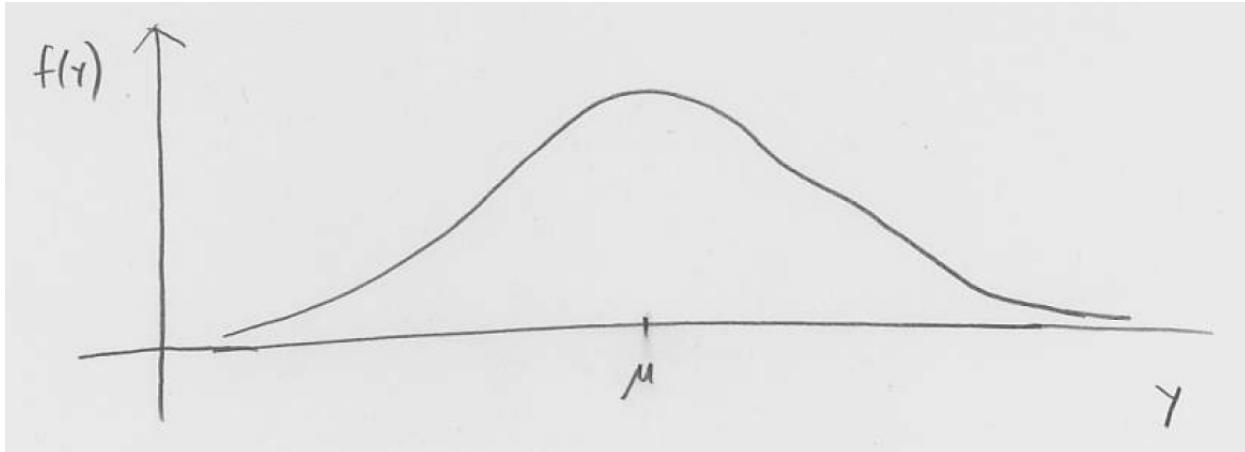
Example A.10 (Variance of $N(\mu, \sigma^2)$).

- If $Y \sim N(\mu, \sigma^2)$,

- then

$$\text{Var}(Y) = \int_{-\infty}^{\infty} (y - E(Y))^2 f(y) dy = \sigma^2$$

(details omitted).



Example A.11 (Variance of A Bernoulli (i.e., $\text{binom}(n = 1, p)$)).

$$\begin{aligned}
 \text{Var}(Y) &= \sum_{y=0}^1 (y - p)^2 p^y (1 - p)^{1-y} \\
 &= (0 - p)^2 p^0 (1 - p)^{1-0} + (1 - p)^2 p^1 (1 - p)^{1-1} \\
 &= p^2 (1 - p) + (1 - p)^2 p \\
 &= p(1 - p)(p + (1 - p)) \\
 &= p(1 - p)
 \end{aligned}$$

Definition A.20 (Standard Deviation of an RV).

$$SD(Y) = \sqrt{Var(Y)}$$

Useful Property of Variance Operator

$$Var(a + bY) = b^2 Var(Y)$$

Example A.12. Let $Y \sim N(\mu, \sigma^2)$ and $Z = \frac{Y-\mu}{\sigma}$

$$\begin{aligned} Var(Z) &= Var\left(-\frac{\mu}{\sigma} + \frac{1}{\sigma}Y\right) \\ &= \frac{1}{\sigma^2} Var(Y) \\ &= \frac{\sigma^2}{\sigma^2} = 1 \end{aligned}$$

Note: $a = -\frac{\mu}{\sigma}$ $b = \frac{1}{\sigma}$.

Example A.13 (Additive Error Model (again)).

Let $\epsilon \sim N(0, \sigma^2)$ and $Y = \mu + \epsilon$ where μ and σ^2 are some constants.

Then,

$$Var(Y) = \sigma^2$$

Note, $a = \mu$, $b = 1$ and ϵ is the originally specified random variable in the statement of the property of the variance operator. (Saying $Y = \epsilon$ would be confusing!)

A.9 Random Vectors

- Here, we treat only a few fundamental concepts involving groups of random variables collected into a vector.
- In Appendix B, we will treat matrices and vectors formally before giving more details on random vectors.

A.9.1 Covariance Operator & Its Properties

- Covariance is a property of two random variables.

Definition A.21 (Covariance).

$$\begin{aligned} \text{Cov}(X, Y) &= E(X - E(X))(Y - E(Y)) \\ &= E(XY) - E(X)E(Y) \end{aligned}$$

Example A.14 (Covariance of Bivariate Normal).

$$\begin{aligned} f(x, y) &= \frac{1}{2\pi\sqrt{\sigma_X^2\sigma_Y^2(1-\rho)^2}} \times \\ &\quad \exp\left\{-\frac{1}{2(1-\rho^2)}\left[\left(\frac{x-\mu_X}{\sigma_X}\right)^2 + \left(\frac{y-\mu_Y}{\sigma_Y}\right)^2\right.\right. \\ &\quad \left.\left.-2\rho\left(\frac{x-\mu_X}{\sigma_X}\right)\left(\frac{y-\mu_Y}{\sigma_Y}\right)\right]\right\} \end{aligned}$$

$$\begin{aligned} \text{Cov}(X, Y) &= \int \int (x - \mu_X)(y - \mu_Y) f(x, y) dx dy \\ &= \sigma_X \sigma_Y \rho \end{aligned}$$

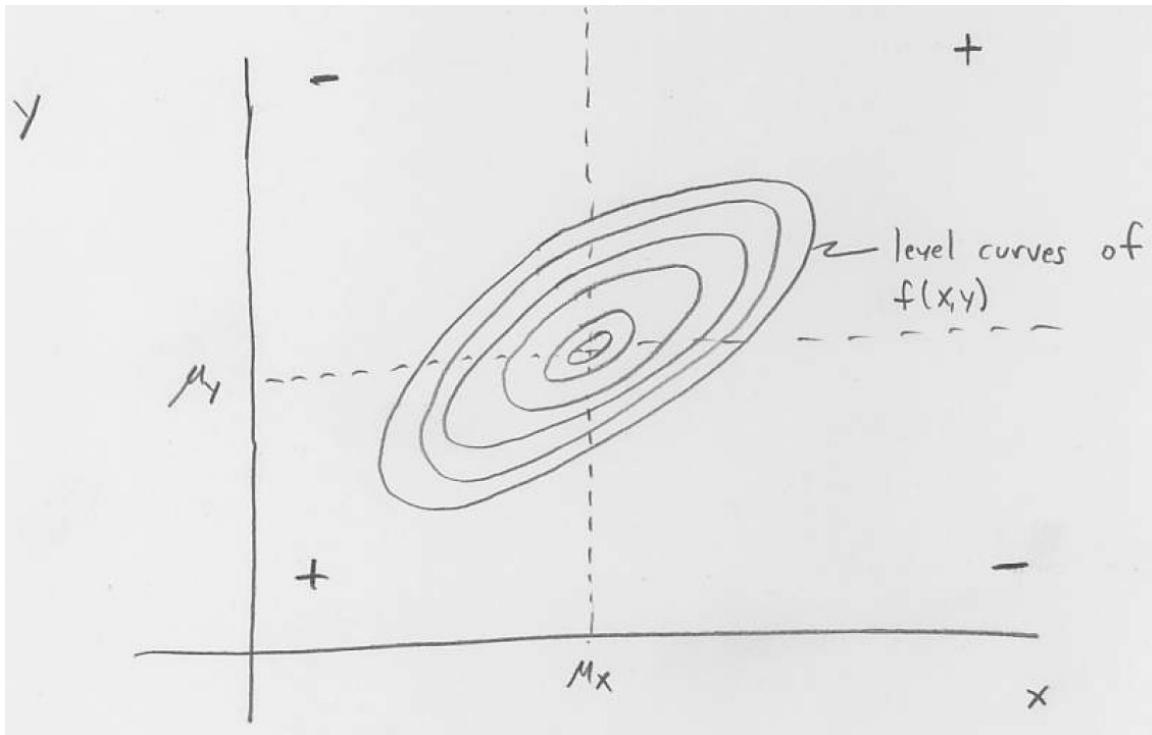


Figure A.10: Sketch to illustrate covariance.

Remark A.3 (Properties of Variance & Covariance Operators).

$$\text{Cov}(Y, Y) = \text{Var}(Y) \quad (\text{right}?!)$$

$$\text{Var}(X + Y) = \text{Var}(X) + \text{Var}(Y) + 2\text{Cov}(X, Y)$$

$$\text{Var}(aX + bY) = a^2\text{Var}(X) + b^2\text{Var}(Y) + 2ab\text{Cov}(X, Y)$$

$$\begin{aligned}
 \text{Var} \left(\sum_{i=1}^n c_i Y_i \right) &= \sum_{i=1}^n \sum_{j=1}^n c_i c_j \text{Cov}(Y_i, Y_j) \\
 &= \sum_{i=1}^n c_i^2 \text{Var}(Y_i) + 2 \sum_{i < j} c_i c_j \text{Cov}(Y_i, Y_j)
 \end{aligned}$$

Take $n = 2$ & $c_1 = c_2 = 1$ to get result for $\text{Var}(X + Y)$ above.

$$\text{Var}(a) = 0$$

$$\text{Cov}(a, Y) = 0$$

Definition A.22 (Correlation). *The correlation between two variables is the covariance between the variables' standardized versions. It's unitless.*

$$\begin{aligned}
 \text{Cor}(X, Y) &= \text{Cov} \left(\frac{X - E(X)}{\sqrt{\text{Var}(X)}}, \frac{Y - E(Y)}{\sqrt{\text{Var}(Y)}} \right) \\
 &= \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X) \text{Var}(Y)}}
 \end{aligned}$$

Example A.15 (Correlation of Bivariate Normal (cont'd)).

$$\begin{aligned}
 f(x, y) &= \frac{1}{2\pi\sqrt{\sigma_X^2\sigma_Y^2(1-\rho)^2}} \times \\
 &\quad \exp \left\{ -\frac{1}{2(1-\rho^2)} \left[\left(\frac{x - \mu_X}{\sigma_X} \right)^2 + \left(\frac{y - \mu_Y}{\sigma_Y} \right)^2 \right. \right. \\
 &\quad \left. \left. - 2\rho \left(\frac{x - \mu_X}{\sigma_X} \right) \left(\frac{y - \mu_Y}{\sigma_Y} \right) \right] \right\}
 \end{aligned}$$

$$\text{Cor}(X, Y) = \rho$$

(details omitted)

A.9.2 Independence

- We will have more to say about independence and joint distributions after covering vectors more formally in Appendix B. For now, we give a brief look at independence.
- X, Y are said to be independent if their joint distribution (e.g. cdf, pdf, pmf) factors into the product of marginal distributions, i.e., (using pdf/pmf notation) if

$$f(x, y) = f(x)f(y).$$

(Not same f of course. Recycling notation!).

- Less precisely but more intuitively, X and Y are independent if the values of X tell us nothing about the values of Y , and vice-versa.
- Always, if X and Y are independent, then $\text{Cov}(X, Y) = 0$.
- But, if $\text{Cov}(X, Y) = 0$, then X and Y are not necessarily independent, normality being a (the?) notable exception: if X and Y are normally distributed, then $\text{Cov}(X, Y) = 0$ implies $f(x, y) = f(x)f(y)$ (look at the bivariate normal pdf nearby).

Example A.16 (Independent Coin Tosses). Let X and Y be the result of two coin tosses. In most circumstances, it would be reasonable to think that the result of one toss has nothing to do with the result of another. Thus,

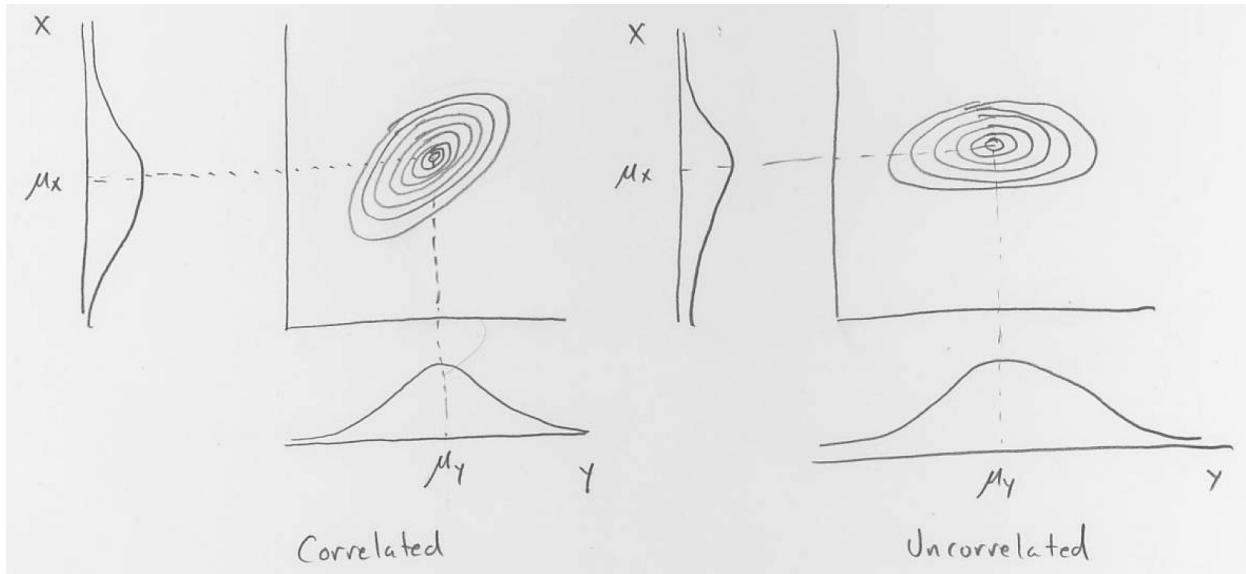


Figure A.11: Sketch of marginal and joint distributions.

independence seems eminently reasonable. Assuming a Bernoulli ($\text{binom}(n = 1, p)$) for each toss we have

$$f(x, y) = f(x)f(y) = p_X^x(1 - p_X)^{1-x}p_Y^y(1 - p_Y)^{1-y}.$$

It may be reasonable to assume $p_X = p_Y = 0.5$.

- Independence is typically an **assumption** (hopefully reasonable) made to facilitate **model building** (rather than something to be checked after a model is already specified).
- It is often easier to specify $f(x)$ and $f(y)$ than to specify $f(x, y)$ directly.

A.10 Linear Combinations of RVs

- We introduced linear combinations of random variables, above, when discussing the properties of the expectation and variance operators.
- We will return to linear combinations of random vectors, using matrix notation, in a subsequent chapter of notes.
- For now, we continue with a few more results on linear combinations using non-matrix notation. (As you will see, the same expressions using matrices are much simpler.)

- If Y_i are rvs and c_i are constants, $i = 1, \dots, n$, then $\sum_{i=1}^n c_i Y_i$ is a **linear combination** of the Y_i with **coefficients** c_i .
- $\sum_{i=1}^n c_i Y_i$ has mean

$$\mathbb{E} \left(\sum_{i=1}^n c_i Y_i \right) = \sum_{i=1}^n c_i \mathbb{E}(Y_i).$$

In words, “**the mean of linear combination is the linear combination of means.**”

- Furthermore, if the Y_i are uncorrelated (e.g., independent), then

$$\text{Var} \left(\sum_{i=1}^n c_i Y_i \right) = \sum_{i=1}^n c_i^2 \text{Var}(Y_i).$$

In words, “**the variance of a linear combination is (almost) the linear combination of variances (with coefficients now squared), if the variables are uncorrelated.**”

- If, in addition, the Y_i are normally distributed, then so is the linear combination:

$$\sum_{i=1}^n c_i Y_i \sim N\left(\sum_{i=1}^n c_i E(Y_i), \sum_{i=1}^n c_i^2 \text{Var}(Y_i)\right)$$

- We will use similar results on linear combinations a lot, typically in matrix form, which is somehow easier. (More elsewhere.)

Example A.17 (Linear Function of Normal RV). If $Y \sim N(\mu, \sigma^2)$, then $Z = \frac{Y-\mu}{\sigma} \sim N(0, 1)$, by the previous results for linear combinations. ($c_1 = -\frac{\mu}{\sigma}$, $Y_1 = 1$ (degenerate at 1), $c_2 = \frac{1}{\sigma}$ and $Y_2 = Y$)

Example A.18 (Linear Function of Normal RV (again)). If $Z \sim N(0, 1)$, then $Y = \mu + \sigma Z \sim N(\mu, \sigma^2)$. ($c_1 = \mu$, $Y_1 = 1$ (degenerate at 1), $c_2 = \sigma$ and $Y_2 = Z$)

Example A.19 (Additive Error Model (again)).

Let $\epsilon \sim N(0, \sigma^2)$ and $Y = \mu + \epsilon$ where μ and σ^2 are some constants.

Then,

$$Y \sim N(\mu, \sigma^2).$$

($c_1 = \mu$, $Y_1 = 1$ (degenerate), $c_2 = 1$ and $Y_2 = \epsilon$)

A.11 Central Limit Theorem

Several versions of the CLT exist. We give one here.

Theorem A.1 (A Central Limit Theorem (CLT)). *If Y_1, Y_2, \dots, Y_n are independent rvs from the same distribution with same mean $E(Y_i) = \mu$ and $\text{Var}(Y_i) = \sigma^2 < \infty$, then*

$$\bar{Y}_n \stackrel{\text{"dot means approx"} \atop \sim}{\cdot} N(\mu, \sigma^2/n)$$

and the approximation improves with increasing n .

- If $Y_i \stackrel{\text{ind}}{\sim} N(\mu, \sigma^2)$, then $\bar{Y}_n \sim N(\mu, \sigma^2/n)$, exactly, for any positive integer n .

A.12 pdqr Functions in R

Typically, for each random variable or distribution implemented in R, there are four associated functions that I call the ‘p’ ‘d’ ‘q’ and ‘r’ functions. (Somehow, ‘pdqr’ rolls off my tongue quicker (pdq? ha) than the alphabetical ‘dpqr’, as some people may refer to these functions.)

- p functions essentially compute a random variable’s cdf (cumulative (probability) distribution function), with some additional options, e.g., **pnorm**, **pbinom**, seen above. Given a value of a random variable, y , p functions give

$$F(y) = P(Y \leq y)$$

(with some added functionality). We will also (at least indirectly) use the ‘cdf’ or ‘p’ functions for t and F random variables, **pt** and **pf**.

- **d** functions compute a random variable's pdf (probability density function), for a **continuous** random variables, e.g., **dnorm**, used to plot the pdf of a normal random variable, above ('bell curve'). For **discrete** random variables the **d** functions give the pmf (probability mass function; sorry, no **d** in pmf!), e.g., **pbinom** used above to plot a binomial pmf. Given a value of a random variable, y , **d** functions give

$$f(y) = P(Y = y)$$

for **discrete** random variables, but $f(y)$ is the pdf (not $P(Y = y)!$) for **continuous** random variables as discussed above (e.g., height of the normal 'bell curve' at y , $f(y)$).

- **q** functions are **quantile** functions or **inverse cdf** functions. Instead of giving a cumulative probability

$$F(y) = P(Y \leq y)$$

for a given value, y , quantile functions go the other way, giving the value

$$y = F^{-1}(p)$$

such that

$$p = P(Y \leq y).$$

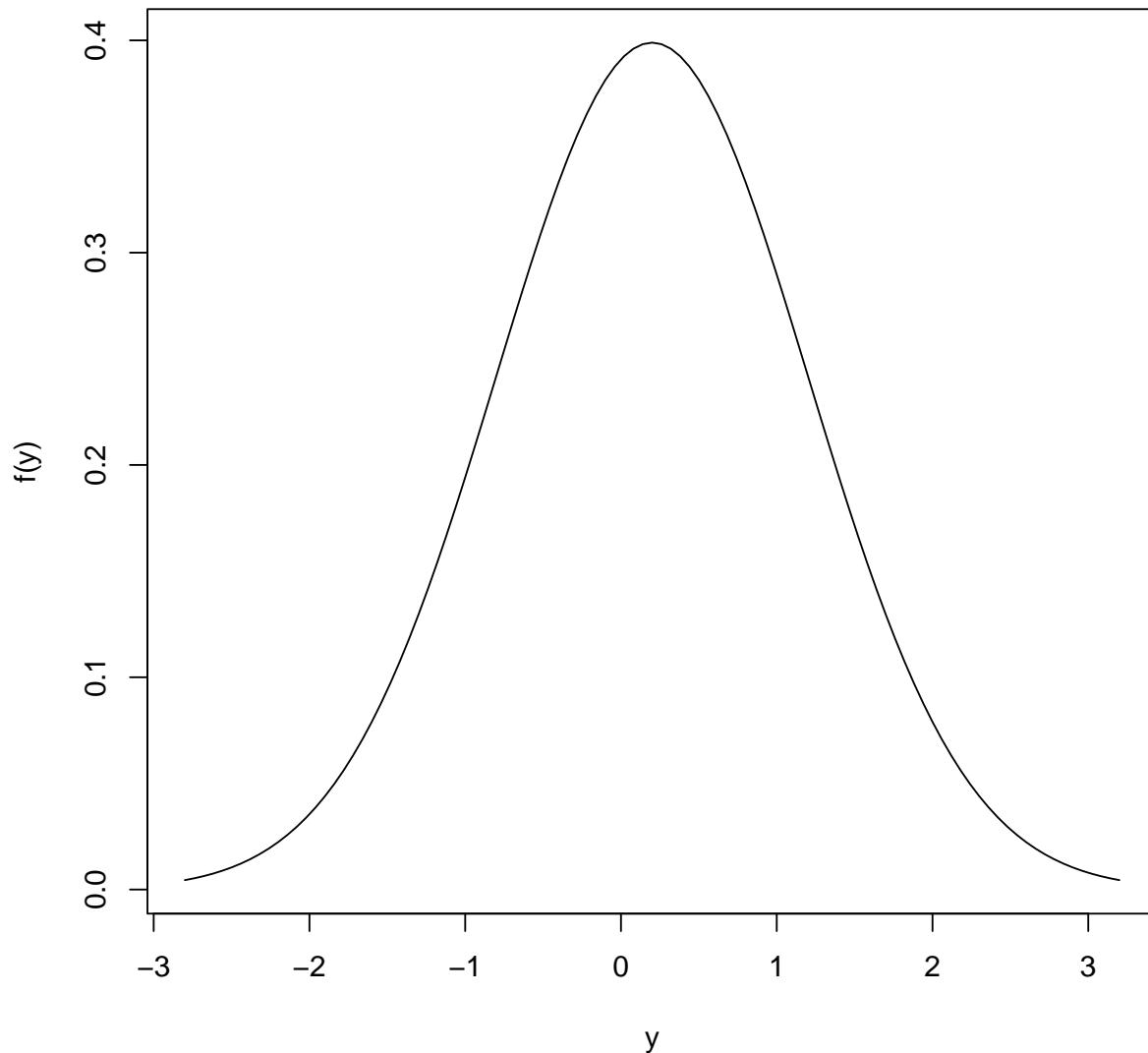
We will (indirectly at least) use quantile functions to give multipliers in the construction **confidence intervals** or for obtaining 'critical values' of random variables beyond which we may reject a hypothesis in a **hypothesis test**.

- **r** functions generate (pseudo-)random values from a variable's distribution.

A.12.1 Example

For example, we illustrate **pnorm**, **dnorm**, **qnorm** and **rnorm** in the context of a typical (simulated) ‘stat 101 z-test’ (type I error probability $\alpha = 0.05$) and 95% confidence interval. (This should look familiar. Some formal details omitted).

```
> ## Mathematical model of the probability distribution of some measured
> ## quantity (mathematical abstraction of desired distribution):
> m<- 0.2; sd<- 1 ## <-- parameters of the distribution typically unknown
> curve(dnorm(x, mean=m, sd=sd), from=-3+m, to=3+m,
+         xlab="y", ylab="f(y)",
+         main="(Super-)Population Model")## plot the d function
```

(Super-)Population Model

```
> ## A faculty sends her student to randomly gather n=30 such quantities:  
> set.seed(8675309) ## <-- only scratching the surface of reproducibility  
> y<- rnorm(n=30, mean=m, sd=sd) ## r function  
>  
> ## Your advisor tells you to compute the average to estimate the mean...  
> (my<- mean(y))  
  
[1] 0.39426
```

```
> ##...and to compute a 95% two-sided confidence interval for the mean...
> (zcrit<- qnorm(1-0.05/2))          ## q function P(Type 1 error) = 0.05
[1] 1.96

> mean(y) + c(-1,1) * zcrit * 1/sqrt(30)  ## 95% conf. interval
[1] 0.03642 0.75210
```

```
> ## ...and asks you if the average is somehow 'large' to have come from
> ## a distribution with mean zero (null hypothesis). After, properly
> ## standardizing your average to a 'z-value', you may compare it to
> ## zcrit, which is larger (in a positive or negative sense) than
> ## 100*(1-0.05/2)% = 95% of such quantities (with mean zero and sd 1), or you
> ## may simply compute the probability of having observed a z-value as
> ## extreme (positive or negative) as the one you just computed from
> ## your data (in several different ways).
> (z<- (mean(y) - 0) / (1/sqrt(30)))

[1] 2.1594

> pnorm(-abs(z)) +                      ## p function
+      pnorm(abs(z), lower.tail=FALSE)
[1] 0.030815

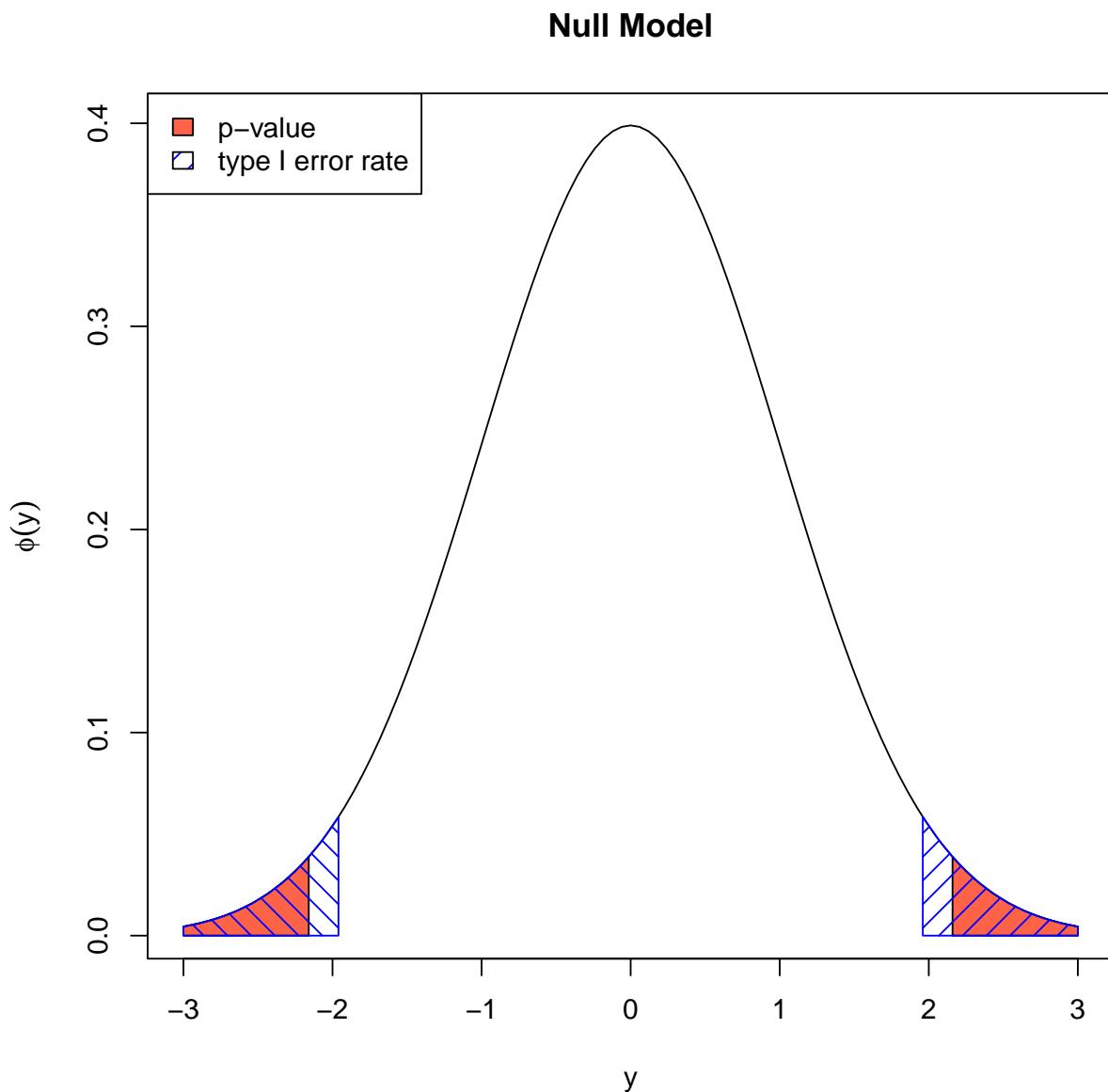
> 2*pnorm(abs(z), lower.tail=FALSE)
[1] 0.030815

> 2*pnorm(-abs(z))
[1] 0.030815

> ## etc.
> 2*(1 - pnorm(abs(z)))
[1] 0.030815
```

```
> ## You compare the extremeness of zcrit and your z, as well as 0.05 and
> ## your p-value (under the null of zero mean), illustrating graphically
> ## the test and your budding R expertise to your advisor.
```

```
> curve(dnorm(x, mean=0, sd=sd), from=-3, to=3,
+         xlab="y", ylab=expression(phi(y)),
+         main="Null Model")## null distribution
> zseq<- seq(abs(z), to=3, length=10)
> dseq<- dnorm(zseq)                      ## d function
> polygon(x=c(abs(z),zseq,3), y=c(0,dseq,0), col="tomato")
> polygon(x=-c(abs(z),zseq,3), y=c(0,dseq,0), col="tomato")
> zseq<- seq(zcrit, to=3, length=10)
> dseq<- dnorm(zseq)                      ## d function
> polygon(x=c(zcrit,zseq,3), y=c(0,dseq,0), col="blue", density=10)
> polygon(x=-c(zcrit,zseq,3), y=c(0,dseq,0), col="blue", density=10,
+           angle=135)
> legend("topleft",legend=c("p-value", "type I error rate"),
+         fill=c("tomato","blue"), density=c(NA,10))
```



Appendix B

Matrices & Vectors

Contents

B.1	Notation, Dimension, Rows, Columns, Elements	589
B.2	Matrix Arithmetic	593
B.2.1	Addition/Subtraction	593
B.2.2	Multiply a Matrix by a Scalar	595
B.2.3	Matrix Multiplication	596
B.2.4	Matrix Transpose	598
B.2.5	Special Matrices	601
B.2.6	Linear Dependence & Rank	606
B.3	Combining Things: Random Vectors and Matrices	618
B.3.1	Expectation of a Random Vector/Matrix	619
B.3.2	Variance(–Covariance) Matrix	620
B.3.3	Linearity of Expectation Operator (just as in scalar case)	620
B.3.4	Variance(–Covariance) of $\mathbf{a} + \mathbf{BY}$	621
B.3.5	Distribution of Linear Function of Normal RV	621
B.4	Normal and χ^2 Results	625
B.5	t and F Distribution Results	631
B.6	Joint, Marginal & Conditional Distributions	634
B.7	Conditional Distribution Model Specification	644

Main Objectives:

- Learn basic matrix operations, including matrix addition, matrix multiplication, scalar multiplication of a matrix, transpose, inverse, determinant and trace.
 - Familiarize ourselves with some of R's basic matrix and vector functionality.
 - Extend previous results on random variables (Appendix A) to random matrices and random vectors (mostly just vectors).
 - Preview common matrices and vectors used in linear models so that these are familiar when we discuss and perform regression and ANOVA computations, special cases of the linear model.
 - **NOTE:** Incidentally, matrices are special cases of multi-dimensional arrays of numbers, i.e., **tensors**, which are critical to a good understanding of **neural nets and deep learning**; more in INF 504 and other courses.
-
-

Reading:

- Much of the material here is based on [KNNL05, Sec. 5.1 – 5.11].

 \mathcal{R} **Definition B.1** (Matrix). *A two-dimensional array of numbers.***Definition B.2** (Matrix Size and Dimensions).

- *The size of a matrix is given by its two dimensions, often called “rows” and “columns.”*
- *Similar to the size of a rectangular room given by its vertical and horizontal dimensions.*
- *An $r \times c$ (size) matrix has r rows and c columns.*
- *(NOTE: The deep learning (neural net) literature has slightly ‘tweaked’ the jargon for tensors, but it’s nothing you can’t get over very quickly. Again, more in INF 504 and other courses.)*

B.1 Notation, Dimension, Rows, Columns, Elements

Example B.1 (A 2 by 2 Matrix).

$$\mathbf{A}_{2 \times 2} = \begin{bmatrix} 1 & 7 \\ 4 & 3 \end{bmatrix}$$

Example B.2 (A 2 by 3 Matrix).

$$\mathbf{B}_{2 \times 3} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix}$$

OR

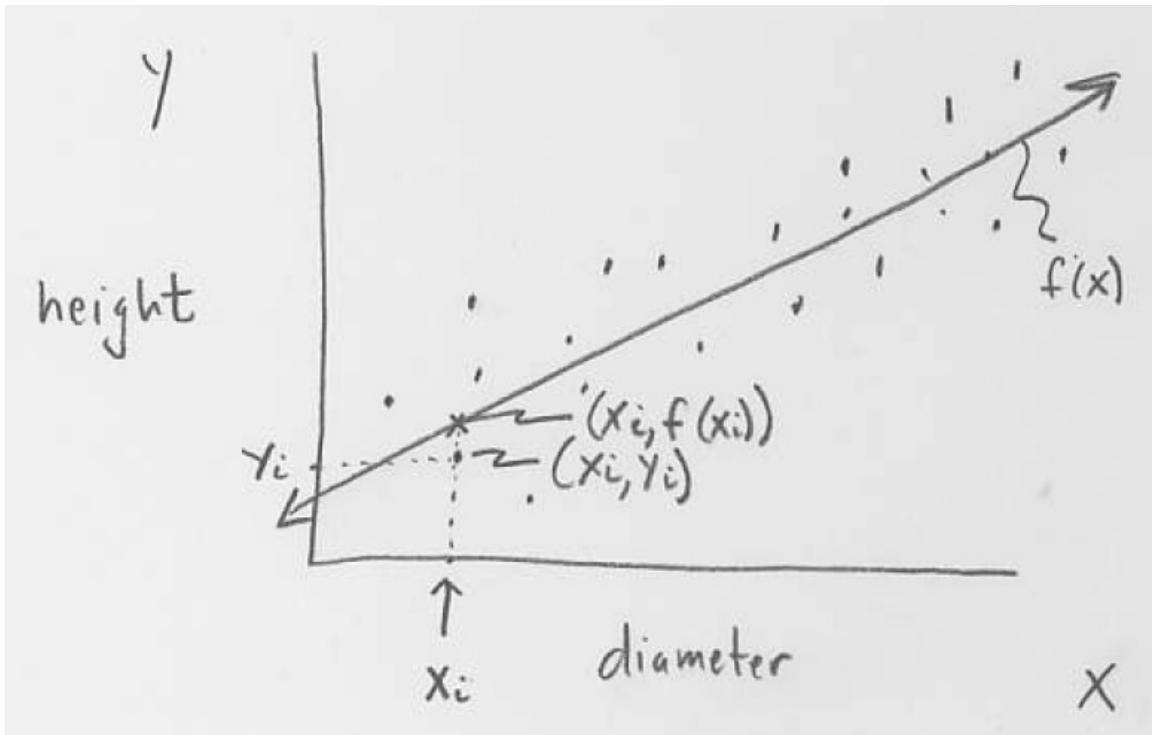
$$\mathbf{B}_{2 \times 3} = [b_{ij}] \quad i = 1, \dots, r, j = 1, \dots, c, r = 2, c = 3$$

Example B.3 (An r by c Matrix).

$$\mathbf{A}_{r \times c} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1c} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2c} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{ic} \\ \vdots & \ddots & \cdots & \vdots & \ddots & \vdots \\ a_{r1} & a_{r2} & \cdots & a_{rj} & \cdots & a_{rc} \end{bmatrix}$$

Example B.4 (Simple Linear Regression).

(tree diameter, tree height): (x_i, y_i) , $i = 1, \dots, n$



Data Model Assumptions (matrices main focus here!):

$$Y_i = \mu_i + \varepsilon_i \quad \text{where } \mu_i = \beta_0 + \beta_1 x_i \\ \varepsilon_i \stackrel{\text{ind}}{\sim} N(0, \sigma^2), \quad i = 1, \dots, n,$$

or, equivalently,

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad \varepsilon_i \stackrel{\text{ind}}{\sim} N(0, \sigma^2)$$

Typical matrices for such a model:

$$\mathbf{Y}_{n \times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \quad \mathbf{X}_{n \times 2} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \quad \boldsymbol{\beta}_{2 \times 1} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \quad \boldsymbol{\varepsilon}_{n \times 1} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

We are on our way to matrix formulation of ANOVA/regression (linear) models:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon};$$

see [Far14, §2.5 item 2].

- The following chunk produces an “X matrix” (from simulated data) that we may use in regression analysis if we were prepared.

```
> ## Fake tree data (no trees harmed in the making of this data)
> xdiam<- c(20,15,21,34,28,19,22,24,25,18)
> n<- length(xdiam)
> yht <- 4.5 + 0.8 * xdiam + rnorm(n=n, sd=0.5)
>
> ## We might like to keep our data in a data frame:
> tree.df<- cbind.data.frame(diam=xdiam, height=yht)
>
> ## The X matrix that would be used in regression:
> (X<- model.matrix(height ~ diam, data=tree.df))

  (Intercept) diam
1             1   20
2             1   15
3             1   21
4             1   34
5             1   28
6             1   19
7             1   22
8             1   24
9             1   25
10            1   18
attr(,"assign")
[1] 0 1

> is.matrix(X)

[1] TRUE

> dim(X) ## or size of the X matrix

[1] 10  2
```

B.2 Matrix Arithmetic

Matrix/vector operations were implicit, above. Now we make them explicit.

B.2.1 Addition/Subtraction

To add/subtract matrices, add/subtract values in the same row/column—easy!

Example B.5 (Adding Matrices).

$$\mathbf{A} = \begin{bmatrix} 1 & 4 \\ 2 & 7 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 2 & -3 \\ 4 & 1 \end{bmatrix}$$

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} 1+2 & 4+(-3) \\ 2+4 & 7+1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 6 & 8 \end{bmatrix}$$

Example B.6 (Adding Matrices (cont'd)).

$$\mathbf{A}_{r \times c} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1c} \\ a_{21} & a_{22} & \cdots & a_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r1} & a_{r2} & \cdots & a_{rc} \end{bmatrix} \quad \mathbf{B}_{r \times c} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1c} \\ b_{21} & b_{22} & \cdots & b_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ b_{r1} & b_{r2} & \cdots & b_{rc} \end{bmatrix}$$

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1c} + b_{1c} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2c} + b_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ a_{r1} + b_{r1} & a_{r2} + b_{r2} & \cdots & a_{rc} + b_{rc} \end{bmatrix},$$

or

$$[a_{ij}]_{rc} + [b_{ij}]_{rc} = [a_{ij} + b_{ij}]_{rc}$$

Definition B.3 (Conformable for Addition).

- *Matrices are said to be conformable for addition if they have the same size.*
- *Matrix addition is not defined for matrices of different sizes.*

- The following chunk illustrates some matrix functionality in R.

```
> (A<- matrix(c(1,2,4,7),nrow=2, ncol=2)); dim(A); nrow(A); ncol(A)
      [,1] [,2]
[1,]    1    4
[2,]    2    7
[1] 2 2
[1] 2
[1] 2

> (B<- matrix(c(2,4,-3,1),nrow=2, ncol=2)); dim(B); nrow(B); ncol(B)
      [,1] [,2]
[1,]    2   -3
[2,]    4    1
[1] 2 2
[1] 2
[1] 2

> (C<- A + B); dim(C)
      [,1] [,2]
[1,]    3    1
[2,]    6    8
[1] 2 2

> is.matrix(C)
```

```
[1] TRUE

> (D<- matrix(c(1,2,3),nrow=3)); dim(D)

[,1]
[1,]    1
[2,]    2
[3,]    3
[1] 3 1

> A+D

Error in A + D: non-conformable arrays
```

B.2.2 Multiply a Matrix by a Scalar

To multiply a matrix by a scalar (single number), multiply each element of the matrix by the scalar—easy!

Example B.7 (Scalar Multiplication).

$$a\mathbf{B}_{rc} = a [b_{ij}]_{rc} = [ab_{ij}]_{rc} = \begin{bmatrix} ab_{11} & ab_{12} & \cdots & ab_{1c} \\ ab_{21} & ab_{22} & \cdots & ab_{2c} \\ \vdots & \vdots & \ddots & \vdots \\ ab_{r1} & ab_{r2} & \cdots & ab_{rc} \end{bmatrix}$$

Example B.8 (Scalar Multiplication (cont'd)).

$$-2 \begin{bmatrix} 1 & 7 \\ 6 & -3 \end{bmatrix} = \begin{bmatrix} -2(1) & -2(7) \\ -2(6) & -2(-3) \end{bmatrix} = \begin{bmatrix} -2 & -14 \\ -12 & 6 \end{bmatrix}$$

B.2.3 Matrix Multiplication

First, some specific examples to illustrate how to multiply matrices.

Example B.9 (Matrix Multiplication).

$$\begin{aligned}
 & \left[\begin{array}{ccc} 1 & 4 & 7 \\ 6 & 3 & 2 \end{array} \right]_{2 \times 3} \times \left[\begin{array}{cc} 1 & 2 \\ 6 & 7 \\ 3 & 4 \end{array} \right]_{3 \times 2} \\
 &= \left[\begin{array}{cc} 1 \times 1 + 4 \times 6 + 7 \times 3 & 1 \times 2 + 4 \times 7 + 7 \times 4 \\ 6 \times 1 + 3 \times 6 + 2 \times 3 & 6 \times 2 + 3 \times 7 + 2 \times 4 \end{array} \right]_{2 \times 2} \\
 &= \left[\begin{array}{cc} 46 & 48 \\ 30 & 41 \end{array} \right]_{2 \times 2}
 \end{aligned}$$

- **Linear Combinations.** We will discuss linear combinations shortly. For now, note that you may see matrix multiplication as linear combinations of columns of the first matrix (left multiplicand) with coefficients given in the second matrix (right multiplicand), or, as linear combinations of the rows of the second with coefficients from the first. We (I) mostly view from the first perspective, linear combinations of columns.
- Corresponding matrix functionality in R follows.

```

> A<- matrix(c(1,6,7,-3),nrow=2, ncol=2)
> ## * is for scalar multiplication,
> ## %*% is for matrix multiplication (IMPORTANT!)
> (C<- -2 * A)

 [,1] [,2]
[1,]   -2   -14
[2,]  -12      6

```

```

> A<- matrix(c(1,6,4,3,7,2),nrow=2, ncol=3)
> B<- matrix(c(1,6,3,2,7,4),nrow=3,ncol=2)
> (C<- A%*%B)

      [,1] [,2]
[1,]   46   58
[2,]   30   41

> ## %*% gives dot product, too!
> A[2,] %*% B[,2]

      [,1]
[1,]   41

```

- More generally and abstractly, let

$$\mathbf{C}_{r \times c} = \mathbf{A}_{r \times l} \mathbf{B}_{l \times c}.$$

Then,

$$[c_{ij}]_{r \times c} = \left[\sum_{k=1}^l a_{ik} b_{kj} \right]_{r \times c}.$$

- Notice the resulting entry in row i and column j is the **dot product** (or **inner product, scalar product**) of row vector i of the left factor and column vector j of the right factor. Formal definition?

Definition B.4 (Conformable for Multiplication).

- Two matrices are said to be **conformable** for multiplication if the left matrix factor in the product (left multiplicand) has the same number of columns ($c = l$, 2nd dimension) as the right factor has rows ($r = l$, 1st dimension), else matrix multiplication is not defined.

```
> A%*%D
      [,1]
[1,]   30
[2,]   18
> D%*%A
Error in D %*% A: non-conformable arguments
```

B.2.4 Matrix Transpose

Example B.10 (Matrix Transpose).

$$\mathbf{A} = \begin{bmatrix} 1 & 4 & 7 \\ 6 & 3 & 2 \end{bmatrix}_{2 \times 3}$$

$$\mathbf{A}' = \begin{bmatrix} 1 & 6 \\ 4 & 3 \\ 7 & 2 \end{bmatrix}_{3 \times 2}$$

- Rows become columns, columns become rows and the element in the i th row and j th column of \mathbf{A} becomes the element in the j th row and i th column of the transposed matrix, i.e.,

$$[a_{ij}]'_{r \times c} = [a_{ji}]_{c \times r},$$

however you wish to look at it.

- The transpose operation is often denoted with “prime,” \mathbf{A}' , or by \mathbf{A}^t or \mathbf{A}^T (the latter notation used in [Far14]).
- Often used result: $(\mathbf{AB})' = \mathbf{B}'\mathbf{A}'$, “**the transpose of a product is the product of the transposes in reverse order**”.

Example B.11 (SLR Continued: Simple Matrix Operations). *Strictly speaking, we now have enough matrix know-how to write down a linear model (e.g., [Far14, §2.5 item 2.]). We continue to use the matrices of the SLR Example B.4. We can write our response vector, \mathbf{Y} , as a matrix sum of a mean vector—which itself is a matrix product—and error vector (again, we learn regression and linear model details later):*

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \quad \text{where, again,}$$

$$\mathbf{Y}_{n \times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \quad \mathbf{X}_{n \times 2} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \quad \boldsymbol{\beta}_{2 \times 1} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} \quad \boldsymbol{\epsilon}_{n \times 1} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

Example B.12 (SLR Continued: Using Transpose and Product). *The matrices $\mathbf{X}'\mathbf{X}$ and $\mathbf{X}'\mathbf{Y}$ are always behind the scenes in linear statistical models:*

$$\begin{aligned} \mathbf{X}'\mathbf{X} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \\ &= \begin{bmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} \mathbf{X}'\mathbf{Y} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \\ &= \begin{bmatrix} \sum_{i=1}^n Y_i \\ \sum_{i=1}^n x_i Y_i \end{bmatrix} \end{aligned}$$

- Corresponding functionality in R is illustrated in the following chunk.
- Note that computing $\mathbf{X}'\mathbf{X}$ and $\mathbf{X}'\mathbf{Y}$ using the transpose and matrix multiplication function, `%*%`, can be inefficient.
- There are even more efficient ways to compute for linear models; see the QR decomposition in [Far14, §2.7].

```
> ### A matrix and its transpose
> (A<- matrix(c(1,6,4,3,7,2), nrow=2, ncol=3))

      [,1] [,2] [,3]
[1,]     1     4     7
[2,]     6     3     2

> t(A)

      [,1] [,2]
[1,]     1     6
[2,]     4     3
[3,]     7     2

> ##  $\mathbf{A}'\mathbf{A}$  is always a symmetric result.
> t(A)%*%A

      [,1] [,2] [,3]
[1,]    37    22   19
[2,]    22    25   34
[3,]    19    34   53

> ## More efficiently.
> crossprod(A)

      [,1] [,2] [,3]
[1,]    37    22   19
[2,]    22    25   34
[3,]    19    34   53
```

```

> ## So is AA', which is generally different!
> A%*%t(A)

      [,1] [,2]
[1,]    66   32
[2,]    32   49

> ## More efficiently
> tcrossprod(A)

      [,1] [,2]
[1,]    66   32
[2,]    32   49

> ## A'B
> (B<- matrix(c(3,9),nrow=2, ncol=1))

      [,1]
[1,]    3
[2,]    9

> t(A)%*%B

      [,1]
[1,]    57
[2,]    39
[3,]    39

> ## More efficiently
> crossprod(A,B)

      [,1]
[1,]    57
[2,]    39
[3,]    39

```

B.2.5 Special Matrices

square: $r = c$, e.g., $B_{2 \times 2} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$

symmetric: $[a_{ij}] = [a_{ji}]$ (necessarily square, right?!)

e.g., $\mathbf{B}_{2 \times 2} = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}$

e.g., (SLR cont'd)

$$\mathbf{X}'\mathbf{X} = \begin{bmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix}$$

\mathbf{XX}' is symmetric, too, but we do not use it in this class. In words, **a matrix is symmetric if and only if it is equal to its own transpose** (in which case we may drop transpose symbols if symmetry is understood).

identity: All ones on diagonal, e.g.,

$$\mathbf{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Used often. Note $\mathbf{IA} = \mathbf{AI} = \mathbf{A}$ for any conformable \mathbf{A} and \mathbf{I} . The matrix version of number 1. Square, symmetric and diagonal.

diagonal: All zeros off main (upper left to lower right) diagonal

e.g.,

$$\mathbf{A}_{rr} = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ 0 & a_{22} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & a_{rr} \end{bmatrix}_{r \times r}$$

e.g.,

$$\begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}_{n \times n}$$

e.g., often in regression/ANOVA

$$\sigma^2 \mathbf{I}.$$

Square and symmetric.

unity: vector:

$$\mathbf{1}_{n \times 1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}_{n \times 1}$$

matrix:

$$\mathbf{J}_{n \times n} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{n \times n}$$

Note $\mathbf{J} = \mathbf{1}\mathbf{1}'$ and $n = \mathbf{1}'_{n \times 1} \mathbf{1}_{n \times 1}$

zero: vector:

$$\mathbf{0}_{n \times 1} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n \times 1}$$

matrix:

$$\mathbf{0}_{n \times n} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}_{n \times n}$$

Note $\mathbf{0} = \mathbf{0A}$

unit vector in i th coordinate direction or standard unit vector: One in i th element zeros elsewhere,

$$\mathbf{e}_i = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n \times 1} .$$

The i th column (or transposed row) of $\mathbf{I}_{n \times n}$.

Corresponding R functionality.

```
> ## Quick way to create identity matrices
> (I2<- diag(2))

[,1] [,2]
[1,]    1    0
[2,]    0    1

> (I3<- diag(3))

[,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1

> ## Identity matrices in action (or lack thereof)
> I2%*%A ## A from previous chunk

[,1] [,2] [,3]
[1,]    1    4    7
[2,]    6    3    2

> A%*%I3

[,1] [,2] [,3]
[1,]    1    4    7
[2,]    6    3    2

> ## Create a diagonal matrix
> diag(c(1,2,3,4))

[,1] [,2] [,3] [,4]
[1,]    1    0    0    0
[2,]    0    2    0    0
[3,]    0    0    3    0
[4,]    0    0    0    4

> ## Extract diagonal from square matrix
> diag(diag(c(1,2,3,4)))

[1] 1 2 3 4
```

```
> ## Check symmetry
> isSymmetric(crossprod(A))

[1] TRUE

> isSymmetric(tcrossprod(A))

[1] TRUE

> ## Matrix 1
> one<- matrix(1,nrow=3)
> crossprod(one)

[,1]
[1,]    3

> ## Matrix J
> matrix(1, nrow=3, ncol=3)

 [,1] [,2] [,3]
[1,]    1    1    1
[2,]    1    1    1
[3,]    1    1    1

> tcrossprod(one)

 [,1] [,2] [,3]
[1,]    1    1    1
[2,]    1    1    1
[3,]    1    1    1

> ## Unit vector
> e2<- rep(0,3)
> e2[2]<-1
> e2

[1] 0 1 0

> as.matrix(e2)

 [,1]
[1,]    0
[2,]    1
[3,]    0
```

```
> diag(3)[,2]
[1] 0 1 0

> diag(3)[3,]
[1] 0 0 1

> diag(3)[,2,drop=FALSE]
[,1]
[1,] 0
[2,] 1
[3,] 0

> diag(3)[2,,drop=FALSE]
[,1] [,2] [,3]
[1,] 0 1 0
```

B.2.6 Linear Dependence & Rank

Definition B.5 (Linear Combination (of vectors):). Let \mathbf{a}_i , $i = 1, \dots, n$, be column matrices (i.e., vectors) of the same size (same number of rows). Let c_i , $i = 1, \dots, n$ be some scalars (numbers not matrices). Then

$$\sum_{i=1}^n c_i \mathbf{a}_i = c_1 \mathbf{a}_2 + \cdots + c_n \mathbf{a}_n$$

is a **linear combination** of the \mathbf{a}_i vectors with **coefficients** c_i .

Example B.13 (Linear Combination).

$$\mathbf{a}_1 = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad \mathbf{a}_2 = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \quad \mathbf{a}_3 = \begin{bmatrix} 8 \\ 5 \end{bmatrix},$$

$c_1 = c_2 = c_3 = 1$. Then,

$$\begin{aligned}\sum_{i=1}^n c_i \mathbf{a}_i &= (1) \begin{bmatrix} 2 \\ 3 \end{bmatrix} + (1) \begin{bmatrix} 3 \\ 1 \end{bmatrix} + (1) \begin{bmatrix} 8 \\ 5 \end{bmatrix} \\ &= \begin{bmatrix} 13 \\ 9 \end{bmatrix}\end{aligned}$$

Definition B.6 (Linear Dependence (of vectors)). Vectors \mathbf{a}_i , $i = 1, \dots, n$, are said to be **linearly dependent** if there exists numbers c_i , $i = 1, \dots, n$, not all zero, such that

$$c_1 \mathbf{a}_1 + \cdots + c_n \mathbf{a}_n = \mathbf{0}.$$

Else, the \mathbf{a}_i are said to be **linearly independent** (different than independence of rvs). In this case, we can express at least one \mathbf{a}_i as a linear combination of the remaining \mathbf{a}_i .

Example B.14 (Linear Dependence). Continuing the above Example B.13, we see $\mathbf{a}_3 = \mathbf{a}_1 + 2\mathbf{a}_2$, i.e., $\mathbf{a}_1 + 2\mathbf{a}_2 + (-1)\mathbf{a}_3 = \mathbf{0}$, i.e.,

$$(1) \begin{bmatrix} 2 \\ 3 \end{bmatrix} + 2 \begin{bmatrix} 3 \\ 1 \end{bmatrix} + (-1) \begin{bmatrix} 8 \\ 5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Thus, the \mathbf{a}_i are said to be linearly dependent. Given a subset of (two in this case) the vectors, we can reproduce the remaining (one in this case) vectors.

Example B.15 (Multiple Linear Regression (MLR):). Using the set-up in the SLR examples above (B.4, B.11, B.12), suppose your colleague wishes to

include a second “ x ” variable in the following manner:

$$x_{i2} = 2x_{i1},$$

where we now include extra subscripts to distinguish our two covariates. (Say the x_{i1} are tree diameters, previously denoted just as x_i , so that, here, the x_{i2} are just twice the diameter values.) In this case, the \mathbf{X} matrix for (multiple) linear regression would be

$$\begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} \end{bmatrix}$$

But, because of the way your colleague created the x_{i2} values, we know

$$(0) \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + (-2) \begin{bmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{n1} \end{bmatrix} + (1) \begin{bmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{n2} \end{bmatrix} = \mathbf{0}.$$

This sort of dependence will lead to problems in regression. (Admittedly, it seems a bit contrived, but you'd be surprised what people do to their data!)

Definition B.7 (Rank). The (maximum) number of linearly independent columns (or rows) of matrix. (Thus, $\text{rank}(\mathbf{A}_{r \times c}) \leq \min(r, c)$.) In other words, rank is the maximum number of columns (rows) that can be selected before one of the selected vectors can be reproduced by a linear combination of the other selected vectors.

Example B.16 (Rank). In Example B.13 we have

$$\begin{aligned}\mathbf{A} &= \begin{bmatrix} | & | & | \\ \mathbf{a}_1 & \mathbf{a}_2 & \mathbf{a}_3 \\ | & | & | \end{bmatrix} \\ &= \begin{bmatrix} 2 & 3 & 8 \\ 3 & 1 & 5 \end{bmatrix}\end{aligned}$$

$\text{rank}(\mathbf{A}) = 2$ Why?

For the strange MLR Example B.15 we have $\text{rank}(\mathbf{X}) = 2$, assuming $n \geq 2$ distinct diameters (usually $n \gg 2$).

Definition B.8 (Inverse (of square matrix)). Let \mathbf{A} be square. The inverse of \mathbf{A} , if it exists, is denoted \mathbf{A}^{-1} , and is such that

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}.$$

e.g., Let $\mathbf{A} = [6]$. Then $\mathbf{A}^{-1} = [1/6]$ and $[1/6][6] = [1] = [6][1/6]$.

e.g.,

$$\begin{bmatrix} 2 & 1 \\ 3 & 2 \end{bmatrix}^{-1} = \begin{bmatrix} 2 & -1 \\ -3 & 2 \end{bmatrix}$$

because

$$\begin{bmatrix} 2 & -1 \\ -3 & 2 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -3 & 2 \end{bmatrix}.$$

NOTE: Perhaps obviously, $(\mathbf{A}^{-1})^{-1} = \mathbf{A}$ (inverse of inverse is the original matrix).

NOTE: $(\mathbf{A}^{-1})' = (\mathbf{A}')^{-1}$ and is often denoted \mathbf{A}^{-T} while using $'$ doesn't seem fashionable.

Often used result: $(\mathbf{AB})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$, the inverse of a product is the product of the inverses in reverse order (assuming the matrices are invertible).

The next chunk illustrates the computation of a matrix inverse in R.

```
> ### A matrix and its inverse.
> set.seed(8675309)
> (B<- matrix(round(10*rnorm(16)), nrow=4, ncol=4))

      [,1]  [,2]  [,3]  [,4]
[1,]   -10    11     6     2
[2,]     7    10     9    -7
[3,]    -6     0   -15   -10
[4,]    20     7    10    20

> (Binv<- solve(B))

      [,1]      [,2]      [,3]      [,4]
[1,] -0.04235901  0.030337  0.01674220  0.023225
[2,]  0.04783484  0.025176  0.05538771  0.031722
[3,] -0.00020141  0.020947 -0.07391742 -0.029607
[4,]  0.02571752 -0.049622  0.00083082  0.030476

> round(Binv%*%B,4)

      [,1]  [,2]  [,3]  [,4]
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
[4,]    0    0    0    1

> round(B%*%Binv,4)

      [,1]  [,2]  [,3]  [,4]
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
[4,]    0    0    0    1
```

A Few More Things

- If \mathbf{A}^{-1} exists, then \mathbf{A} is said to be **non-singular**, else it's **singular**.
(Square matrix implied here.)

- If $\text{rank}(\mathbf{A}_{r \times c}) = \min(r, c)$, then \mathbf{A} is said to be **full rank**, else, if $\text{rank}(\mathbf{A}_{r \times c}) < \min(r, c)$, then it is not full rank, and we say \mathbf{A} is **rank deficient**.
- If \mathbf{A} is square, then \mathbf{A} being full rank is equivalent to existence of \mathbf{A}^{-1} .
- If \mathbf{A} is square, then \mathbf{A} being full rank is equivalent non-zero determinant, $|A|$. (Non-full rank equivalent to zero determinant). Formal definition of determinant?
- $\text{rank}(AB) = \min(\text{rank}(A), \text{rank}(B))$
- Relate to eigen decomposition and non-zero eigenvalues?

```

> ## We saw (square) matrix B, above, is full rank:
> det(B)
[1] 79440

> ## Matrix A (rank 2) in a previous e.g. of linear dependence:
> A<- matrix(c(2,3,8,
+               3,1,5), nrow=2, byrow=TRUE)
> ## rank(A'A) = min(rank(A'),rank(A))= rank(A) = 2 = rank(AA')
> det(crossprod(A)) ## 3x3 not full rank
[1] 0

> solve(crossprod(A)) ## does not exist
Error in solve.default(crossprod(A)): Lapack routine dgesv: system is exactly
singular: U[3,3] = 0

> det(tcrossprod(A)) ## 2x2 full rank
[1] 294

> solve(tcrossprod(A)) ## exists
[,1]      [,2]
[1,]  0.11905 -0.16667
[2,] -0.16667  0.26190

```

Whoops, I forgot to mention the trace of a matrix.

```
> ## Trace of (square) matrix
> sum(diag(B))
```

```
[1] 5
```

```
> ## Eigen value / vector decomposition (sans further detail!)
> eigen(B)

eigen() decomposition
$values
[1] 16.567+11.346i 16.567-11.346i -15.000+ 0.000i -13.135+ 0.000i
```

```
$vectors
[,1] [,2] [,3] [,4]
[1,] 0.03737+0.21907i 0.03737-0.21907i -0.69264+0i 0.17818+0i
[2,] -0.14690+0.54477i -0.14690-0.54477i 0.44818+0i -0.45660+0i
[3,] -0.23254+0.04194i -0.23254-0.04194i -0.38299+0i 0.83134+0i
[4,] 0.75922+0.00000i 0.75922+0.00000i 0.41558+0i -0.26199+0i
```

```
> eigen(A)
```

```
Error in eigen(A): non-square matrix in 'eigen'
```

Example B.17 (Regression with 2 or More x Variables (MLR)).

Modeling Assumptions:

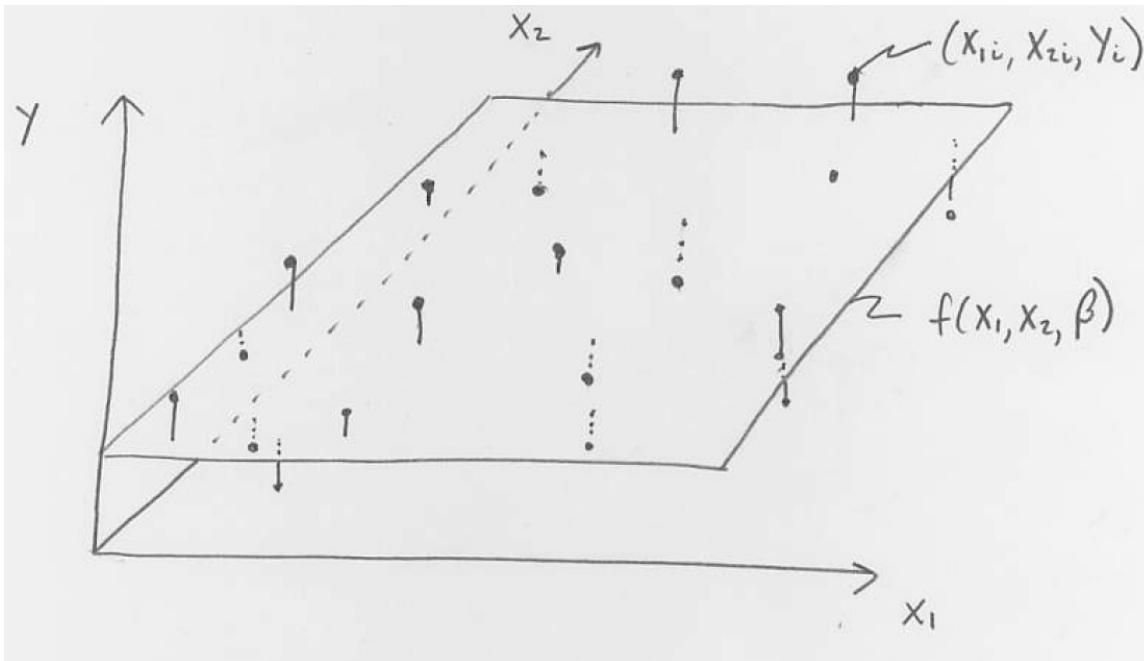
$$Y_i = \mu_i + \varepsilon_i, \quad \varepsilon_i \stackrel{ind}{\sim} N(0, \sigma^2) \quad i = 1, \dots, n$$

where

$$\mu_i = \overbrace{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_{p-1} x_{i(p-1)}}^{\mu(Y|x_{i1}, \dots, x_{i(p-1)})}$$

i.e.,

$$Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_{p-1} x_{i(p-1)} + \varepsilon_i.$$



(Can you find **notational inconsistency** in above figure?)

- **Major Goal:** not surprisingly, estimate the mean(s), μ_i , which, in this case, means estimate

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{p-1} \end{bmatrix}.$$

- We will also estimate σ^2 .
- On the way to estimating β , we will come to the **normal equations** (unrelated to the normal distribution) (see [Far14, Chap. 2]):

$$\mathbf{X}'\mathbf{X}\beta = \mathbf{X}'\mathbf{Y} \quad \text{details omitted.}$$

- If $(\mathbf{X}'\mathbf{X})^{-1}$ exists, we left-multiply both sides of the above to get

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}.$$

$\hat{\beta}$ is the **estimator** or **estimate** of the unknown parameter β .

- See [Far14, §2.4] for normal equations, etc.
- Note that we used p to denote the number of parameters in the linear regression model whereas your text's author uses p to denote the number of predictors; the same thing if no intercept ($\beta_0 = 0$) of course ([Far14, §1.3]).
- We wait to estimate σ^2 . (Remember, strictly speaking, we're still just doing matrix manipulations at this point, and we don't want to get too far ahead of ourselves, though some of you might be "getting" the regression stuff a bit.)
- Again, there are more efficient ways to do computations for linear models than suggested here. See the QR decomposition in [Far14, §2.7].

Example B.18 (Rank Deficiency and Redundancy in Regression).

- In regression/ANOVA, the number of rows, n , of \mathbf{X} (n is number of observations) is typically greater than the number of columns/mean parameters ($n > p$).
- Thus, if \mathbf{X} is full rank, $\text{rank}(\mathbf{X}) = \min(n, p) = p$ and, by a previously stated result, $\text{rank}(\mathbf{X}'\mathbf{X}) = \min(\text{rank}(\mathbf{X}'), \text{rank}(\mathbf{X})) = \min(\min(n, p), \min(n, p)) = \min(n, p) = p$.
- That is $\mathbf{X}'\mathbf{X}$ is full-rank ($= p$), and is non-singular, and $(\mathbf{X}'\mathbf{X})^{-1}$ exists, and we may solve for β .
- (**NOTE:** Again, your author uses p to denote the number of predictors in a linear model. With his convention, then, if we do not include an intercept in our model, then full-rank = p . However, if we include an intercept, then full-rank = $p + 1$.)

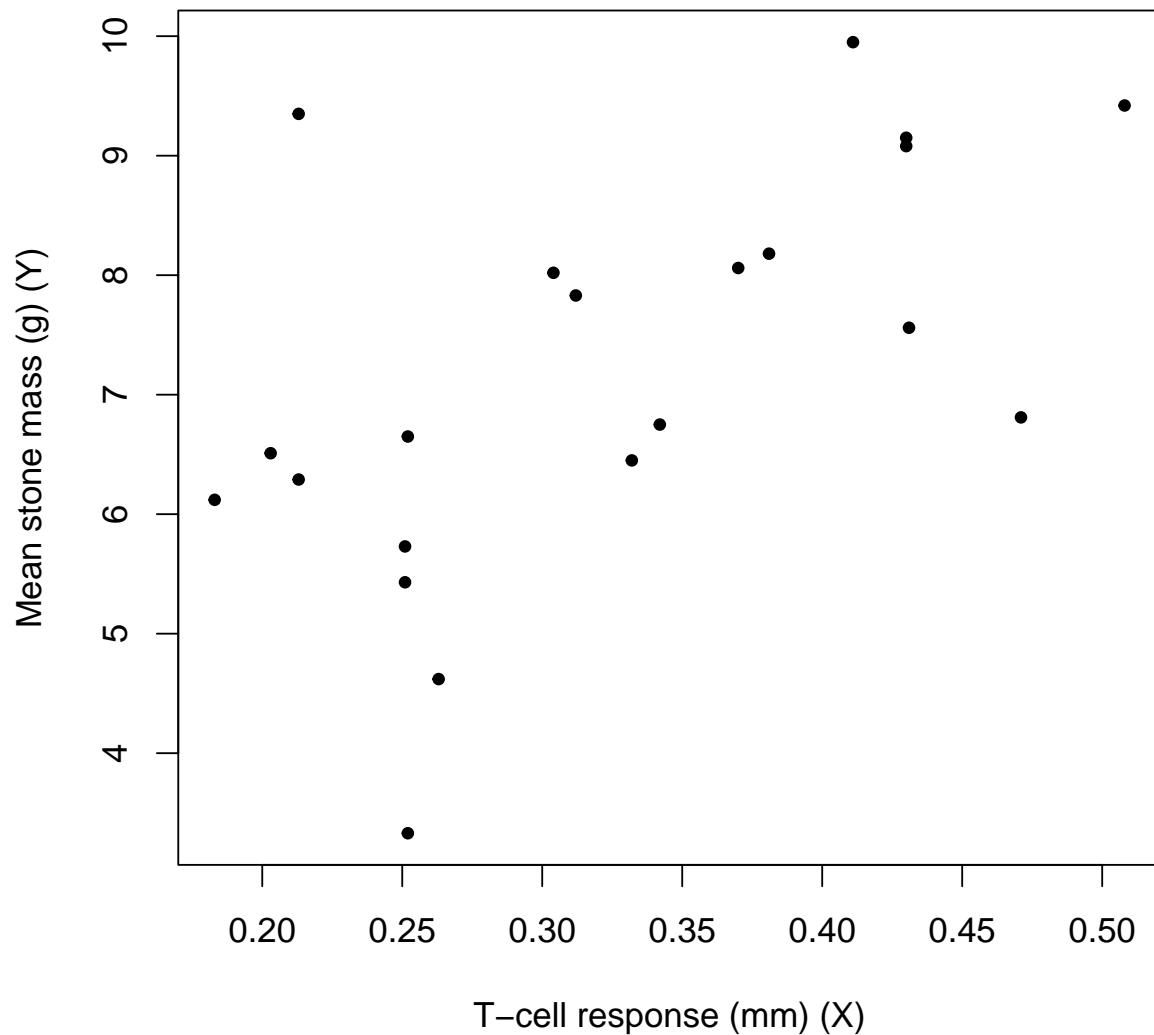
- Else, if $\text{rank}(X) < p$, then $X'X$ is singular and the inverse does not exist. In this case, we must resort to “rank deficient” methods, which we do not cover.
- We will encounter briefly rank deficient X when discussing ANOVA, but will introduce fairly straightforward ways to get to an equivalent (in some sense) full rank X so that $X'X$ is invertible to get solutions to our linear models.
- If X is not full-rank, that means we have linear dependence among the columns of X ; we can reproduce at least one column from a linear combination of the others.
- In this sense, we have redundant information amongst our X variables and may be able to omit one or more. See Example B.15.
- Related to multicollinearity and non-identifiability / non-estimability of mean parameters and variable selection and shrinkage. See [Far14, §2.10, 2.11, 7.3 & Chap. 10 & 11]. Perhaps more later.

- See [KNNL05, Sec. 5.7] for more basic matrix results.

- The following chunk illustrates even more matrix functionality in R in the context of regression.
- Again, we haven’t arrived at regression yet. We are simply illustrating some of the matrix functionality presented above.
- Of course, it’s meant to show also that matrices are useful for regression/ANOVA computations.

```
> ### Data
> ex0727.df<- Sleuth3::ex0727
>
> ### Plot:
> par(cex=1.2)
> plot(Mass ~ Tcell, data=ex0727.df,
+       pch=20,
+       main="Weight-lifting and Bird Health",
+       xlab="T-cell response (mm) (X)",
+       ylab="Mean stone mass (g) (Y)")
```

Weight-lifting and Bird Health



```
> X<- model.matrix(Mass ~ Tcell + I(Tcell^2), data= ex0727.df)
> head(X, n=3); tail(X,n=3)

(Intercept) Tcell I(Tcell^2)
1           1 0.252  0.063504
2           1 0.263  0.069169
3           1 0.251  0.063001
(Intercept) Tcell I(Tcell^2)
```

```

19      1 0.213  0.045369
20      1 0.508  0.258064
21      1 0.411  0.168921

> dim(X)
[1] 21  3

> n<- dim(X)[1]
> Y<- ex0727.df$Mass
>
> (betahat<- solve(t(X) %*% X) %*% t(X) %*% Y)

[,1]
(Intercept) 5.7669
Tcell       -1.8012
I(Tcell^2)  17.7497

> solve(t(X) %*% X, t(X) %*% Y)

[,1]
(Intercept) 5.7669
Tcell       -1.8012
I(Tcell^2)  17.7497

> coefficients(lm(Mass ~ Tcell + I(Tcell^2), data = ex0727.df))

(Intercept)      Tcell   I(Tcell^2)
5.7669        -1.8012    17.7497

```

B.3 Combining Things: Random Vectors and Matrices

- Not surprisingly, a random matrix is a matrix whose elements are random variables. We focus on vectors.
- Just like random (scalar) variables (Appendix A), random vectors are associated with probability models (pdf, pmf), but we will minimize dis-

cussion of these joint models, for the moment, and focus instead on easy-to-understand summarizing properties of these models.

- We will cover joint distributions as well as marginal distributions and conditional distributions more thoroughly, shortly.

B.3.1 Expectation of a Random Vector/Matrix

If

$$\mathbf{Y}_{n \times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}$$

is an rv, then

$$E(\mathbf{Y}) = \begin{bmatrix} E(Y_1) \\ E(Y_2) \\ \vdots \\ E(Y_n) \end{bmatrix}$$

In words: **expected value of a random vector is the vector of expected values.**

Example B.19 (Expected Value of Normal RV). *If we assume $Y_i \stackrel{\text{ind}}{\sim} N(\mu, \sigma^2)$, then*

$$E(\mathbf{Y}) = \begin{bmatrix} E(Y_1) \\ E(Y_2) \\ \vdots \\ E(Y_n) \end{bmatrix} = \begin{bmatrix} \mu \\ \mu \\ \vdots \\ \mu \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [\mu]$$

This result has nothing to do with independence. See, incidentally, [Far14, §2.2 p. 15], for this “null model.”

B.3.2 Variance(-Covariance) Matrix

$$\text{Var}(\mathbf{Y}) = \begin{bmatrix} \text{Var}(Y_1) & \text{Cov}(Y_1, Y_2) & \cdots & \text{Cov}(Y_1, Y_n) \\ \text{Cov}(Y_2, Y_1) & \text{Var}(Y_2) & \cdots & \text{Cov}(Y_2, Y_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{Cov}(Y_n, Y_1) & \text{Cov}(Y_n, Y_2) & \cdots & \text{Var}(Y_n) \end{bmatrix}$$

which can be written as

$$\begin{aligned} [\text{Cov}(Y_i, Y_j)] &= [\mathbb{E}((Y_i - \mathbb{E}(Y_i))(Y_j - \mathbb{E}(Y_j)))] \\ &= \mathbb{E} \left(\begin{bmatrix} Y_1 - \mathbb{E}(Y_1) \\ Y_2 - \mathbb{E}(Y_2) \\ \vdots \\ Y_n - \mathbb{E}(Y_n) \end{bmatrix} \begin{bmatrix} Y_1 - \mathbb{E}(Y_1) & Y_2 - \mathbb{E}(Y_2) & \cdots & Y_n - \mathbb{E}(Y_n) \end{bmatrix}' \right) \\ &= \mathbb{E}((\mathbf{Y} - \mathbb{E}(\mathbf{Y}))(\mathbf{Y} - \mathbb{E}(\mathbf{Y}))') \end{aligned}$$

In words, **the variance of a random vector is the matrix of variances and covariances computed from the elements of the random vector.**

B.3.3 Linearity of Expectation Operator (just as in scalar case)

$$\begin{aligned} \mathbb{E}(\mathbf{a} + \mathbf{B}\mathbf{Y}) &= \mathbb{E}(\mathbf{a}) + \mathbb{E}(\mathbf{B}\mathbf{Y}) \\ &= \mathbf{a} + \mathbf{B}\mathbb{E}(\mathbf{Y}) \end{aligned}$$

where \mathbf{a} is a vector of constants, \mathbf{B} is a matrix of constants, and \mathbf{Y} is an rv. We gave an (almost) analogous result in terms of scalars in a previous remark (A.2).

Example B.20 (Additive Error Model (again)).

- Let $\epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$ and $\mathbf{Y} = \mathbf{X}\beta + \epsilon$ where \mathbf{X} is a constant matrix, β a constant vector and σ^2 are some constant scalar.

- Then,

$$E(\mathbf{Y}) = \mathbf{X}\boldsymbol{\beta}.$$

- See Example B.22, below.

B.3.4 Variance(–Covariance) of $\mathbf{a} + \mathbf{B}\mathbf{Y}$

$$\begin{aligned}\text{Var}(\mathbf{a} + \mathbf{B}\mathbf{Y}) &= \text{Var}(\mathbf{a}) + \text{Var}(\mathbf{B}\mathbf{Y}) \\ &= 0 + \mathbf{B}\text{Var}(\mathbf{Y})\mathbf{B}' \\ &= \mathbf{B}\text{Var}(\mathbf{Y})\mathbf{B}',\end{aligned}$$

which reduces to result given back in §A.8.2 when \mathbf{a} , \mathbf{Y} and \mathbf{B} are scalars.

Example B.21 (Additive Error Model (again)).

- Let $\epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I})$ and $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \epsilon$ where \mathbf{X} is a constant matrix, $\boldsymbol{\beta}$ a constant vector and σ^2 are some constant scalar.
- Then,

$$\text{Var}(\mathbf{Y}) = \mathbf{I}\text{Var}(\epsilon)\mathbf{I}^t = \sigma^2 \mathbf{I}$$

- See Example B.22, below.

B.3.5 Distribution of Linear Function of Normal RV

This next result, along with the results in the previous subsections, B.3.3 and B.3.4, will be used repeatedly throughout the course, though the details will often be behind the scenes, and the notation may change with context. The scalar analog was given in §A.10.

If $\mathbf{Y} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ (see [KNNL05, p.197] for pdf of such a vector), then
 $(\mathbf{a} + \mathbf{B}\mathbf{Y}) \sim N(\mathbf{a} + \mathbf{B}\boldsymbol{\mu}, \mathbf{B}\boldsymbol{\Sigma}\mathbf{B}')$

Example B.22 (Introductory Statistics Model in Matrix Form).

Assume

$$Y_i \stackrel{ind}{\sim} N(\mu, \sigma^2) \quad i = 1, \dots, n$$

Equivalently,

$$Y_i = \mu + \varepsilon_i \quad \varepsilon_i \stackrel{ind}{\sim} N(0, \sigma^2) \quad i = 1, \dots, n.$$

(We have a result for this equivalence, right?! Note, by the way, $\varepsilon_i = Y_i - \mu = Y_i - E(Y_i)$.)

"Stacking things" and using matrix/vector multiplication/addition, we can write

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

where

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \quad \boldsymbol{\beta} = [\mu], \quad \boldsymbol{\epsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}.$$

That is,

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [\mu] + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}.$$

Using our previously presented results, we know

-

$$\begin{aligned} E(\mathbf{Y}) &= E(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}) = E(\mathbf{X}\boldsymbol{\beta}) + E(\boldsymbol{\epsilon}) \\ &= \mathbf{X}\boldsymbol{\beta} + \mathbf{0} = \mathbf{X}\boldsymbol{\beta} = \begin{bmatrix} \mu \\ \mu \\ \vdots \\ \mu \end{bmatrix}, \end{aligned}$$

•

$$\begin{aligned}
 \text{Var}(\epsilon) &= [\text{Cov}(\varepsilon_i, \varepsilon_j)] \\
 &= \begin{bmatrix} \text{Var}(\varepsilon_1) & \text{Cov}(\varepsilon_1, \varepsilon_2) & \cdots & \text{Cov}(\varepsilon_1, \varepsilon_n) \\ \text{Cov}(\varepsilon_2, \varepsilon_1) & \text{Var}(\varepsilon_2) & \cdots & \text{Cov}(\varepsilon_2, \varepsilon_n) \\ \vdots & & \ddots & \vdots \\ \text{Cov}(\varepsilon_n, \varepsilon_1) & \text{Cov}(\varepsilon_n, \varepsilon_2) & \cdots & \text{Var}(\varepsilon_n) \end{bmatrix} \\
 &= \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{I}_n
 \end{aligned}$$

•

$$\begin{aligned}
 \text{Var}(\mathbf{Y}) &= \text{Var}(\mathbf{X}\boldsymbol{\beta} + \epsilon) \\
 &= \text{Var}(\mathbf{X}\boldsymbol{\beta}) + \text{Var}(\epsilon) \\
 &= 0 + \text{Var}(\epsilon) = \sigma^2 \mathbf{I}_n
 \end{aligned}$$

- So, we could denote our model as

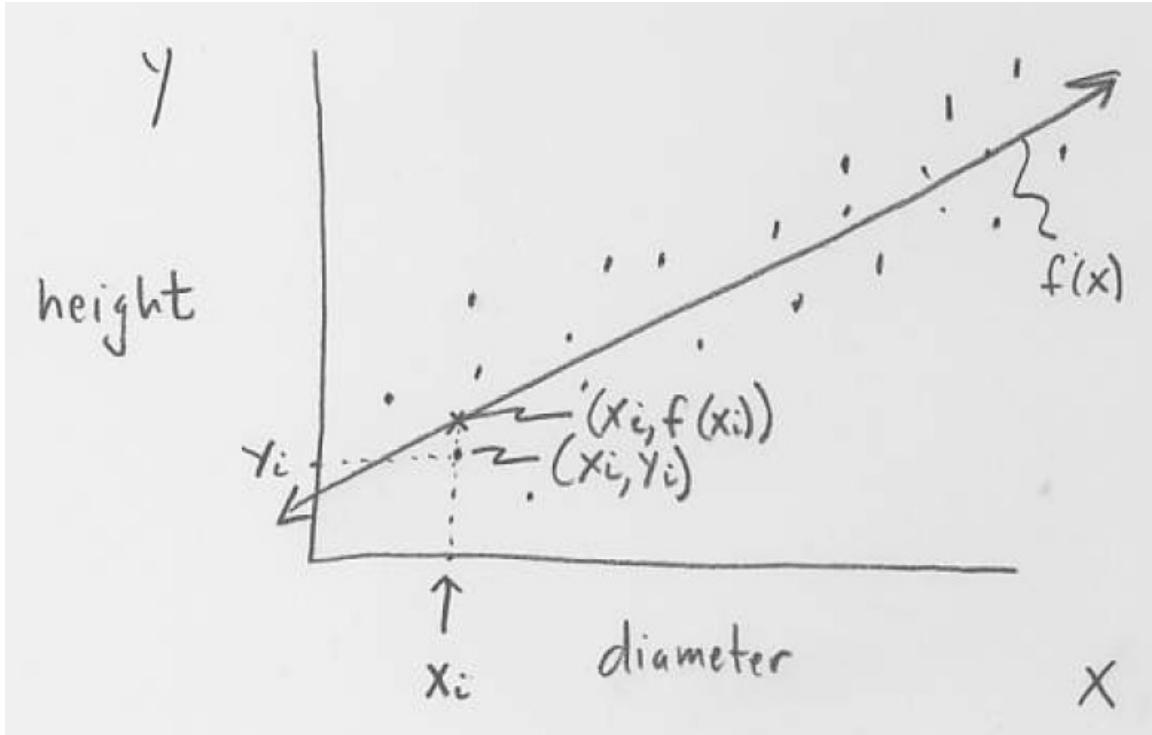
$$\mathbf{Y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n),$$

or as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \epsilon \quad \epsilon \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$$

- See [[Far14](#), Chap. 3, top half of p. 33].

Notice how the matrix notation encapsulates the previous simple model (for which our linear model machinery is a bit overkill) suffices essentially unchanged for a simple linear regression model, below (and for any other linear regression model or ANOVA model).

Example B.23 (SLR Model in Matrix Form).(tree diameter, tree height): $(X_i, Y_i), i = 1, \dots, n$ *Model Assumptions (just an example):*

$$Y_i = \mu_i + \varepsilon_i \quad \text{where } \mu_i = \mu(Y | x_i) = \beta_0 + \beta_1 x_i \\ \varepsilon_i \stackrel{\text{ind}}{\sim} N(0, \sigma^2)$$

or, equivalently,

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad \varepsilon_i \stackrel{\text{ind}}{\sim} N(0, \sigma^2),$$

or

$$Y_i \stackrel{\text{ind}}{\sim} N(\beta_0 + \beta_1 X_i, \sigma^2).$$

As in the previous example, we can write our model as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_n)$$

where (now a bit differently),

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

That is,

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}.$$

And,

$$E(\mathbf{Y}) = \mathbf{X}\boldsymbol{\beta} = \begin{bmatrix} \beta_0 + \beta_1 x_1 \\ \beta_0 + \beta_1 x_2 \\ \vdots \\ \beta_0 + \beta_1 x_n \end{bmatrix},$$

$$Var(\boldsymbol{\epsilon}) = \sigma^2 \mathbf{I},$$

$$Var(\mathbf{Y}) = \sigma^2 \mathbf{I}.$$

B.4 Normal and χ^2 Results

- **σ^2 Known.** In this section, we quickly present additional standard results that follow from assuming our model errors are normally distributed. Some of these results rely on knowing the error variance, σ^2 , which is not practical.
- **σ^2 Unknown.** Estimating σ^2 with MSE (§2.4), we then give analogous t and F distribution results (in a subsequent section).

- **Primary Probability Distributions.** These latter distributions will serve as our primary reference distributions for testing (error probabilities, p-values) and confidence (“compatibility”) interval/region levels under the assumption of normal errors.
- β . These results rely on knowing β , too, but our methods get around this.

Recall our model, now assuming normal errors.

$$\begin{aligned} Y_i &= \mu_i + \epsilon_i \quad \text{where } \mu_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} \\ \epsilon_i &\stackrel{\text{iid}}{\sim} N(0, \sigma^2) \end{aligned}$$

(iid = independent and identically distributed) or, equivalently,

$$Y_i \stackrel{\text{ind}}{\sim} N(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik}, \sigma^2),$$

(ind = independent) which we can write in matrix notation

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}).$$

Result B.1 (Normal Distribution for $\hat{\boldsymbol{\beta}}$).

With the estimator's mean and variance, discussed in §2.4, the red result in §B.3.5 tells us that, if our errors are normal, then our LS estimator is **normal**,

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1}).$$

Result B.2 (Normal Distribution for Linear Combination $\mathbf{C}\hat{\boldsymbol{\beta}}$).

By the *red result* in §B.3.5 applied to the above result B.1, we have the more general result

$$\mathbf{C}\hat{\boldsymbol{\beta}} \sim N(\mathbf{C}\boldsymbol{\beta}, \sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T),$$

for \mathbf{C} a matrix of one or more known row vectors (linear combination coefficients).

Result B.3 (Standard Normal Distribution for Standardized $\mathbf{C}\hat{\boldsymbol{\beta}}$).

Again, by the same *red result* in §B.3.5, we have

$$(\sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1/2} (\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta}) \sim N(\mathbf{0}, \mathbf{I}).$$

- **Z-Score.** Notice, of course, the analogy to the standardized ‘z-score’ of previous courses: $Z = (Y - \mu)/\sigma \sim N(0, 1)$.
- We use the notation $(\sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1/2}$ to denote the inverse of a “matrix square-root”: For any symmetric, non-negative definite matrix \mathbf{A} , there is a (square, symmetric, nnd) matrix \mathbf{R} such that $\mathbf{A} = \mathbf{R}^2$ [Har97, Theorem 21.9.1].
- Other types of matrix square roots (more or less) may be more popular [Har97, §14.5].
- Incidentally, in practice, most variance matrices have such factorizations into the product of “square-root” matrices.

Result B.4 ($\chi^2(df=\text{rank}(\mathbf{C}))$ for Quadratic Form of $\mathbf{C}\hat{\boldsymbol{\beta}}$).

- If we take the standard normal vector of Result B.3, and we take the inner product of that standard normal vector with itself, so that we

get a sum-of-squared independent standard normals, then we get a $\chi^2(df=rank(\mathbf{C}))$.

- That is,

$$(\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta})^T (\text{Var}(\mathbf{C}\hat{\boldsymbol{\beta}}))^{-1} (\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta}) \sim \chi^2(\text{rank}(\mathbf{C})),$$

or, more precisely,

$$(\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta})^T (\sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1} (\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta}) \sim \chi^2(\text{rank}(\mathbf{C})).$$

- See discussion of the **rank** of a matrix in Definition B.7 and §B.2.6.
- Typically, in the context of linear model practice, it refers to the number of (linearly independent) rows of \mathbf{C} , i.e., the number of linear combinations of $\boldsymbol{\beta}$ about which we wish to (simultaneously) infer. See §B.2.6 for linear (in)dependence of vectors.
- $Z^2 = \chi^2$. Notice, of course, the analogy to squaring a standard normal to get a χ^2 (with 1 df), which you may have heard of before.

Result B.5 ($N(0,1)$ for Standardized Scalar $\mathbf{C}\hat{\boldsymbol{\beta}}$).

- If $\mathbf{C}\boldsymbol{\beta}$ is a scalar (i.e., if \mathbf{C} is a single row), then Result B.3 gives a single $N(0, 1)$ rv.
- That is,

$$\frac{\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta}}{\sqrt{\text{Var}(\mathbf{C}\hat{\boldsymbol{\beta}})}} \sim N(0, 1),$$

or, more precisely,

$$\frac{\mathbf{C}\hat{\boldsymbol{\beta}} - \mathbf{C}\boldsymbol{\beta}}{\sqrt{\sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T}} \sim N(0, 1).$$

- We may perform **hypothesis tests** and construct **confidence intervals** using this reference distribution (if we know σ^2).
- $Z^2 = \chi^2$ Again. See Results B.7 and B.8, below.

- For example, we may want to infer about the linear regression function at a particular covariate value, i.e., about $\mathbf{x}^T\boldsymbol{\beta}$ (in this case \mathbf{C} is just the row vector \mathbf{x}^T).

Result B.6 (Standard Error of an Estimator).

- If $\hat{\theta}$ is a scalar estimator, then the standard error is

$$se(\hat{\theta}) \equiv \sqrt{Var(\hat{\theta})}.$$

- In other words, the **standard error** of an estimator is just another name for the **standard deviation** of an estimator, which we mentioned before.

- Typically, the standard error (standard deviation) is discussed only for scalars, as the square root of their variance, but you could define a standard error (standard deviation) square root matrix, which we just mentioned in passing, above.
- For us, in the scalar case of $\mathbf{C}\hat{\beta}$ (Result B.5),

$$se(\mathbf{C}\hat{\beta}) = \sqrt{\sigma^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T}.$$

- And, more particularly, in the case of $\hat{\beta}_j$ (Result B.7),

$$\text{se}(\hat{\beta}_j) = \sqrt{\sigma^2(\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}}.$$

Result B.7 ($N(0,1)$ for Standardized $\hat{\beta}_j$).

- An important special case of Result B.5 is when \mathbf{C} is a row vector consisting of a single element of 1 and remaining elements 0.
- That is,

$$\frac{\hat{\beta}_j - \beta_j}{\sqrt{\sigma^2(\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}}} \sim N(0, 1),$$

where $(\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}$ denotes the diagonal element of $(\mathbf{X}^T \mathbf{X})^{-1}$ corresponding to parameter β_j (perhaps with a fix-up for beginning at $j = 0$).

- $(\text{Var}(\hat{\beta})) = \sigma^2(\mathbf{X}^T \mathbf{X})^{-1}$ is just the variance matrix of the $\hat{\beta}$ vector, and we are merely picking off the correct variance element from its diagonal.)
- We may perform **hypothesis tests** and construct **confidence intervals** using this reference distribution (if we know σ^2).

Result B.8 ($\chi^2(df = 1)$ for Inferring $\hat{\beta}_j$).

- Squaring the above standard normal in Result B.7 we get a $\chi^2(df=1)$.
- That is,

$$\frac{(\hat{\beta}_j - \beta_j)^2}{\text{Var}(\hat{\beta}_j)} \sim \chi^2(df = 1),$$

or, more precisely,

$$\frac{(\hat{\beta}_j - \beta_j)^2}{\sigma^2(\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}} \sim \chi^2(df = 1).$$

B.5 t and F Distribution Results

- Upon replacing σ^2 with $\hat{\sigma}^2 = \text{MSE}$ in the above results on normal and χ^2 -square random vectors / variables, we get analogous results for t and F random variables, respectively, which do not depend on σ^2 , and which are more practically useful for inference. (F is not quite analogous to χ^2 ; we need to scale a bit.)
- Of course, we still don't know β , but we will get around that.

Result B.9 ($F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p)$ for Quadratic Form of $\mathbf{C}\hat{\beta}$).

- Replacing σ^2 (in $\text{Var}(\mathbf{C}\hat{\beta})$) in the χ^2 Result B.4 with our unbiased estimate $\hat{\sigma}^2$, we get a $F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p)$ random variable. (We need a bit of additional scaling to get from χ^2 to F ; F means are $df_2/(df_2 - 2)$ ($df_2 > 2$), while 'corresponding' χ^2 means are df_2 .)
- That is,

$$(\mathbf{C}\hat{\beta} - \mathbf{C}\beta)^T (\widehat{\text{Var}}(\mathbf{C}\hat{\beta}))^{-1} (\mathbf{C}\hat{\beta} - \mathbf{C}\beta) / \text{rank}(\mathbf{C}) \sim F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p),$$

or, more precisely,

$$(\mathbf{C}\hat{\beta} - \mathbf{C}\beta)^T (\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T)^{-1} (\mathbf{C}\hat{\beta} - \mathbf{C}\beta) / \text{rank}(\mathbf{C}) \sim F(df_1 = \text{rank}(\mathbf{C}), df_2 = n - p),$$

Result B.10 ($t(df = n - p)$ for Standardized Scalar $\mathbf{C}\hat{\beta}$).

- If $\mathbf{C}\beta$ is a scalar (i.e., if \mathbf{C} is a single row), then, upon replacing σ^2 (in $\text{Var}(\mathbf{C}\hat{\beta})$) in the normal Result B.5 with our unbiased estimate $\hat{\sigma}^2$, we get a t random variable.

- That is,

$$\frac{\mathbf{C}\hat{\beta} - \mathbf{C}\beta}{\sqrt{\text{Var}(\mathbf{C}\hat{\beta})}} \sim t(df = n - p),$$

or, more precisely,

$$\frac{\mathbf{C}\hat{\beta} - \mathbf{C}\beta}{\sqrt{\hat{\sigma}^2 \mathbf{C}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}^T}} \sim t(df = n - p).$$

Result B.11 ($t(df = n - p)$ for Standardized $\hat{\beta}_j$).

- Similarly, and more particularly, upon replacing σ^2 in the normal Result B.7 with our unbiased estimate $\hat{\sigma}^2$, we get a t random variable.

- That is,

$$\frac{\hat{\beta}_j - \beta_j}{\sqrt{\hat{\sigma}^2 (\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}}} \sim t(df = n - p),$$

where, again, $(\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}$ denotes the diagonal element of $(\mathbf{X}^T \mathbf{X})^{-1}$ corresponding to parameter β_j .

- $(\widehat{\text{Var}}(\hat{\beta})) = \hat{\sigma}^2 (\mathbf{X}^T \mathbf{X})^{-1}$ is just the estimated variance matrix of the $\hat{\beta}$ vector, and we are merely picking off the correct variance element from its diagonal.)

- Note that $t^2(n - p) = F(1, n - p)$, but we typically use t results when $\text{rank}(\mathbf{C}) = 1$, as in Results B.10 and B.11.
- These t and F results are used repeatedly throughout linear models (linear regression and ANOVA).
- Again, these results depend on our linear model assumptions, including ASSUMPTION OF NORMAL ERRORS, though, as mentioned, results may be approximately true, with approximations improving as n increases.

Result B.12 (Estimated Standard Error of an Estimator).

- If $\hat{\theta}$ is a scalar estimator, then the estimated standard error

$$\widehat{se}(\hat{\theta}) \equiv \sqrt{\widehat{Var}(\hat{\theta})}.$$

- In other words, the **estimated standard error** of an estimator is just another name for the **estimated standard deviation** of an estimator.
- Note that we often hear just **standard error**, which typically refers to the estimated standard error.

- Typically, the standard error (standard deviation) is discussed only for scalars, as the square root of their variance, but you could define a standard error (standard deviation) square root matrix. (Not common.)
- For us, in the scalar case of $\mathbf{C}\hat{\beta}$ (Result B.10),

$$\widehat{se}(\mathbf{C}\hat{\beta}) = \sqrt{\hat{\sigma}^2 \mathbf{C}^T (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{C}}.$$

- And, more particularly, in the case of $\widehat{\beta}_j$ (Result B.11),

$$\widehat{se}(\widehat{\beta}_j) = \sqrt{\widehat{\sigma}^2 (\mathbf{X}^T \mathbf{X})_{(jj)}^{-1}}.$$

B.6 Joint, Marginal & Conditional Distributions

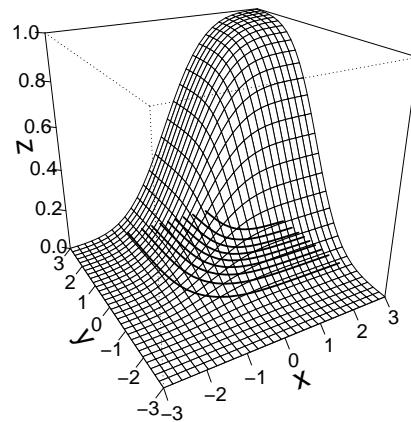
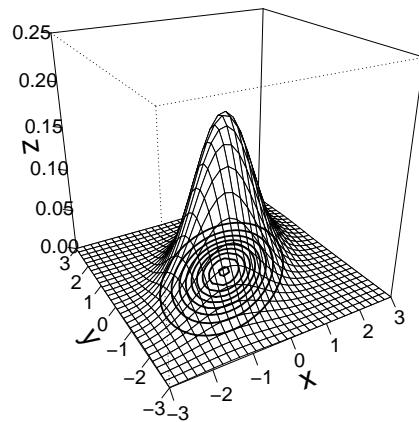
- We gave some summary properties of distributions of random vectors in the previous section, B.3.
- As for scalar random variables (Appendix A.7), we have analogous definitions of (joint) probability distributions (or related functions) of random vectors.

A generic bivariate pdf and associated cdf are shown in the code below. (Actually, they depict a bivariate normal.)

```
> par(mfcol=c(1,2))
> ## bivariate pdf  $N(0,1)$  correlation = 0.5
> x<- y<- seq(-3,3,length=30)
> xygrid<- expand.grid(x,y)
> Sig<- matrix(c(1,.5,.5,1),2,2)
> z<- matrix(mvtnorm::dmvnorm(xygrid, s=Sig),30,30)
> trans<- persp(x,y,z, theta=-30, phi=25, r=3,lwd=.5,
+                  ticktype=c("detailed"), zlim=c(0,0.25),
+                  main="pdf f(x,y)", cex.axis=.7)
> clines<- contourLines(x,y,z)
> invisible(lapply(clines,
+                   function(contour) {
+                     lines(trans3d(contour$x, contour$y, z=0, trans))
+                   }))
> ## cdf F(x,y)
```

```
> z<- matrix(apply(xygrid,1,function(xy,Sig)
+     mvtnorm::pmvnorm(lower=c(-Inf,-Inf), upper=xy, s=Sig),
+     Sig=Sig), 30, 30)
> trans<- persp(x,y,z, theta=-30, phi=25, r=3,lwd=.5,
+     ticktype=c("detailed"), zlim=c(0,1),
+     main="cdf F(x,y)",cex.axis=.7)
> clines<- contourLines(x,y,z)
> invisible(lapply(clines,
+     function(contour) {
+         lines(trans3d(contour$x, contour$y, z=0, trans))
+     }))

```

pdf $f(x,y)$ cdf $F(x,y)$ 

```
> par(mfcol=c(1,1))
```

Definition B.9 (Joint cdf).

- *The joint cumulative distribution function (cdf) of a random vector*

$\mathbf{Y} = (Y_1, \dots, Y_n)^T$ is given by

$$F(y_1, \dots, y_n) = P(Y_1 \leq y_1, \dots, Y_n \leq y_n),$$

where $P(Y_1 \leq y_1, \dots, Y_n \leq y_n)$ is interpreted as the “probability” of the random variables Y_i being less than or equal to some numbers y_i , simultaneously (“jointly”), $i = 1, \dots, n$.

- Note the **lowercase** y_i represent fixed (non-random) values that must be specified in order to get a value for the function F , just like a **typical mathematical function**.
- As we (may or may not have) mentioned in §A.7, this definition seems to imply that the (joint) cdf, F , is defined as a function of some sort of probability function, P , as if P exists first.
- In practice, we typically specify the cdf, F , (or, more likely, the pmf/pdf, below), which induces a corresponding probability function P , which we do not discuss; as we said, we avoid a formal definition of probability.
- In a very real sense, when we refer to a random variable, we are referring to its distribution, and vice-versa.

Definition B.10 (Joint pdf).

- If there exists a function $f(y_1, \dots, y_n)$ such that

$$F(y_1, \dots, y_n) = \int_{-\infty}^{y_1} \cdots \int_{-\infty}^{y_n} f(t_1, \dots, t_n) dt_1 \cdots dt_n$$

is a cdf, then $f(\mathbf{y})$ is called a (joint) **probability density function (pdf)**.

- Our definition of joint **pdf** implies the Y_i are (sounds good) **continuous random variables** (§A.7).
- The analogous definition of the **joint pmf** (probability mass function) for vector of **discrete random variables**, Y_i , (§A.7) **replaces integrals by sums** in the above definition.
- The **support** of \mathbf{Y} is implicitly incorporated into $f(\mathbf{y})$.
- The **analogous definition of the joint pmf** (probability mass function) for a vector of discrete random variables replaces integrals by sums in the above definition.
- In practice, we **typically specify mathematical functions for the joint pdf or joint pmf** and do not work nearly as much with cdfs.
- We might also specify a function corresponding to the joint distribution of a collection of continuous *and* discrete random variables (not much in this class) (In practice, to do this, we often rely on the **multiplication/product rule** (below) and **hierarchical modeling**.)

Generic bivariate pmf and associated cdf (to be illustrated in class).

Example B.24 (Multivariate Normal pdf). *If the $Y_i \sim N(\mu_i, \sigma_i^2)$, independently, $i = 1, \dots, n$, then the (joint) pdf of $\mathbf{Y} = (Y_1, \dots, Y_n)^T$ is*

$$f(y_1, \dots, y_n) = (2\pi)^{-n/2} \prod_i (\sigma_i^{-1}) \exp \left(-\frac{1}{2} \sum_i \left(\frac{y_i - \mu_i}{\sigma_i} \right)^2 \right),$$

which we can write in matrix form as

$$f(\mathbf{y}) = (2\pi)^{-n/2} |\Sigma|^{-1/2} \exp \left(-\frac{1}{2} (\mathbf{y} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{y} - \boldsymbol{\mu}) \right),$$

where $\mathbf{y} = (y_1, \dots, y_n)^T$, $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T$, Σ is the $n \times n$ diagonal matrix with i th diagonal element σ_i^2 , and $|\Sigma|$ denotes the determinant of (square) matrix Σ . We saw the multi-variate normal distribution and likelihood function between note sections §2.4 and 2.5.

- This model results from the assumption of **independence** (below).
- Our job is to specify a **model for the mean**, e.g., for linear models, $\mu_i = \mathbf{x}_i^t \boldsymbol{\beta}$, and/or a **model for the variance**, e.g., often $\sigma_i = \sigma$ (constant sd/var), or more fancy variance models (INF 512 and, perhaps, [Far14, Chap. 8]).

```
> ## E.g., cdf computation for vector of n iid N(0,1) rvs
> n<- 30
> ## F(y_1 <= 2, ..., y_{\{n\}} <= 2) = (indep) Phi(2)Phi(2),...,Phi(2) (n times)
> prod(pnorm(rep(2,n)))
[1] 0.50138

> ## More carefully
> exp(sum(pnorm(rep(2,n), log=TRUE)))
[1] 0.50138
```

```

> ## Or
> library(mvtnorm)
> pmvnorm(lower=-Inf, upper=rep(2,n),
+           mean=rep(0,n), sigma=diag(n))

[1] 0.50138
attr(,"error")
[1] 0
attr(,"msg")
[1] "Normal Completion"

> detach(package:mvtnorm)
>
> ## Or
> mvtnorm::pmvnorm(lower=-Inf, upper=rep(2,n),
+           mean=rep(0,n), sigma=diag(n))

[1] 0.50138
attr(,"error")
[1] 0
attr(,"msg")
[1] "Normal Completion"

```

Definition B.11 (Square Bracket Notation for pdf/pmf).

- Instead of using alphabetical notation, such as $f(\mathbf{y})$, we will often use $[\mathbf{y}]$ to denote the pdf/pmf of random vector \mathbf{Y} .
- e.g., $[\mathbf{x}, \mathbf{y}]$ denotes the joint pdf/pmf of two random vectors \mathbf{X} and \mathbf{Y} .
- [Far14] does not use square-bracket notation.
- Note that we also use \mathbf{X} to denote a matrix of **observed** covariate values. This should cause little confusion as we will typically consider covariates to be fixed in INF 504/511/512.

Generic bivariate joint pdf and associated marginal pdfs (to be drawn in class)

Definition B.12 (Marginal pdf/pmf/distribution).

- If $[x, y]$ is the (joint) pdf (distribution) of two (continuous) random vectors, \mathbf{X} and \mathbf{Y} , then the marginal pdfs (distributions) of \mathbf{X} and \mathbf{Y} are

$$[x] = \int [x, y] dy,$$

and

$$[y] = \int [x, y] dx.$$

- Multiple integration implied.
- For discrete random vectors (pmfs), sums replace integrals.
- Don't get confused: the marginals are joints for their sub-vectors, too, right?

Generic bivariate joint pdf and associated conditional pdfs (to be drawn in class)

Definition B.13 (Conditional pdf/pmf/distribution).

- If $[x, y]$ is the (joint) pdf (distribution) of two random vectors, \mathbf{X} and \mathbf{Y} , then the conditional pdfs (distributions) are

$$[x | y] = \frac{[x, y]}{[y]}$$

and

$$[y | x] = \frac{[x, y]}{[x]}.$$

- A **conditional distribution** describes the distribution of one random vector for a given ("|" conditional on) fixed value of the other variable/vector
- Notice the marginal in the denominator ensures that the conditional integrates (or sums) to one.
- We denote the associated **conditional random vectors/variables** as $\mathbf{X} | \mathbf{y}$ and $\mathbf{Y} | \mathbf{x}$, respectively.
- Don't get confused: the conditionals are joints for their sub-vectors, too, right?

- Joint, marginal and conditional distributions are **distributions like any others** with corresponding random vectors, like scalar distributions and random variables (Appendix A).
- As such, these distributions have **properties like any other distribution, e.g., means, variances, covariances, etc.**, as we discussed in §B.3, above.

- We're very close to **Bayes' theorem**, but we'll wait until our context is Bayesian statistics before discussing this theorem.
- **Model Building.** We typically specify models for conditional distributions in the process of building up joint distributions instead of computing conditional distributions from joints, as if someone hands us the joint distribution.

Definition B.14 (Multiplication/Product Rule for Joint pdf/pmf/distribution).

If $\mathbf{Y} = (Y_1, \dots, Y_n)$ is a random vector with (joint) distribution, $[y] = [y_1, \dots, y_n]$, then

$$[y_1, \dots, y_n] = [y_1][y_2 | y_1][y_3 | y_1, y_2] \cdots [y_n | y_1, y_2, \dots, y_{n-1}],$$

for any ordering of the Y_i .

- This implies that, if we have a joint distribution, then (in principle) it factors into any one of $n!$ (factorial...a lot for most any n) possible products of marginal and conditional distributions. One joint, many possible ways to factor. Gee, that's mildly interesting.
- A more practical implication of the multiplication rule is that we can build a joint distribution from a product of marginal and conditional distributions. This seems much more useful for our purposes and is the essence of **hierarchical modeling** (aka **multi-level modeling**).
- Typically, in practice, we specify only one factorization to build a joint distribution, and, typically, it is difficult to specify two or more different factorizations that correspond to the same joint model!

Definition B.15 (Independent Random Variables).

Y_1, \dots, Y_n are said to be independent random variables if their **joint distribution factors into the product of (scalar) marginals**, i.e., if their joint distribution, $[y] = [y_1, \dots, y_n]$, can be written as

$$[y_1, \dots, y_n] = [y_1][y_2][y_3] \cdots [y_n]$$

- This is a special case of the **multiplication/product rule** where conditional distributions do not actually depend on the conditioning variables, i.e., $[y_i | y_1, \dots, y_{i-1}] = [y_i]$
- Independence is by far the most common way to **build a joint distribution** as we are often much more comfortable with thinking about distributions of individual random variables rather than of vectors of random variables.
- The **joint normal distribution example**, above, was obtained via the assumption of independence.

B.7 Conditional Distribution Model Specification

- Considering our linear model, if we specify a distributional model only for the **conditional random variable $\mathbf{Y} | \mathbf{x}$** (or $\mathbf{Y} | \mathbf{x}, \boldsymbol{\beta}, \sigma^2$, if we want to be explicit about dependence (linear) model parameters, too), then we are implicitly assuming the **joint distribution** factors into the form

$$[\mathbf{y}, \mathbf{x} | \boldsymbol{\beta}, \sigma^2, \gamma], = [\mathbf{y} | \mathbf{x}, \boldsymbol{\beta}, \sigma^2][\mathbf{x} | \gamma].$$

- That is, traditional (non-)linear modeling, including “regression and ANOVA,” implicitly **assumes that the distribution of the covariates does not depend on, hence does not inform, the parameters**

β and σ^2 , hence we may ignore the distribution, $[x | \gamma]$, for purposes of inferring β (and σ^2) (else we may be throwing away information about β and will essentially be estimating a different parameter (despite having the same symbol)).

Definition B.16 (Regression Function).

The **conditional mean** of $Y | x$ (i.e., of its distribution) is denoted as

$$E(Y | x),$$

and is called the **regression function**.

- This is our target, f , at least for our additive error model, $y = f(x) + \epsilon$, and our least squares criterion. See the unnumbered Introduction to our note chapter 2 and [HTF01, §2.4]
- That is, the thing we seek to model is $f(x) = E(Y | x)$.
- Of course, as we know, for much of what we do, our regression model will be linear, that is, we will often assume a model for the regression function of the form

$$E(Y | x) = x^t \beta,$$

though we are aware of **model bias**, and we will want to check our model to help ensure that it is not terribly biased (along with other checks).

- Similarly, we may denote the **conditional variance** (variance of the conditional distribution) as $\text{Var}(Y | x)$, for which we will often assume the simple **constant variance model**,

$$\text{Var}(Y | x) = \sigma^2;$$

we hope to say more about more interesting models for the variance function, later ([Far14, Chap. 8] and INF 512).

Appendix C

Bayesian Linear Model

Contents

C.1	Introduction	650
C.1.1	Data Distribution	650
C.1.2	Bayes Theorem: Data, Prior, Joint, Posterior $[\boldsymbol{\theta} \mathbf{y}]$	651
C.1.3	Prior Predictive $[\mathbf{y}]$	654
C.1.4	Posterior Predictive $[\mathbf{y}^* \mathbf{y}]$	655
C.1.5	Summary	658
C.1.6	Remarks	659
C.2	Linear Model	660
C.2.1	Overview	661
C.2.2	Conditional Normal Prior & Posterior for $\boldsymbol{\beta} \sigma^2$	662
C.3	Conjugate Prior	665
C.3.1	Posterior	667
C.3.2	Marginal Posterior for $\boldsymbol{\beta}$ is a t	670
C.3.3	Posterior Predictive is a t	672
C.3.4	Remarks	674
C.4	A Common Improper Prior	675
C.4.1	Posterior	676
C.4.2	Marginal Posterior for $\boldsymbol{\beta}$ is a Familiar t	676
C.4.3	Posterior Predictive is a Familiar t	677
C.5	STAT 101 Redux a la Bayes	678
C.5.1	t -based Intervals for β_j	679
C.5.2	t -based Test for β_j	681
C.5.3	t -based Intervals for $E(Y \mathbf{x}) = \mathbf{x}^t \boldsymbol{\beta}$	683

C.5.4	<i>t</i> -based Prediction Intervals for $Y \mathbf{x}$	684
C.6	Prostate Data Example with Improper Prior	685
C.6.1	Frequentist R Summary	687
C.6.2	Bayesian Summary	687
C.7	A Common Independence Prior	692
C.7.1	Full Conditional Posterior Distributions	693
C.8	2-Stage Gibbs Sampling	695
C.9	3-Stage Gibbs Sampling: Prostate Data Example with “Combo” Prior	696
C.9.1	Eliciting a Prior	697
C.9.2	Full Conditionals	698
C.9.3	Gibbs Sampling Algorithm	701
C.9.4	Gibbs Sampling Code	702
C.9.5	Posterior Convergence Diagnostics with <code>coda</code>	705
C.9.6	Posterior Summaries with <code>coda</code>	711
C.9.7	Regression Function and Posterior Predictive	714
C.10	HMC in Stan: Prostate Data Example with “Combo” Prior	716
C.10.1	Functions Block	717
C.10.2	Data Block	717
C.10.3	Transformed Data Block	718
C.10.4	Parameters Block	718
C.10.5	Transformed Parameters Block	718
C.10.6	Model Block	718
C.10.7	Generated Quantities Block	719
C.10.8	Altogether for Stan	719
C.10.9	Translate Stan to C++ with <code>stan_c</code>	721
C.10.10	Make an Executable Stan Model with <code>stan_model</code>	721
C.10.11	Data List for Stan	722
C.10.12	List of Initial Value Lists for Stan	722
C.10.13	Executing a Stan Model with <code>sampling</code>	723
C.10.14	Posterior Convergence Diagnostics with <code>coda</code>	723
C.10.15	Posterior Summaries with <code>coda</code>	732
C.11	Prostate Data Example Summary	733
C.12	Other Priors	737

Main Objectives:

- Bayesian linear model
 - Bayes theorem, prior distribution, posterior distribution, prior predictive distribution, posterior predictive distribution, conjugate prior, improper prior, full conditional distribution.
 - Shrinkage
 - Gibbs sampling
 - Stan and **rstan**
-
-

Reading:

[Bis06, §3.3], [Mur12, §4.4], [Wak13, §3.2, 3.4, 3.7, 3.8, 3.12, 4.4, 4.11, 5.12, 5.13] R

C.1 Introduction

Now is a good time to review §B.6 Joint, Marginal & Conditional Distributions. This Appendix relies heavily on that section, but uses notation more typical of a Bayesian context.

C.1.1 Data Distribution

- **Data Distribution: A Conditional Distribution.** Let \mathbf{Y} be a random vector of **observables** (responses/outputs) whose distribution depends on a vector of parameters, generically denoted as $\boldsymbol{\theta}$ for the moment: we may write the random vector as $\mathbf{Y} | \boldsymbol{\theta}$ and denote its (conditional) distribution as

$$[\mathbf{y} | \boldsymbol{\theta}].$$

- **Sometimes Called Sampling Distribution.** We will sometimes use the term ‘sampling distribution’ for our data distribution (normal (aka, Gaussian) for us), a term which is typically used with little more than coincidental connection to our notion of ‘sampling distribution’ in our note chapter 5 ([Far14, §3.4]); in the current context, it’s just another name used for our data distribution.
- **Often Called Likelihood.** We will also refer to our data distribution as the likelihood of our data. If we view the data distribution as a function of the parameters, data held fixed at their observed values, we get the

likelihood function. See the unnumbered section after §2.4 for more on the normal likelihood.

- **E.g., Our Classical Linear Model.** More in class.
- **Conventional Notation?** It's conventional, among Bayesian statisticians, to explicitly condition on random variables (θ is now an rv...see below) and to omit conditioning on fixed quantities, like covariate/input x , in a linear model, but this convention is not universally followed, particularly outside of the Bayesian statistical literature.

C.1.2 Bayes Theorem: Data, Prior, Joint, Posterior $[\theta | y]$

- **Prior and Data Give Joint.** If we specify a **prior distribution** (a marginal distribution in the current context),

$$[\theta],$$

then, with the data (conditional) distribution, we can **build a joint distribution using the multiplication (aka product) rule** (Definition B.14)

$$[y, \theta] = [y | \theta][\theta].$$

(Again, see §B.6.)

- **BTW, Probabilistic Software.** This joint distribution is what users specify in essentially all probabilistic programming languages that are used for Bayesian inference (e.g., BUGS, WinBUGS, OpenBUGS, JAGS, Stan, NIMBLE). Thus, in some sense, once this is specified, Bayesian inference follows, in principle, though some finessing may be required.
- **Informatics PhD.** What should we expect from an informatics PhD? More than just a user of a probabilistic programming language?

- **From Joint to Conditional (Holy Grail) via Bayes Theorem.** From this joint distribution, we can (in principle) use the definition of a conditional distribution (Def. B.13) to get our desired (**posterior**) conditional distribution as

$$[\boldsymbol{\theta} | \mathbf{y}] = \frac{[\mathbf{y}, \boldsymbol{\theta}]}{[\mathbf{y}]}, \quad \text{or}$$

$$[\boldsymbol{\theta} | \mathbf{y}] = \frac{[\mathbf{y} | \boldsymbol{\theta}][\boldsymbol{\theta}]}{[\mathbf{y}]}.$$

See [Wak13, Expr. (3.1)]. This result is known as **Bayes Theorem**, and we sometimes hear the term **inverse probability** because it “inverts” the data (probability (density)) distribution, $[\mathbf{y} | \boldsymbol{\theta}]$ to the posterior (probability (density)) distribution, $[\boldsymbol{\theta} | \mathbf{y}]$. (Notice how close we were to Bayes Theorem in Definition B.13.)

- **Prior Predictive, Another Marginal.** In principle, we obtain the marginal $[\mathbf{y}]$, called the prior predictive distribution, via the definition of marginal distribution (Def. B.12); we just sum or integrate out the remaining variables from the joint distribution.
- **Kernel.** More to the point, while the prior predictive, $[\mathbf{y}]$, ensures that the posterior integrates/sums to a total probability of 1, it is a constant wrt $\boldsymbol{\theta}$; thus, it does not help to distinguish among different values of $\boldsymbol{\theta}$. In this sense, it is not a necessary part of the **kernel** (the “important factor”) of the posterior distribution but is part of the posterior’s **normalizing constant**, which, if needed, can be obtained, in principle, integrating out all other variables besides \mathbf{Y} in the joint to get the marginal $[\mathbf{y}]$, perhaps scaled by other constants wrt $\boldsymbol{\theta}$ if the joint is not normalized (essentially using Def. B.12).
- **Recognizing The Kernel.** Knowing that we’re looking for a distribution of $\boldsymbol{\theta}$, we may look at the joint in the numerator of Bayes theorem (ignoring the normalizing constant $[\mathbf{y}]$ or any other multiplicative constants wrt $\boldsymbol{\theta}$) to see if we recognize the kernel of a distribution in $\boldsymbol{\theta}$. “Hey, that factor looks like it belongs to a normal distribution for $\boldsymbol{\theta}$ (up

to a normalizing constant that does not depend on θ)!" In this case, we're done, up to summarizing our posterior, at least. We don't have to integrate (or sum) to get the normalizing constant to get the posterior.

- **Classical Bayesian linear models are relatively simple**, and we (someone) often recognize (conditional) kernels to give us a posterior of familiar form (e.g., normal or t or inverse gamma) or give us familiar full conditional posterior distributions ('full conditionals') which are a step toward the full posterior.
- **Otherwise, Typically, Not So Simple.** For all but the simplest situations, **we may not recognize the kernel** of the posterior. We may think then that we must somehow use numerical quadrature methods to integrate (sometimes in very high dimensional space) or to otherwise numerically approximate integrals (see, e.g., [Wak13, §3.7]) in order to obtain $[y]$ for use in our original statement of Bayes Theorem. But...
- **MCMC To the Rescue.** While there exist several methods to compute or approximate the posterior distribution—these methods somehow seen as approximations to the integral/sum necessary to get the normalizing constant—we will focus more on Markov chain Monte Carlo (**MCMC**) methods, which effectively get around such integration. (We may view (MC)MC methods as integration methods, but, as often implemented, (MC)MC tends to obscure an integration perspective.)
- **Normalizing Contant Not Necessary.** In other words, most MCMC methods do not require the normalizing constant, $[y]$ (or other multiplicative constants), to effectively reproduce the posterior $[\theta | y]$. (Poor normalizing constant, almost no one needs you...)
- **Numerator of Bayes Theorem.** Thus, in this spirit of ingoring constants, we see the tendency to express Bayes Theorem as

$$[\theta | y] \propto [y | \theta][\theta],$$

which we often hear as

"posterior is proportional to likelihood times prior",

which, to reiterate, is almost always the main ingredient in all popular probabilistic programming languages.

C.1.3 Prior Predictive $[y]$

- **Prior Predictive.** As, mentioned, above, we have (in principle) the **marginal distribution of the data** (part of the normalizing constant of the posterior),

$$[y] = \int [y | \theta][\theta]d\theta,$$

which is also referred to as the prior predictive distribution, because it is (in principle) the distribution we would use to infer about (predict) our data, y , before we observe it, after integrating/summing/averaging out/spreading over the unknown parameter θ , thus accounting for uncertainty of the quantities we don't know. It depends on the hyper-parameters of the prior distribution.

- **Model Assessment.** The prior predictive is sometimes used to assess models (likelihood and prior, i.e., the joint model discussed above) by comparing it to observed data. Intuitively, if we plug in our actual, observed data, y , into $[y]$, and we get an unusually small value, then this suggests that we have not done a good job, a priori, of predicting our data, suggesting remodeling of our likelihood or prior or both. Or, similarly, we can plot observed data together with fake data generated from $[y]$ to see if the observed is somehow consistent with the generated data.
- **Type II Maximum Likelihood.** Also, the prior predictive distribution is sometimes used in an **empirical Bayes** analysis to specify a prior distribution in a so-called type-II maximum likelihood approach ([Ber85]): While we have integrated/summed/averaged over θ to obtain $[y]$, $[y]$ still contains the ("hyper") parameters of the prior distribution, $[\theta]$, and

these need to be given particular values. (If you specify a normal prior distribution for θ , you'll need a prior mean and a prior (co)variance, or a (hyper) prior for these...etc.) Thus, we may view $[y]$ as a likelihood to be maximized, the resulting estimates of the hyperparameters plugged into $[\theta]$, which is then used in a subsequent (approximate, technically not fully Bayesian) analysis. Incidentally, **machine learners** sometimes refer to this empirical Bayes approach to prior specification as **evidence approximation** ([Bis06, §3.5]). Empirical Bayes / evidence approximation is strictly not Bayesian because prior distributions, including their parameters, should be fully specified in Bayesian analysis *a priori*, before we look at the data. Type II maximum likelihood uses the data to get “prior” distribution parameters, then we use the data again in Bayes theorem; this double use of the data is technically not strictly (subjective) Bayesian. But, not many practitioners care about this relatively philosophical distinction.

C.1.4 Posterior Predictive $[y^* | y]$

- **More Unknowns to Infer.** As we considered briefly in the frequentist linear model (chapter 4), we may want to predict an as yet unobserved (i.e., unknown) response/output (vector), \mathbf{Y}^* , in addition to or perhaps with less emphasis on the parameter θ , discussed above.
- **“Prediction”.** We often hear of “prediction” used loosely to refer to estimation of

$$E(\mathbf{Y} | \mathbf{x}),$$

or to prediction of

$$\mathbf{Y} | \mathbf{x}$$

(as in [Far14]). The posterior predictive refers to (unobserved) $\mathbf{Y} | \mathbf{x}$ values, not (directly) to the posterior of $E(\mathbf{Y} | \mathbf{x})$ ((our model of) which

is a function of parameters, θ , so we can get its posterior, in principle, if we want it).

- **Notation.** We will use

$$\mathbf{y}^* \text{ and } \mathbf{Y}^*$$

to distinguish unobserved responses from (to be) observed responses \mathbf{y} and \mathbf{Y} ; in chapter 4, we use \mathbf{x}_0 in $Y | \mathbf{x}_0$ to indicate prediction of a new response, but, as mentioned, we now tend to suppress notation of fixed quantities, like \mathbf{x} or \mathbf{x}_0 ; we may use an occasional \mathbf{x}^* here and there.

- **Observables and Unobservables Jargon.** We might say that Y and Y^* are (potential) observables, with Y^* being unobserved, but that parameters, θ , are unobservable. (We'd have to observe an entire population of values of Y to know their mean, θ .)
- **Joint Posterior.** We might simply include such additional unknowns simply as part of θ , or alongside θ , thus implying we want a joint posterior $[\mathbf{y}^*, \theta | \mathbf{y}]$. (We skip Bayes Theorem in the context of this more general introduction, here, but state it in the context of our linear model in §C.3.3, below.)
- **Hierarchically.** In practice, we often view the joint distribution. $[\mathbf{y}^*, \theta | \mathbf{y}]$, via the multiplication rule (Definition B.14), as

$$[\mathbf{y}^*, \theta | \mathbf{y}] = [\mathbf{y}^* | \mathbf{y}, \theta] [\theta | \mathbf{y}].$$

- **A Marginal Posterior.** Then, in principle, we obtain the posterior predictive by application of the definition of marginal distribution (Definition B.12),

$$[\mathbf{y}^* | \mathbf{y}] = \int [\mathbf{y}^* | \mathbf{y}, \theta] [\theta | \mathbf{y}] d\theta,$$

where $[\mathbf{y}^* | \mathbf{y}, \theta]$ is the distribution of unobserved data (unobserved response/outputs) \mathbf{y}^* conditional on (observed) \mathbf{y} , which we can write as $[\mathbf{y}^* | \mathbf{y}, \theta] = [\mathbf{y}^* | \theta]$, if \mathbf{y}^* and \mathbf{y} are independent; as in, e.g., the classical linear model. See [Wak13, Expr. (3.9)].

- (And, in this joint context, we see our previous $[\theta | y]$ as a marginal (conditional!) posterior, too.)
- **Sometimes Closed Form.** In simple situations, we may be able to perform the integral analytically to get an expression for the distribution of the posterior predictive, which we may recognize as a standard distribution (e.g., t). Great, we're done, up to summarizing the distribution (means, variances, etc.).
- **More Often Not Closed Form.** But, more typically, for all but the simplest models, we cannot obtain a close-form expression, beyond the integral expression, for the posterior predictive. While there are (approximate) numerical integration techniques, we will focus on MCMC procedures, which, practically speaking, allow us to avoid evaluating the integral (well, technically, we more/less evaluate it via MC(MC) integration, but the integration is sort of hidden).
- **Composition Sampling.** Notice, because $[y^* | y, \theta]$ (or, with independence, $[y^* | \theta]$) typically follows from the specification of the data model (e.g., regression model), this gives us a relatively easy way to obtain (samples from) the posterior predictive. That is, if we can obtain samples of θ from the (marginal conditional!) posterior $[\theta | y]$ (from MCMC, for example), and if we know the conditional $[y^* | y, \theta]$ (again, it should be obvious from our data model!), then we can generate $y^* | \theta, y$ so that we obtain samples of $(y^*, \theta | y)$ from the joint posterior...one $\theta | y$ then one $y^* | \theta, y$, another $\theta | y$ then another $y^* | \theta, y$, etc. If you want a sample from the (marginal posterior), $[y^* | y]$ (our posterior predictive), then just ignore the sample of $\theta | y$ values (and vice-versa)! This process sometimes called composition sampling; see [[Wak13](#), §3.8.4].
- Again, **linear models are relatively simple** in that we often have closed forms for posterior predictives (t distributions), but, still, we may use MCMC or composition sampling anyway as a convenient way to get posterior summaries, mean, variance, intervals, etc., for y^* .

C.1.5 Summary

- In short,
 - We specify the **data distribution (likelihood)**, $[y | \theta]$ or $[y^*, y | \theta]$ (a conditional distribution of the data given parameters, perhaps with unobserved observables), and we specify the
 - **prior** (a marginal distribution of the parameters of the data model).
 - **Joint Model, Numerator of Bayes theorem.** Together, the likelihood and prior define a joint probability model in the numerator of Bayes theorem, which is the focus of modeling for probabilistic programming languages.
 - From the joint specification, we obtain, in principle, the **posterior (predictive)** (a conditional distribution of the parameters (and/or unobserved outputs) given the observed data) via Bayes theorem, up to a normalizing constant anyway.
- **Kernel.** Given our discussion of the kernel as the “important part”, if we recognize the kernel, we’re essentially done; we can avoid having to compute the integral/sum to get the normalizing constant, and we can use existing results to summarize the posterior. But, again, this requires that we recognize the form of the kernel to correspond to a known distribution, perhaps, e.g., normal, normal-gamma or t, from which we could compute easier-to-grasp summarizing quantities for our posterior, like means, medians, modes, quantiles, intervals, etc., using ordinary methods implemented in many softwares. Sampling from the posterior is not necessary in this case but many people may find it convenient to approximate such posterior quantities with corresponding sample quantities, with the approximation improving with the size of the sample obtained from the posterior.
- **Other Methods to Get at the Posterior.** When we fail to recognize kernels, we avail ourselves of other methods, such as MCMC or integrated nested Laplace approximation (INLA), which usually do not

require the normalizing constant, hence the focus on the joint model in the numerator of Bayes theorem.

- **Posterior Predictive.** Similarly, in simple cases, the posterior predictive may be obtained in closed form by performing analytical integration. But, most often, we obtain samples from the posterior predictive via **composition sampling**; failing that, via MCMC or other method.

C.1.6 Remarks

- **Fixed, Unknown Parameters.** Until this appendix, we have conceptualized the unknown parameters (e.g., β or σ^2) in our data distribution as **unknown but fixed** quantities to be estimated, with **uncertainty** in estimation following from **sampling distribution theory** (in simple cases, e.g. [Wak13, p. 29 & Ch. 5] and much of our previous notes involving t and F distributions) or from **asymptotic theory** (e.g., much of [Wak13, Ch. 2]).
- **Characterize Uncertainty About Parameters via Randomness.** In a Bayesian analysis, we take a different tack, and we characterize our **uncertainty** about unknown parameters more directly via a **prior distribution** (a marginal distribution) and a **posterior distribution** (a conditional distribution) for the parameter itself (rather than with the distribution of an estimator of the fixed parameter, as in our t and F material so far). E.g., [Wak13, Ch. 3].
- **Conceptually Fixed, But Formally Random.** In other words, we may still conceptualize the parameters as fixed, but our accounting of uncertainty of the fixed quantities is formalized by probability distributions for parameters. That is, we treat the parameters as random as a way to characterize our uncertain about them.

- **Bayesian Holy Grail.** In a Bayesian analysis, the goal is often to obtain the conditional distribution

$$[\boldsymbol{\theta} | \mathbf{y}].$$

- **Infer Unknowns Given What You Know.** We have a distribution of our unknown quantities to use for inferring about the unknown quantities, with an accounting of information in our data (via the likelihood as we will see) and information in (\mathbf{y})our prior distribution (as we will see). This seems like an agreeable object to have for making inference. (Of course, it's all based on the assumption that our model, now including a prior distribution, is somehow "correct!")
- **Broadly Speaking.** More generally, we might consider our goal to be the distribution of

$$[unknowns | knowns],$$

whether the unknowns are unobservable parameters or unobserved but potentially observables. Given our discussion so far, we may denote our goal as $[\mathbf{y}^*, \boldsymbol{\theta} | \mathbf{y}]$ (a **joint posterior** of sorts).

- (We do not formally pursue Bayesian decision theory, whereby we add another element—a loss function—and decisions or actions or estimates are typically the target. However, we may point out typical estimates and their corresponding loss function as we go: mean and squared error loss, median and absolute error loss).

C.2 Linear Model

Here, we repeat, more or less, the previous section's generic presentation in the context of our familiar linear model and notation.

C.2.1 Overview

- **Data Distribution.** Recall our linear model from previous chapters,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N(\mathbf{0}, \Sigma),$$

where, for nearly all of what we do, $\Sigma = \sigma^2 \mathbf{I}$. Thus, following our generic Bayesian discussion, above, we write our data distribution in the current context as

$$[\mathbf{y} | \boldsymbol{\beta}, \sigma^2].$$

What's θ in our regression context?

- **Conventional Notation?** In the classic Bayesian linear model, as in the frequentist version, \mathbf{X} (or a generic row \mathbf{x}) will be considered known (see §B.7), and we may not use notation to explicitly indicate conditioning on \mathbf{x} or \mathbf{X} .
- **Factorization Again.** This conditioning on inputs and not giving the inputs a distribution—typical for classical linear regression—assumes a factorization of the prior analogous to that for $[\mathbf{y}, \mathbf{x}]$, discussed in §B.7, so that we do not have to consider the prior for the parameters of the covariate distribution. We skip further discussion of this.
- **Prior.** Continuing to follow the above, generic Bayesian development, we must specify a prior distribution for the parameters, which, in the current context, we denote as

$$[\boldsymbol{\beta}, \sigma^2].$$

- **Posterior.** We will discuss common prior distributional specifications and, for each, a corresponding posterior distribution, which, in our current context, we denote as

$$[\boldsymbol{\beta}, \sigma^2 | \mathbf{y}],$$

or as

$$[\mathbf{y}^*, \boldsymbol{\beta}, \sigma^2 | \mathbf{y}],$$

if we are also interested in predicting unobserved response, \mathbf{y}^* .

- **Posterior Predictive.** We will discuss the corresponding posterior predictive distribution, which we denote now as (some of the same notation as before I suppose)

$$[\mathbf{y}^* | \mathbf{y}],$$

where \mathbf{y}^* denotes a vector of n^* unobserved outputs, with associated inputs in a matrix, \mathbf{X}^* , that we would like to predict.

- **Again, Suppressed Notation.** Again, we tend to suppress notation on covariates, \mathbf{x} , \mathbf{X} or $\mathbf{x}^* \mathbf{X}^*$.
- **Again, Slight Change of Notation.** In our previous, frequentist material, we didn't put stars * on values to be predicted or their associated covariates, but perhaps we should have done this for consistency in notation.

C.2.2 Conditional Normal Prior & Posterior for $\beta | \sigma^2$

- **Generic Hierarchical Prior.** By the multiplication/product rule (Def. B.14), we can write the prior hierarchically,

$$[\beta, \sigma^2] = [\beta, | \sigma^2][\sigma^2].$$

- **(Conditional) Normal Prior for β .** We specify a normal prior for the regression function parameters,

$$\beta | \sigma^2 \sim N(\mathbf{m}_0, \Sigma_0).$$

In practice, $[\beta | \sigma^2]$ may/may not actually depend on σ^2 , despite notation here; more as we go.

- **Pay Attention.** This conditional normal prior (and posterior) of this section shows up (often unannounced) in many different contexts (with minor specialization here and there), e.g., ridge regression, regularized regression splines, model selection and averaging, machine learning 'evidence approximation', etc.

- **Bayes Theorem & Conditional Posterior for β .** Conditioning on σ^2 , for now, Bayes theorem gives the conditional posterior,

$$[\beta | \sigma^2, \mathbf{y}] = \frac{[\mathbf{y} | \beta, \sigma^2][\beta | \sigma^2]}{[\mathbf{y}]}.$$

In the current case (omitting details that you can undoubtedly find in many books and Internet memes), we can recognize the (conditional) kernel to indicate, specifically, the conditional posterior,

$$[\beta | \sigma^2, \mathbf{y}] = N(\hat{\mathbf{m}}, \sigma^2 \hat{\Sigma}),$$

with **mean & variance**,

$$\begin{aligned}\hat{\mathbf{m}} &= (\Sigma_0^{-1} + \sigma^{-2} \mathbf{X}^t \mathbf{X})^{-1} (\Sigma_0^{-1} \mathbf{m}_0 + \sigma^{-2} (\mathbf{X}^t \mathbf{X}) \hat{\beta}), \\ \sigma^2 \hat{\Sigma} &= (\Sigma_0^{-1} + \sigma^{-2} (\mathbf{X}^t \mathbf{X}))^{-1}.\end{aligned}$$

- **(Conditional) Conjugacy & Parameter Update Formulas.** Note that the (conditional) prior for β is normal, and the conditional posterior (given σ^2) is normal (whether the conditional prior actually depends on σ^2 as a scalar multiple in the prior variance or not). This is an example of (conditional) conjugacy, where the (conditional) posterior is of the same distributional form as the (conditional) prior—normal in this case. See Def C.1 below for more on conjugacy. Conjugacy allows us to use simple parameter update formulas to get from our prior to our posterior; the result of Bayes theorem has largely been obtained for us with relatively little effort (just some algebra, which you will undoubtedly find all over the Internet). While full conjugacy is relatively rare in Bayesian models, conditional conjugacy occurs relatively often, and is sometimes used to permit so-called ‘Gibbs steps’ using the resulting conditionally conjugate (full) conditional posterior distributions in MCMC sampling.
- **Notice a few things:**
 - **Conditional Posterior Mean.** The conditional normal posterior mean of β (given σ^2) is a **weighted average** of the prior mean of

β and a data-based (unbiased) estimate of β . Notice the **shrinkage** toward the prior mean, aka **borrowing of strength** of the posterior (mean) from the prior (mean), when the data are relatively weak (variable) (or the prior is relatively strong (precise)).

- **Conditional Posterior Variance.** The conditional normal posterior variance of β (given σ^2) is the **inverse of the sum of its prior precision (inverse of variance) and the precision of a (hopefully familiar!) data-based estimate**. Notice that the variance gets small as either the data or prior are precise (large precision matrices).

$$\begin{aligned}\widehat{\boldsymbol{m}} &= (\mathbf{I} - \mathbf{W})\boldsymbol{m}_0 + \mathbf{W}\widehat{\boldsymbol{\beta}}, \\ \widehat{\boldsymbol{\Sigma}} &= \mathbf{W}(\mathbf{X}^t\mathbf{X})^{-1} \\ \mathbf{W} &= (\sigma^2\boldsymbol{\Sigma}_0^{-1} + \mathbf{X}^t\mathbf{X})^{-1}(\mathbf{X}^t\mathbf{X})\end{aligned}$$

- **This Prior is Used A Lot.** To reiterate, this conditional posterior is used a lot and comes in particular forms depending on the particular values chosen for the prior mean and variance:

- **A more specific conditional prior:** $\boldsymbol{\Sigma}_0 = \sigma^2\boldsymbol{\Sigma}_{00}$; see §C.3 Conjugate Prior, below. You may often see $\boldsymbol{m}_0 = \mathbf{0}$ and $\boldsymbol{\Sigma}_{00} = \mathbf{I}$.
- **Zellner's g -prior** (another, specific conditional prior): $\boldsymbol{m}_0 = \mathbf{0}$, $\boldsymbol{\Sigma}_0 = \sigma^2 g(\mathbf{X}^t\mathbf{X})^{-1}$, used in Bayesian model selection (BMS) and averaging (BMA). (We may skip this in our classes, but I may give you notes in INF 504.)
- **A common shrinkage prior** $\boldsymbol{m}_0 = \mathbf{0}$ and $\boldsymbol{\Sigma}_0 = \frac{\sigma^2}{\lambda}\mathbf{I}$, used in ridge regression, a special case of generalized ridge regression used in regularized regression splines. INF 504.
- **Improper (Jefferey's) prior:** $[\beta | \sigma^2] \propto 1$ (no prior mean or variance); this gives a similar conditional normal posterior but is somewhat out of place here unless we consider this flat prior as the limiting result of a flattened normal distribution as its variance “goes to infinity” (conjugacy in a limiting sense). We consider this prior in §C.4

- [C.6](#), using the Galapagos island biogeography data and the body fat data to illustrate.
- **Independent prior:** \mathbf{m}_0 and Σ_0 do not depend on σ^2 (i.e., β is, a priori, independent of σ^2). We'll see an example of this sort of prior in [§C.7 - C.9](#) in an example using the prostate data (if we haven't already seen those data). Often, $\mathbf{m}_0 = \mathbf{0}$ and $\Sigma_0 = \sigma_\beta^2 \mathbf{I}$ (σ_β^2 is not σ^2).
- **Ahead.** In subsequent sections, we consider some of these particular conditional priors for β , and we complete our prior specification with a prior for σ^2 , largely ignored for the moment. In each case, we will find that the conditional posterior for $\beta | \sigma^2$ is a special case of that considered in this section, above. Of course, we will get to the (perhaps conditional) posterior for σ^2 , too, thus getting at the entire posterior (of parameters anyway), one way or another.

C.3 Conjugate Prior

Definition C.1 (Conjugate Family). A family [or set or class], \mathcal{F} , of [prior] probability distributions on [parameter space or support] Θ is said to be conjugate (or closed under sampling) for a likelihood function $[x | \theta]$ $[\theta \in \Theta]$ if, for every [prior distribution] $[\theta] \in \mathcal{F}$, the posterior distribution $[\theta | x]$ also belongs to \mathcal{F} .

- This is just a restatement of the relatively abstract definition [[Rob01](#), Def. 3.3.1]; I added the terms in square brackets for some context.
- The likelihood and corresponding conjugate prior are sometimes called a **conjugate pair**.
- Notice, we could envision the family, \mathcal{F} , to consist of *all* distributions, but this seems trivial and useless.

- The **conjugate prior distribution** for the mean parameter, β , and variance parameter, σ^2 , considered collectively in the normal linear model, is the product (using the multiplication (product) rule in Definition B.14) of a **conditional normal distribution** and a **scaled inverse-chi-square**,

$$\begin{aligned} [\beta, \sigma^2] &= [\beta | \sigma^2][\sigma^2] \\ &= N(\mathbf{m}_0, \sigma^2 \Sigma_0) \times \text{inv-}\chi^2(\nu_0, \sigma_0^2) \end{aligned}$$

where we must specify values (or further distributions—not here) for the prior **(hyper)parameters**,

$$\begin{aligned} \mathbf{m}_0 &= \text{conditional prior mean,} \\ \sigma^2 \Sigma_0 &= \text{conditional prior variance,} \\ \nu_0 &= \text{marginal prior degrees of freedom} \\ \sigma_0^2 &= \text{marginal prior scale (squared).} \end{aligned}$$

See [Wak13, §3.7.1 & p. 223] for a brief discussion of conjugacy.

- As we said in §C.2, above, the conditional normal prior for $\beta | \sigma^2$ is used a lot. The conditional posterior, here, can be obtained from the expressions for the posterior in §C.2.2 using the particular conditional prior variance matrix $\sigma^2 \Sigma_0$, here, in place of the more generic conditional posterior variance Σ_0 of §C.2.2.
- Reparameterization of Inverse Gamma.** Incidentally, $\text{inv-}\chi^2(\nu_0, \sigma_0^2)$ is the same as $\text{inv-gamma}(a = \nu_0/2, b = \sigma_0^2 \nu_0/2)$, which is a popular parameterization for the prior of σ^2 .
- Intuitive Names.** Shortly, we will see more intuition for why ν_0 and σ_0^2 are called degrees of freedom and scale (squared), respectively, and hence, why $\nu_0 \sigma_0^2$, may be seen as a **prior sum-of-squares** (by analogy to a degrees of freedom times a variance giving a sum-of-squares in many typical contexts).
- Distribution Parameterizations.** We will attempt to follow the distribution parameterizations of [GCS⁺14, Appendix A], unless otherwise indicated, though our symbols will differ.

- **Posterior By Conjugacy. Done.** Thus, given our conjugate prior above, we should expect the posterior (see below) to be the product of a **conditional normal distribution** and a **scaled inverse-chi-square** (with different hyperparameters that are somehow updated to incorporate our observed data, of course).
- **Theoretical Justification.** (Conditional) Conjugacy is not only convenient—giving close forms of (updated) familiar distributions—but also has theoretical justification. For certain families of data distributions, including ours, here, mixtures of conjugate distributions are also conjugate, and we can get arbitrarily close to many distributions with such mixtures ([Rob01, Lemma 3.4.2 & Theorem 3.4.3]). (Granted, we typically do not specify mixtures of conjugate priors in practice.)

- **Cheap But Common E.g.: Binomial-beta Conjugate Pair** (in class). This e.g. has little to do with our linear models, but it's a common illustration of conjugacy besides the normal-normal conditional conjugacy that we just saw, above (§C.2.2), or the (joint) conjugacy of the current section in what follows. (again, in class)

C.3.1 Posterior

- According to **Bayes theorem**, we have

$$[\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] \propto [\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} | \sigma^2][\sigma^2],$$

which, by conjugacy (and other standard results—omitted), can be shown to give the **posterior**

$$\begin{aligned} [\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] &= [\boldsymbol{\beta} | \sigma^2, \mathbf{y}][\sigma^2 | \mathbf{y}] \\ &= \mathcal{N}(\widehat{\mathbf{m}}, \sigma^2 \widehat{\Sigma}) \times \text{inv-}\chi^2(\widehat{\nu}, \widehat{\sigma}^2) \end{aligned}$$

where

$$\begin{aligned}
 \widehat{\boldsymbol{m}} &= (\Sigma_0^{-1} + (\mathbf{X}^t \mathbf{X}))^{-1} (\Sigma_0^{-1} \boldsymbol{m}_0 + (\mathbf{X}^t \mathbf{X}) \widehat{\boldsymbol{\beta}}) \quad \text{cond. post. mean,} \\
 \sigma^2 \widehat{\Sigma} &= \sigma^2 (\Sigma_0^{-1} + (\mathbf{X}^t \mathbf{X}))^{-1} \quad \text{cond. post. variance,} \\
 \widehat{\nu} &= \nu_0 + n \quad \text{post. df} \\
 \widehat{\sigma}^2 &= (1/\widehat{\nu})(\nu_0 \sigma_0^2 + (\mathbf{y} - \mathbf{X} \widehat{\boldsymbol{\beta}})^t (\mathbf{y} - \mathbf{X} \widehat{\boldsymbol{\beta}}) + \\
 &\quad (\boldsymbol{m}_0 - \widehat{\boldsymbol{\beta}})^t (\Sigma_0 + (\mathbf{X}^t \mathbf{X})^{-1})^{-1} (\boldsymbol{m}_0 - \widehat{\boldsymbol{\beta}})) \quad \text{post. scale (squared)}
 \end{aligned}$$

- **Parameter Update Formulas.** As in our previous discussion of conditional conjugacy for $\boldsymbol{\beta} | \sigma^2$ (§C.2.2), (joint) conjugacy of the current section simply requires that we use simple parameter update formulas to get from our prior to our posterior; the result of Bayes theorem has largely been obtained for us with relatively little effort (just some algebra, which you will undoubtedly find all over the Internet).
- **Same as Before.** Notice that you can plug in the prior variance, $\sigma^2 \Sigma_0$, here, for the more general prior variance, Σ_0 , of the previous section C.2.2 on the conditional distribution of $\boldsymbol{\beta} | \sigma^2$, to get the conditional posterior mean and variance here (after a small bit of algebra).
- **Notice a few things (again):**
 - **Posterior Mean.** The conditional normal posterior mean of $\boldsymbol{\beta}$ (given σ^2) is a **weighted average** of the prior mean of $\boldsymbol{\beta}$ and a data-based (unbiased) estimate of $\boldsymbol{\beta}$. Notice the **shrinkage** toward the prior mean aka **borrowing of strength** of the posterior mean from the prior mean when the data are relatively weak (variable) (or the prior is relatively strong (precise)).
 - **Posterior Variance.** The conditional normal posterior variance of $\boldsymbol{\beta}$ (given σ^2) is the **inverse of the sum of its prior precision (inverse of variance) and the precision of a data-based estimate**. Notice that the variance gets small as either the data or prior are precise (large precision matrices).

- **Posterior df.** The marginal posterior degrees of freedom is a **sum of the prior degrees of freedom and sample size**. We see that, in as much as n is related to our usual notion of degrees of freedom (usually $n-p$ for what we've done), ν_0 is analogous to n and degrees of freedom, hence the name prior "degrees of freedom" for ν_0 , or, at least, that's my explanation for the name of this parameter.
- **Posterior Sum-of-Squares.** The marginal posterior sum-of-squares is a sum of the prior sum-of-squares, the usual error sum-of-squares (RSS (or SSE), numerator of our usual MSE) and a sum-of-squares that is a sort of discrepancy between the prior and data-based means of β . We see that $\nu_0\sigma_0^2$ is analogous to RSS (SSE), hence is a prior "sum-of-squares". And, of course, with our explanation for ν_0 as the prior degrees of freedom, we see σ_0^2 as a prior "scale (squared)" similar to how σ^2 is the square of σ which scales our errors (and responses).
- **As the (conditional) prior variance "decreases (increases),"** relative to the data-based version of variance, the discrepancy between prior (conditional) mean and data-based mean is up(down)-weighted, making the posterior scale (squared) larger (smaller) hence making the posterior variance (and mean) of σ^2 larger (smaller). Thus, as we are more (less) precise about β , if its prior mean differs from the data based version, then we risk inflating (deflating) the posterior for the error variance, σ^2 . I find this to be **unattractive**. We need know the mean and variance of a scaled inv- χ^2 for this comment to make sense ([GCS⁺14, Appendix A]); in terms of the prior:

$$E(\sigma^2 | \sigma_0, \nu_0) = \frac{\nu_0}{\nu_0 - 2} \sigma_0^2,$$

$$Var(\sigma^2 | \sigma_0, \nu_0) = \frac{2\nu_0}{(\nu_0 - 2)^2(\nu_0 - 4)} \sigma_0^4,$$

and similarly for the posterior mean and variance.

- **As prior df increases (decreases),** the posterior df increases (decreases), of course, and the posterior scale (squared) looks more

(less) like the prior scale (squared) hence the posterior mean (and variance) of σ^2 looks more (less) like the prior scale (squared). (See above (prior) mean and variance for σ^2 .)

- **A Priori and a Bit Late To Mention.** As the error variance is smaller (larger), then this says that we are more (less) precise about β , which may seem a bit **unattractive**. More generally, it is often relatively difficult to specify a joint distribution. After all, our prior is supposed to reflect our belief, a priori, about parameters, and this behavior may not be easy to believe.
- We'll see other priors that avoid these unattractive features.

C.3.2 Marginal Posterior for β is a t

- **Marginal Posterior.** The marginal posterior, $[\beta | \mathbf{y}]$, is what we would use to infer about β (i.e., we want to use a distribution of $[unknowns|knowns]$). In some (rough) sense, it's the Bayesian counterpart to the normal/ $\chi^2/t/F$ results in note chapter 3 for inferring about β . We will use the following result about a t distribution to get $[\beta | \mathbf{y}]$ (and to get the posterior predictive, $[\mathbf{y}^* | \mathbf{y}]$, shortly thereafter). (Thus, we avoid having to integrate over σ^2 (and β in the predictive case).)
- **t Distribution as Mixture.** The (generic, not necessarily sample size) n dimensional t distribution (pdf),

$$t_n(\nu, \mu, \sigma_0^2 \Sigma),$$

is often defined as a **scale mixture** of a conditional normal with an inv- χ^2 (scaled inverse chi-square) mixing distribution (alternatively and equivalently, an inv-gamma mixing distribution, but I prefer the parameterization of an inv- χ^2). (n is the dimension, ν is called degrees of freedom, μ is the mean ($\nu > 1$), and $\sigma_0^2 \Sigma$ is called a scale matrix (often with σ_0 absorbed into Σ)).

- In other words, the t pdf is often defined as the marginal distribution,

$$[\boldsymbol{\theta}] = t_n(\nu, \boldsymbol{\mu}, \sigma_0^2 \boldsymbol{\Sigma}),$$

where $(\boldsymbol{\theta}^t, \sigma^2)^t$ has joint distribution

$$[\boldsymbol{\theta}, \sigma^2] = [\boldsymbol{\theta} | \sigma^2][\sigma^2]$$

defined by the conditional normal

$$\boldsymbol{\theta} | \sigma^2 \sim N_n(\boldsymbol{\mu}, \sigma^2 \boldsymbol{\Sigma})$$

and marginal inv- χ^2

$$\sigma^2 \sim \text{inv-}\chi^2(\nu, \sigma_0^2).$$

- This means, any time we see this form for a joint distribution, we can immediately write down the marginal

$$[\boldsymbol{\theta}] = t_n(\nu, \boldsymbol{\mu}, \sigma_0^2 \boldsymbol{\Sigma}).$$

- Incidentally, this is the multivariate, shifted and scaled (*not* non-central) version of the univariate standard t (i.e., **Student's t**) of “STAT 101.” In other words, if you multiplied a standard (Student's) t (with $df = \nu$) by a scalar, σ_0 , then added μ , you would have a variable distributed as

$$t \sim t_1(\nu, \mu, \sigma_0^2),$$

where $E(t) = \mu$ ($\nu > 1$) and $\text{Var}(t) = \frac{\nu}{\nu-2} \sigma_0^2$ ($\nu > 2$).

- (NOTE: [GCS⁺¹⁴, Appendix A] absorbs σ_0^2 into $\boldsymbol{\Sigma}$, and [Wak13, Appendix D] uses different symbols in a different order.)
- **Punchline.** Thus, we can use this mixture result to get our marginal (t) posterior (right!?).

$$[\boldsymbol{\beta} | \mathbf{y}] = t_p(\hat{\nu}, \hat{\boldsymbol{m}}, \hat{\sigma}^2 \hat{\boldsymbol{\Sigma}}),$$

where all updated posterior parameters are as defined above.

C.3.3 Posterior Predictive is a t

- **Unobserved Data Model.** Our model for n^* unobserved, “future” responses/outputs, which we denote as \mathbf{Y}^* , to distinguish them from observed inputs, \mathbf{Y} , follow our same data model as they did when discussing frequentists methods though we may not have been this explicit (nothing new here),

$$\mathbf{Y}^* \sim N(\mathbf{X}^* \boldsymbol{\beta}, \sigma^2 \mathbf{I}),$$

independent of \mathbf{Y} .

- **Joint (Unobserved and Observed) Data Distribution.** Thus, we have a joint data (unobserved and observed) distribution,

$$[\mathbf{y}^*, \mathbf{y} | \boldsymbol{\beta}, \sigma^2] = [\mathbf{y}^* | \mathbf{y}, \boldsymbol{\beta}, \sigma^2][\mathbf{y} | \boldsymbol{\beta}, \sigma^2] \stackrel{\text{ind.}}{=} [\mathbf{y}^* | \boldsymbol{\beta}, \sigma^2][\mathbf{y} | \boldsymbol{\beta}, \sigma^2],$$

which, again, is the same as it's always been, though, again, we may not have been this explicit.

Bayes Theorem. We can use our joint data distribution, along with our conjugate prior, in a slightly revised use of Bayes Theorem:

$$\begin{aligned} [\mathbf{y}^*, \boldsymbol{\beta}, \sigma^2 | \mathbf{y}] &= \frac{[\mathbf{y}^* | \boldsymbol{\beta}, \sigma^2][\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} | \sigma^2][\sigma^2]}{[\mathbf{y}]} \\ &= [\mathbf{y}^* | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} | \sigma^2, \mathbf{y}][\sigma^2 | \mathbf{y}], \quad \text{why?}, \end{aligned}$$

which we could have just written straightaway via the multiplication (product) rule (Definition B.14) applied to the joint posterior, $[\mathbf{y}^*, \boldsymbol{\beta}, \sigma^2 | \mathbf{y}]$, right?

- **Conditional Posterior Predictive: Marginal of Joint Normal Is Normal.** Before applying the above t mixture result to get the posterior predictive (without σ^2), we “integrate out” $\boldsymbol{\beta}$ (not really), to get a (marginal) conditional normal (note $[\mathbf{y}^* | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta} | \sigma^2, \mathbf{y}]$ is jointly normal, and marginals of normals are normal),

$$[\mathbf{y}^* | \sigma^2, \mathbf{y}],$$

thus getting a marginal posterior that is the product of a conditional normal posterior and the same marginal posterior $\text{inv-}\chi^2$,

$$[\mathbf{y}^*, \sigma^2 | \mathbf{y}] = [\mathbf{y}^* | \sigma^2, \mathbf{y}][\sigma^2 | \mathbf{y}],$$

where, in particular, (skipping some details)

$$\begin{aligned} [\mathbf{y}^* | \sigma^2, \mathbf{y}] &= N(\hat{\boldsymbol{\mu}}^*, \sigma^2 \hat{\boldsymbol{\Sigma}}^*), \\ \hat{\boldsymbol{\mu}}^* &= \mathbf{X}^*(\boldsymbol{\Sigma}_0^{-1} + (\mathbf{X}^t \mathbf{X}))^{-1}(\boldsymbol{\Sigma}_0^{-1} \mathbf{m}_0 + (\mathbf{X}^t \mathbf{X}) \hat{\boldsymbol{\beta}}), \\ &= \mathbf{X}^* \hat{\mathbf{m}} \\ \hat{\boldsymbol{\Sigma}}^* &= (\mathbf{I} + \mathbf{X}^*(\boldsymbol{\Sigma}_0^{-1} + (\mathbf{X}^t \mathbf{X}))^{-1} \mathbf{X}^{*t}), \\ &= (\mathbf{I} + \mathbf{X}^* \hat{\boldsymbol{\Sigma}} \mathbf{X}^{*t}), \end{aligned}$$

and (again) $[\sigma^2 | \mathbf{y}]$ is the same $\text{inv-}\chi^2$ posterior as above with the same df and scale parameters as defined above.

- **Shrinkage, Borrowing of Strength.** Notice a property for the conditional normal posterior predictive, $[\mathbf{y}^* | \sigma^2, \mathbf{y}]$, similar to that mentioned above for the conditional normal posterior $[\boldsymbol{\beta} | \sigma^2, \mathbf{y}]$: shrinkage towards—aka borrowing of strength by the posterior mean prediction from—a sort of prior mean prediction, $\mathbf{X}^* \mathbf{m}_0$, when the data are relatively weak (variable) (or when the prior is relatively strong (precise)).
- **Posterior Predictive from t Mixture Result.** We can use the above t mixture result to write down the desired (unconditional, no σ^2) posterior predictive distribution as

$$[\mathbf{y}^* | \mathbf{y}] = t_{n^*}(\hat{\nu}, \hat{\boldsymbol{\mu}}^*, \hat{\sigma}^2 \hat{\boldsymbol{\Sigma}}^*),$$

where all posterior parameters have been defined above (thus skipping integration over σ^2).

C.3.4 Remarks

- **Again, This Conjugate Prior May Offend.** Alas, as mentioned, the above normal \times inv- χ^2 conjugate prior for linear models may seem **unrealistic** in the sense that β becomes increasingly concentrated (disperse) around its prior mean, m_0 , as σ^2 becomes smaller (larger). Or, this just may seem like an unnatural way to express your prior belief about β .
- **Side Note: Connection to Other, Specific Priors.** As mentioned, Zellner's g -prior fits into the current discussion (and that of the next section) by taking $\sigma^2 \Sigma_0 = \sigma^2 g(\mathbf{X}^t \mathbf{X})^{-1}$ and specifying a value (or prior) for $g > 0$ (but usually setting $[\sigma^2] \propto \sigma^{-2}$, which is different than we do here), which seems to alleviate the previous item's concern (and we seem to have a more informative prior on β in the sense of using its (known covariate and hopefully familiar) data-based covariance structure, $(\mathbf{X}^t \mathbf{X})^{-1}$). Zellner's g -prior is traditionally used in Bayesian variable selection, which we may (not) cover in INF 504. And, as mentioned, a typical shrinkage prior specifies $\sigma^2 \Sigma_0 = \frac{\sigma^2}{\lambda} \mathbf{I}$ (with value/prior for shrinkage parameter λ ...). In Bayesian ridge regression, we often see an independence prior, $\beta \sim N(\mathbf{0}, \sigma_\beta^2 \mathbf{I})$ (β is (a priori) independent of σ^2 (σ^2 is not σ_β^2)); in this case the shrinkage parameter is $\lambda = \sigma^2 / \sigma_\beta^2$.
- **Other Priors.** Or, we may consider other priors. But, generally speaking, we do not get (full) conjugacy, but, instead, conditional conjugacy, e.g., $\beta | \sigma^2$ is normal a priori and a posteriori, and σ^2 is a priori inverse gamma as is $\sigma^2 | \beta$, a posteriori, but σ^2 is not unconditionally inverse gamma a posteriori. Or, perhaps we get no conjugacy (unusual for typical normal linear models). See below.
- **Upcoming Prior.** In the next section, we consider a popular **non-informative, improper prior**, that results in effectively the same form of posterior and posterior predictive (and our work here is not for naught!).

C.4 A Common Improper Prior

This section and the next two sections illustrate a curious (numerical) connection between our previous, frequentist linear model inference methods and Bayesian linear model inference when using a particular prior.

- A common **improper prior** distribution for the normal linear model is ([Wak13, Expr. (5.42)])
$$[\beta, \sigma^2] \propto \sigma^{-2}.$$
- **Jeffreys' Priors.** Incidentally, this is the product of **Jeffreys' prior** for β ($\propto 1$) and **Jeffreys' prior** for σ^2 (but, perhaps confusingly, is *not* Jeffreys' prior for (jointly) (β, σ^2) !...)
- **Improper Prior.** An improper distribution is one that does not integrate (or sum) to a finite value, thus it cannot be normalized to integrate (or sum) to 1.
- **Posterior Propriety.** It's the propriety of the posterior that counts: if you specify an improper prior, technically, you must check for posterior propriety (Calculus II convergence of integrals/series); we don't use any improper priors that lead to improper posteriors here.
- **Improper Limit of Previous Proper Prior.** Loosely, this improper prior can be viewed as the previous section's proper conjugate prior with df $\nu_0 = 0$ and prior precision (almost) $\Sigma_0^{-1} = 0$.
- **Not Conjugate.** It's not a conjugate prior (unless we consider this improper prior as a limiting case of the normal-inv- χ^2 conjugate family...).

C.4.1 Posterior

- **Posterior.** In short, we have the posterior

$$\begin{aligned} [\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] &= [\boldsymbol{\beta} | \sigma^2, \mathbf{y}][\sigma^2 | \mathbf{y}] \\ &= N(\widehat{\mathbf{m}}, \sigma^2 \widehat{\Sigma}) \times \text{inv-}\chi^2(\widehat{\nu}, \widehat{\sigma}^2) \end{aligned}$$

where

$$\begin{aligned} \widehat{\mathbf{m}} &= \widehat{\boldsymbol{\beta}} \quad \text{cond. post. mean,} \\ \sigma^2 \widehat{\Sigma} &= \sigma^2 (\mathbf{X}^t \mathbf{X})^{-1} \quad \text{cond. post. variance,} \\ \widehat{\nu} &= n - p \quad \text{degrees of freedom} \\ \widehat{\sigma}^2 &= (1/\widehat{\nu})(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}})^t(\mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}) \quad \text{post. scale squared} \end{aligned}$$

(e.g., [Wak13, top of pg. 223 & Expr. (5.45)] (typo $n - p - 1$ should be $n - p$ or $n - k - 1$ in Expr. (5.45)))

- **Mind Your df.** NOTE: now, we must have $n > p$ (posterior df $(n-p) > 0$) and \mathbf{X} must be full rank, neither of which were strictly necessary in the conjugate case, above, as long as Σ_0 was a valid variance matrix (positive definite).
- **Deja Vu.** Things are looking strangely familiar...

C.4.2 Marginal Posterior for $\boldsymbol{\beta}$ is a Familiar t

- **Apply t Mixture Result.** Again, the posterior is of the form of the above t scale mixture result, giving **marginal posterior**

$$[\boldsymbol{\beta} | \mathbf{y}] = t_p(n - p, \widehat{\boldsymbol{\beta}}, \widehat{\sigma}^2 (\mathbf{X}^t \mathbf{X})^{-1})$$

([Wak13, Expr. (5.44)], $p = k + 1$, and his parameter arguments are in a different order).

- **Deja Vu??** In particular, the MLE (and LS estimator) is the same as the (marginal) posterior mean, $\hat{\beta}$ (when $(n - p) > 1$), and the posterior variance (when $(n - p) > 2$) is the same as $\widehat{Var}(\hat{\beta})$!
- **Deja Vu?** In other words, each β_j is

$$t_1(n - p, \hat{\beta}_j, \hat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}^{-1})$$

where $(\mathbf{X}^t \mathbf{X})_{(jj)}^{-1}$ is the j th diagonal element of $(\mathbf{X}^t \mathbf{X})^{-1}$.

- **Deja Vu!** In other words, standardizing, we get

$$\frac{\beta_j - \hat{\beta}_j}{\sqrt{\hat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{jj}^{-1}}} \sim t(n - p) \quad whaoahuh!?$$

- **Result B.11.** This should look very familiar, but is, in some sense, very different as, now, in the Bayesian context, the randomness comes from β and not from $\hat{\beta}$ and $\hat{\sigma}^2$!
- **What's Random?** To be sure, β (β_j) is now random, not $\hat{\beta}$ ($\hat{\beta}_j$), which is now fixed, as is $\hat{\sigma}^2$.
- **Multivariate t .** In Appendix B, we could have stated a multi-variate t result for $\hat{\beta}$. Alas, we didn't.

C.4.3 Posterior Predictive is a Familiar t

- **Deja Vu??** In short, we have the **posterior predictive**

$$[\mathbf{y}^* | \mathbf{y}] = t_{n^*}(\hat{\nu}, \hat{\mu}^*, \hat{\sigma}^2 \hat{\Sigma}^*),$$

where (we've skipped a few details)

$$\begin{aligned}\hat{\boldsymbol{\mu}}^* &= \mathbf{X}^* \hat{\boldsymbol{\beta}}, \\ \hat{\Sigma}^* &= (\mathbf{I} + \mathbf{X}^* (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^{*t}), \\ \hat{\nu} &= n - p, \\ \hat{\sigma}^2 &= (1/\hat{\nu})(\mathbf{y} - \mathbf{X} \hat{\boldsymbol{\beta}})^t (\mathbf{y} - \mathbf{X} \hat{\boldsymbol{\beta}})\end{aligned}$$

- **Deja Vu?** In other words,

$$[\mathbf{y}^* | \mathbf{y}] = t_{n*}(n - p, \mathbf{X}^* \hat{\boldsymbol{\beta}}, \hat{\sigma}^2 (\mathbf{I} + \mathbf{X}^* (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{X}^{*t})),$$

which means y_i^* is

$$t_1(n - p, \mathbf{x}_i^{*t} \hat{\boldsymbol{\beta}}, \hat{\sigma}^2 (1 + \mathbf{x}_i^{*t} (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{x}_i^*)).$$

- **Deja Vu!** In other words, if we standardize, we get

$$\frac{Y_i^* - \mathbf{x}_i^{*t} \hat{\boldsymbol{\beta}}}{\sqrt{\hat{\sigma}^2 (1 + \mathbf{x}_i^{*t} (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{x}_i^*)}} \sim t(n - p)$$

(see §4.1).

- This, too, should look very familiar (§4.1), but, while y_i^* is random, Bayesian or not, $\hat{\boldsymbol{\beta}}$ and $\hat{\sigma}^2$ are now, again, fixed. (In note chapter 4, we did not use * notation.)

C.5 STAT 101 Redux a la Bayes

- **Examples, Revisited.** In light of the previous section's **particular improper prior** for our normal linear model, we revisit results for our familiar Galapagos and Body Fat examples. More discussion in class.
- **Streamlined Notation.** In this section, to streamline notation in the

hope of better drawing the coincidence with frequentist results, we will often suppress conditioning for the Bayesian results, e.g., we use β_j instead of $\beta_j | \mathbf{y}$ or use $E(Y | \mathbf{x}) = \mathbf{x}^t \boldsymbol{\beta}$ instead of $E(Y | \mathbf{y}, \mathbf{x}) = \mathbf{x}^t \boldsymbol{\beta} | \mathbf{y}$ or use $Y | \mathbf{x}$ instead of $Y | \mathbf{y}, \mathbf{x}$.

C.5.1 t -based Intervals for β_j

- **Recall Bayesians.** Reiterating our Bayesian results, we have

$$\beta_j \sim t_1(n - p, \widehat{\beta}_j, \widehat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}^{-1}).$$

- **Recall Frequentists.** Reiterating one of our frequentist sampling distribution results (Appendix B), we have

$$\widehat{\beta}_j \sim t_1(n - p, \beta_j, \widehat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}^{-1}).$$

- **Either Case.** In either case, we can standardize to get

$$Pr \left(t(n - p, \alpha/2) \leq \frac{\beta_j - \widehat{\beta}_j}{\sqrt{\widehat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}}} \leq t(n - p, 1 - \alpha/2) \right) = 1 - \alpha$$

(draw a picture) (dimension 1, dropped from notation).

- **Algebra.** After a bit of algebra and using the symmetry of the t distribution we get (see §3.5)

$$Pr \left(\widehat{\beta}_j - t(n - p, 1 - \alpha/2) \sqrt{\widehat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}} \leq \beta_j \leq \widehat{\beta}_j + t(n - p, 1 - \alpha/2) \sqrt{\widehat{\sigma}^2(\mathbf{X}^t \mathbf{X})_{(jj)}} \right) = 1 - \alpha$$

- **Quantitatively Same.** In other words, for the particular improper prior under current consideration in this current section, the Bayesian and frequentist intervals (§3.5) are the same, *quantitatively*.

- **Different Randomness.** However, the probability ("Pr") is computed differently in these two cases, right?

- **Frequentists.** For frequentists, the randomness comes from the data via the hatted quantities, $\hat{\beta}$ and $\hat{\sigma}^2$, and β_j is fixed (and unknown); after the observed data values are plugged in, the interval is fixed, and there is no randomness left, hence frequentists do not use “probability” to refer to their intervals, but, as we know, use the term **confidence intervals** with a **long-run relative frequency interpretation** using the notion of **hypothetical replications**.
- **Bayesians.** For Bayesians, the data and the hatted quantities are fixed, and β_j is random. Thus, the Bayesian interval is a fixed interval—numerically the same as the frequentist intervals for the prior under current consideration—within which a random quantity resides with some **probability**; still, we call these intervals **credible intervals**, and there is no need to resort to long-run relative frequency or hypothetical replications to interpret the intervals. Of course, this assumes that we somehow accept probability as serving as its own interpretation, perhaps as a degree of belief or degree of uncertainty about β_j .
- **Numerically Same, But Different Interpretation.** In short, some of the numerical results from our frequentist inference correspond exactly to the numerical results of a Bayesian inference under our particular improper prior, but interpretation is very different.
- **One-Sided Intervals.** The numerical correspondence also holds for one-sided intervals.

We may now view these frequentists “confidence intervals” (in the code/output below) from our new found Bayesian perspective (with a particular prior) as “credible intervals”, which we can now associate with the probability of containing a parameter. (Recall frequentists avoid use of the term “probability” and instead use “confidence”.)

```
> data(gala, package="faraway")
> lmod <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, gala)
> ## Frequentists: ``confidence'' intervals (contains with confidence):
> ## Bayesians: ``credible'' intervals (contains with probability) :
> confint(lmod)

              2.5 %    97.5 %
(Intercept) -32.464101 46.600542
Area         -0.070216  0.022339
Elevation     0.208710  0.430219
Nearest       -2.166486  2.184774
Scruz         -0.685093  0.204044
Adjacent      -0.111336 -0.038273
```

C.5.2 t -based Test for β_j

- **Kumbaya for Intervals and One-Sided Tests.** As just discussed, the numerical correspondence of frequentist and Bayesian intervals (for the particular improper prior under consideration) holds for **two-sided intervals** and **one-sided intervals**. A numerical correspondence holds for **one-sided tests**, too.
- **Paradoxically.** However, this Bayes/Freq correspondence does **not generally hold for two-sided tests with a point null hypothesis**; we often get very different numerical results when comparing frequentist p-values to Bayesian probabilities of null hypotheses consisting of a single value of the parameter as in our typical two-sided hypothesis tests. You can begin to see the reason for differences in such point null cases if you consider that, for a continuous parameter, θ , $Pr(\theta = \theta_0) = 0$, but we do not pursue these testing differences further (see the Jeffreys-Lindley Paradox in [Wak13, §4.4]).

- **One-Sided Test.** For one-sided tests, consider

$$\begin{aligned} H_0 : \beta_j &\leq b_0 \quad \text{and} \\ H_a : \beta_j &> b_0. \end{aligned}$$

- **Frequentists.** For frequentists, we compute $t_{stat} = (\hat{\beta}_j - b_0)/\widehat{se}(\hat{\beta}_j)$ and p-value = $Pr(t > t_{stat})$.
- **Bayesians** For Bayesians, we compute the **probability of the null hypothesis**,

$$\begin{aligned} Pr(H_0) &= Pr(\beta_j \leq b_0) \\ &= Pr\left(\frac{\beta_j - \hat{\beta}_j}{\widehat{se}(\hat{\beta}_j)} \leq \frac{b_0 - \hat{\beta}_j}{\widehat{se}(\hat{\beta}_j)}\right) \\ &= Pr(t \leq -t_{stat}) \\ &= Pr(t \geq t_{stat}) = \text{p-value} \end{aligned}$$

where the last equality follows by symmetry of the Student's (standard) t distribution.

- **Kumbaya.** That is, for the particular improper prior under current consideration, the usual one-sided tests for β_j give the same numerical result—p-value or probability of the null—but the interpretation is different.
- **Frequentist Interpretation.** For frequentists, the p-value has a **long-run relative frequency interpretation** using the notion of **hypothetical replications**.
- **Bayesian Interpretation.** For Bayesians, we can speak of the probability of the null being true.
- Despite the correspondence between the p-value and the probability of the null hypothesis being true, in this one-sided case, more generally, the **p-value is not the probability of the null hypothesis**.

```

> ## one-sided tests
> bhat<- coef(lmod)
> se<- sqrt(diag(vcov(lmod)))
>
> ## Ho: bj <= b0, Ha: bj > b0
> b0<- 0 ## for illustration only
> tstat<- (bhat - b0)/se
>
> ## Freq p-value = Pr(t > tstat):
> pt(tstat, df=lmod$df, lower.tail=FALSE)

(Intercept)      Area      Elevation      Nearest       Scruz
0.3576754001  0.8518410071  0.0000019117  0.4965753234  0.8623958887
   Adjacent
0.9998514673

> ## Bayes: Pr(bj <= b0) = Pr((bj - bhat)/se <= (b0-bhat)/se) = Pr(t <=
> ## -tstat) = Pr(t >= tstat) = p-value
> ## (same; no recomputation needed!)

```

C.5.3 t -based Intervals for $\mathbf{E}(Y | \mathbf{x}) = \mathbf{x}^t \boldsymbol{\beta}$

- **Bayesian.** We did not say it, above, but our Bayesian results, under the currently discussed improper prior, lead to

$$\mathbf{x}^t \boldsymbol{\beta} | \mathbf{y} \sim t_1(n - p, \mathbf{x}^t \widehat{\boldsymbol{\beta}}, \widehat{\sigma}^2 \mathbf{x}^t (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{x}),$$

which follows from (unstated) properties of the t distribution and the fact that $\boldsymbol{\beta} | \mathbf{y}$ is a t as shown above.

- **Frequentist.** The (unstandardized) frequentist version is (see notes near Result 3.2 for standardized (Student's t) version)

$$\mathbf{x}^t \widehat{\boldsymbol{\beta}} \sim t_1(n - p, \mathbf{x}^t \boldsymbol{\beta}, \widehat{\sigma}^2 \mathbf{x}^t (\mathbf{X}^t \mathbf{X})^{-1} \mathbf{x}).$$

- Thus, in the same way as above, we may now view the frequentist R output for our linear model from our (particular improper prior) Bayesian perspective.

```

> data(fat, package="faraway")
> lmod <- lm(brozek ~ age + weight + height + neck + chest +
+               abdom + hip + thigh + knee + ankle +
+               biceps + forearm + wrist, data=fat)
> x <- model.matrix(lmod)
> (x0 <- apply(x, 2, median))

(Intercept)      age     weight    height    neck
      1.00     43.00    176.50    70.00   38.00
      chest     abdom      hip     thigh     knee
      99.65     90.95    99.30    59.00   38.50
      ankle     biceps    forearm    wrist
      22.80     32.05    28.70    18.30

> ## Interval estimator of the mean  $E(y | x_0 = \text{median})$ 
> (est<- predict(lmod, new=data.frame(t(x0)),
+                  interval="confidence", se.fit=TRUE))

$fit
    fit     lwr     upr
1 17.493 16.944 18.042

$se.fit
[1] 0.27867

$df
[1] 238

$residual.scale
[1] 3.988

```

C.5.4 t -based Prediction Intervals for $Y | x$

In essentially the same way as above, omitting details, we get similar t results and may now view the frequentist R output for prediction intervals from our Bayesian perspective. (Y here denotes an unobserved response despite our not using $*$ notation as we did above.)

```
> ## Prediction of Y / x0 = median
> (pred<- predict(lmod,new=data.frame(t(x0)),
+                   interval="prediction", se.fit=TRUE))

$fit
    fit     lwr     upr
1 17.493 9.6178 25.369

$se.fit
[1] 0.27867

$df
[1] 238

$residual.scale
[1] 3.988
```

C.6 Prostate Data Example with Improper Prior

- **No Animals Were Harmed.** At the risk of beating dead horses, we continue to illustrate the ongoing improper prior results (§C.4) using an example taken from [Wak13, §5.12] based on our familiar prostate data (same data as used in [Far14]). (In subsequent sections, we'll continue analysis of these data under a different prior and using different methods to get at the posterior, in which case the Bayesian-frequentist numerical correspondence does not hold.)
- **Standardize Covariates.** To put the effects (the β_j , right!?) on the same scale for plot comparisons, we standardized covariates to the interval, $[0,1]$. (We will also use this same standardized data in a subsequent section of our notes to elicit an informative prior distribution following [Wak13, §5.12], shortly.)
- **Side Note on Standardization.** Often, covariates are standardized in some way in regularization/shrinkage methods so that a single shrinkage

parameter more sensibly applies to multiple effects (weights/parameters) associated with covariates (or hidden units). More in INF 504 and other courses.

```
> ## We standardize x's to get betas on same scale for
> ## plotting, shortly.
> data(prostate, package="faraway")
> standx<- function(x) {
+   rangex<- range(x)
+   (x - rangex[1]) / diff(rangex)
+ }
> zprostate<- prostate$lpsa
> for (k in 1:8) zprostate<-
+   cbind.data.frame(zprostate,
+                     standx(prostate[,k]))
> names(zprostate)<- c("lpsa",paste0("z.",names(prostate)[1:8]))
> round(head(zprostate, n=3), 3)

  lpsa z.lcavol z.lweight z.age z.lbph z.svi z.lcp z.gleason
1 -0.431    0.148    0.106 0.237      0      0      0     0.000
2 -0.163    0.068    0.253 0.447      0      0      0     0.000
3 -0.163    0.162    0.085 0.868      0      0      0     0.333
z.pgg45
1    0.0
2    0.0
3    0.2

> round(tail(zprostate, n=3), 3)

  lpsa z.lcavol z.lweight z.age z.lbph z.svi z.lcp z.gleason
95 5.143    0.823    0.274 0.289  0.000      1 0.897    0.333
96 5.478    0.818    0.375 0.711  0.793      1 0.686    0.333
97 5.583    0.932    0.429 0.711  0.491      1 1.000    0.333
z.pgg45
95    0.1
96    0.8
97    0.2
```

C.6.1 Frequentist R Summary

Been there, done that.

```
> summary(zprostate.lm<- lm(lpsa ~ ., data=zprostate))

Call:
lm(formula = lpsa ~ ., data = zprostate)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.733 -0.371 -0.017  0.414  1.638 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.421     0.296    1.42   0.1578    
z.lcavol     3.034     0.454    6.68  2.1e-09 ***  
z.lweight    1.696     0.635    2.67   0.0090 **   
z.age        -0.746     0.425   -1.76   0.0823    
z.lbph       0.397     0.217    1.83   0.0704    
z.svi        0.766     0.244    3.14   0.0023 **   
z.lcp        -0.453     0.390   -1.16   0.2496    
z.gleason    0.135     0.472    0.29   0.7750    
z.pgg45      0.453     0.442    1.02   0.3089    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.708 on 88 degrees of freedom
Multiple R-squared:  0.655, Adjusted R-squared:  0.623 
F-statistic: 20.9 on 8 and 88 DF,  p-value: <2e-16
```

C.6.2 Bayesian Summary

- **t Posteriors.** Here we use the previously discussed marginal t posteriors of the β_j for comparison with the corresponding frequentist results just

given, reproducing [Wak13, Fig. 5.10] (up to errata), which displays the marginal t posteriors of the regression function parameters (excluding β_0) for the particular improper prior that we continue to consider.

- **Unusual to the Uninitiated.** This section may seem a bit peculiar to practitioners who are accustomed to seeing samples from a posterior and sample summaries that approximate posterior quantities. Though the computation of quantities here in this section is exact, we do not typically get such exact results for Bayesian models, and most practitioners are accustomed to samples from the posterior and sample approximations to the exact results, even if exact results are available, as they are here.

```
> ## Bayes posterior t means (df>1) and LS/MLE:
> tpostmeans<- coef(zprostate.lm)
> ## Bayes posterior t df (n-p) and freq error df:
> tpostdf<- zprostate.lm$df
> ## Bayes posterior t squared scales:
> tpostscales2<- diag(vcov(zprostate.lm))
> ## Bayes posterior t variances (df>2):
> tpostvars<- tpostdf / (tpostdf -2) * tpostscales2
> ## Bayes posterior t standard deviations:
> tpostsds<- sqrt(tpostvars)
> ## Freq ses are Bayes t scales, not quite same as Bayes t sds:
> tfreqses<- sqrt(tpostscales2)
> ## Intervals:
> t975<- qt(p=1-0.05/2, df=tpostdf)
> tpost25lb<- tpostmeans - t975*sqrt(tpostscales2)
> tpost975ub<- tpostmeans + t975*sqrt(tpostscales2)
> ## Bayes/freq summary (See Figs. 5.10 and 5.11 in Wakefield's BFRM)
> round(cbind("pmean"=tpostmeans, "psd"=tpostsds,
+           "freqse"=tfreqses, "25lb"=tpost25lb,
+           "975ub"=tpost975ub), 4)

          pmean      psd freqse     25lb   975ub
(Intercept) 0.4214 0.2992 0.2958 -0.1664 1.0092
z.lcavol    3.0338 0.4596 0.4544  2.1308 3.9368
z.lweight   1.6964 0.6419 0.6346  0.4352 2.9575
z.age       -0.7462 0.4295 0.4246 -1.5899 0.0975
```

```

z.lbph      0.3974 0.2195 0.2170 -0.0338 0.8287
z.svi       0.7662 0.2471 0.2443  0.2806 1.2517
z.lcp       -0.4525 0.3950 0.3905 -1.2285 0.3235
z.gleason   0.1354 0.4779 0.4724 -0.8034 1.0742
z.pgg45     0.4525 0.4472 0.4421 -0.4261 1.3311

> ## Again, Bayes CI's are same (numerically) save as freq CIs:
> confint(zprostate.lm)

              2.5 % 97.5 %
(Intercept) -0.16638 1.009229
z.lcavol     2.13079 3.936757
z.lweight    0.43525 2.957535
z.age        -1.58994 0.097518
z.lbph       -0.03379 0.828687
z.svi        0.28064 1.251670
z.lcp        -1.22855 0.323483
z.gleason   -0.80336 1.074208
z.pgg45     -0.42609 1.331139

```

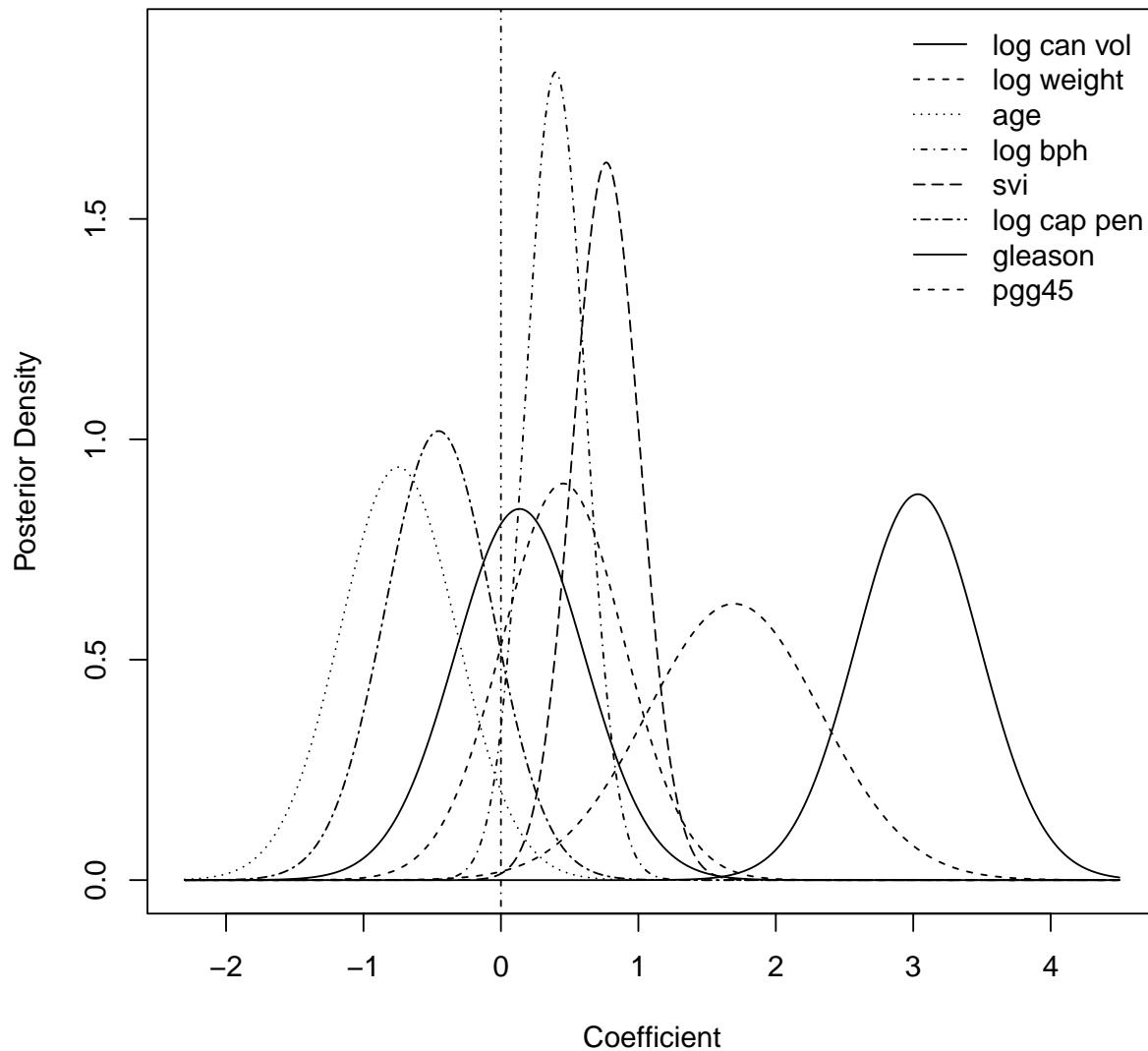
The following figure reproduces the marginal posterior distributions for the β_j in [Wak13, Fig. 5.10] (up to typos in his code and manifested in his figure, at least in my printing of his textbook).

```

> ## (Wakefield BFRM Fig 5.10 (up to errors):
> ## marginal betaj posterior distributions under improper prior)
> par(mfrow=c(1,1),mar=c(5, 4, 4, 2)+0.1)
> ## Grid of beta values to plot marg. post. t densities
> bgrid<- seq(-2.3,4.5,.01)
> ## Omit intercept marginal (param=1) as in Fig. 5.10:
> invisible(sapply(2:9, FUN=function(param,bgrid,pmean,pscale, pdf){
+   ## Using location-scale properties of t distribution to plot:
+   tmargdens<- dt((bgrid - pmean[param])/pscale[param], df=pdf)/
+     pscale[param]
+   if(param==2)
+     plot(bgrid,tmargdens,type="l",xlab="Coefficient",
+           ylab="Posterior Density",
+           ylim=c(0,1.9), lty=param-1)
+   else

```

```
+      lines(bgrid,tmargdens,lty=param-1)
+ },
+ bgrid=bgrid,
+ pmean=tpostmeans,
+ pscale=sqrt(tpostscales2),
+ pdf=tpostdf
+ ))
> abline(v=0,lty=4) ## not sure why 4
> legend("topright",legend=c("log can vol","log weight","age","log bph",
+ "svi","log cap pen","gleason","pgg45"),
+ bty="n",lty=1:8)
```



Side Note on Above Plotting of t Distributions. To understand how the plots are created in the code, we should know that t distributions are what is called a **location-scale family** of distributions. For univariate t distributions, like our posterior marginals for the β_j , we have

$$t \sim t_1(\mu, \sigma^2, \nu),$$

where μ is the **location** parameter, which, in the current context, corresponds to our posterior mean, i.e., $\mu = \hat{\beta}_j$, (when degrees of freedom $\nu = (n - p) > 1$). And, σ corresponds to the **scale** parameter, which, in our current context, corresponds to the posterior scale $\sigma = \sqrt{\widehat{se}(\hat{\beta}_j)}$, and note that $\text{Var}(t) = (\nu/(\nu - 2))\sigma^2$ (the scale is not the standard deviation as indicated briefly in previous code).

If we denote $f(t)$ to be the pdf of t , then

$$f(t) = \frac{1}{\sigma} h\left(\frac{t - \mu}{\sigma}\right)$$

where

$$h(t) = t(0, 1, \nu)$$

is the pdf of Student's t , i.e., the standard t (analogous to $N(0, 1)$). In other words, just as we can use a standard normal (table) to compute probabilities/quantiles (and densities) of general—shifted (located) and scaled—normal random variables, we can use a standard (Student's) t for more general t random variables from their location-scale family.

C.7 A Common Independence Prior

$$\begin{aligned} [\boldsymbol{\beta}, \sigma^2] &= [\boldsymbol{\beta}][\sigma^2] \\ &= \mathcal{N}(\mathbf{m}_0, \Sigma_0) \times \text{inv-}\chi^2(\nu_0, \sigma_0^2) \end{aligned}$$

- **Independence.** Notice, in particular, that we now assume independence a priori. (Again, an inverse gamma is equivalent to an inv- χ^2 , and, incidentally, saying $\sigma^{-2} \sim \text{gamma}(\alpha, \beta)$ is the same as saying $\sigma^2 \sim \text{inv-gamma}(\alpha, \beta)$.)
- **No Unconditional Conjugacy. Form of Posterior Not Familiar.**
This independence prior is not a conjugate prior (unless we consider

$\Sigma_0^{-1} = \mathbf{0}$ and $\nu_0 = 0$, which gives the previous improper prior and closed-form $N \times \text{inv-}\chi^2$ posterior results), and we do not get a recognizable posterior distribution (for which we usually know means, variances, probabilities, R functions, etc., to help us summarize results).

- **Now What?** So, we are without a posterior at the moment, unlike our previous two cases discussed in §C.3 - C.6, wherein we had posteriors of known form.

C.7.1 Full Conditional Posterior Distributions

We do, however, know the form of **full conditional posterior distributions** (often shortened to “**full conditionals**”), the distribution of sub-vectors of unknown quantities conditional on, or given, ‘all’ of the others. (Sometimes, we have conditional independence, so that some ‘full’ conditionals may not actually depend on all of the other quantities, despite the name and the full conditional notation.)

$$\begin{aligned}\boldsymbol{\beta} | \sigma^2, \mathbf{y} &\sim N(\widehat{\mathbf{m}}, \sigma^2 \widehat{\Sigma}) \\ \sigma^2 | \boldsymbol{\beta}, \mathbf{y} &\sim \text{inv-}\chi^2(\widehat{\nu}, \widehat{\sigma}^2),\end{aligned}$$

where

$$\begin{aligned}\widehat{\mathbf{m}} &= (\Sigma_0^{-1} + \sigma^{-2} \mathbf{X}^t \mathbf{X})^{-1} (\Sigma_0^{-1} \mathbf{m}_0 + \sigma^{-2} (\mathbf{X}^t \mathbf{X}) \widehat{\boldsymbol{\beta}}), \\ \sigma^2 \widehat{\Sigma} &= (\Sigma_0^{-1} + \sigma^{-2} (\mathbf{X}^t \mathbf{X}))^{-1} \\ \widehat{\nu} &= \nu_0 + n \\ \widehat{\sigma}^2 &= (1/\widehat{\nu})(\nu_0 \sigma_0^2 + (\mathbf{y} - \mathbf{X} \boldsymbol{\beta})^t (\mathbf{y} - \mathbf{X} \boldsymbol{\beta})).\end{aligned}$$

Note, the (full) conditional posterior for $\boldsymbol{\beta} | \sigma^2$ is the same form (normal) as the often occurring conditional given previously in §C.2.2. And, note that we have conditional conjugacy (normal-normal and $\text{inv-}\chi^2$ - $\text{inv-}\chi^2$). But, while our prior is normal \times $\text{inv-}\chi^2$, our posterior is not, so we do not have

full conjugacy. As we will see, these known full conditionals, resulting from **conditional conjugacy** in this case, can be used in MCMC routines to sample from the posterior even though we do **not** have a known form for the full posterior in this case where **full conjugacy** does not follow.)

- **Common, More Specific Prior.** We've mentioned the commonly used prior with $\mathbf{m}_0 = \mathbf{0}$ and $\Sigma_0 = \sigma_\beta^2 \mathbf{I}$, i.e.,

$$[\boldsymbol{\beta}] = \mathcal{N}(\mathbf{0}, \sigma_\beta^2 \mathbf{I}),$$

with the same inv- χ^2 prior as just given above.

In this **particular case**, we have the **full conditional**,

$$[\boldsymbol{\beta} | \sigma^2, \mathbf{y}] = \mathcal{N}(\widehat{\mathbf{m}}, \sigma^2 \widehat{\Sigma}),$$

where, now,

$$\begin{aligned}\widehat{\mathbf{m}} &= (\sigma^{-2} \mathbf{I} + \sigma^{-2} \mathbf{X}^t \mathbf{X})^{-1} (\sigma^{-2} (\mathbf{X}^t \mathbf{X}) \widehat{\boldsymbol{\beta}}), \\ \sigma^2 \widehat{\Sigma} &= (\sigma_\beta^{-2} \mathbf{I} + \sigma^{-2} \mathbf{X}^t \mathbf{X})^{-1},\end{aligned}$$

and the other full conditional posterior, $[\sigma^2 | \boldsymbol{\beta}, \mathbf{y}]$, is the same as previously given.

- We have not answered our previous question, 'Now what?' That is, how does knowing the **full conditionals** get us closer to the posterior (and posterior predictive)? There are many answers to that question, the most popular answer being **Gibbs sampling** or, more generally, Markov chain Monte Carlo (MCMC) methods (McMC?...). (I think I may have mentioned this before.)

C.8 2-Stage Gibbs Sampling

- **Known Posterior.** When we know the posterior distribution, e.g., normal \times inv- χ^2 , as in the conjugate prior case (§C.3) and as in the improper prior case (§C.4), then we ‘merely’ have to summarize the distribution as we see fit. For example, in §C.6 we (i.e., Jon Wakefield [Wak13]) summarized the posterior using known properties (mean, variance, etc.) of the resulting known distribution. But, as mentioned, that sort of summary is not what most (uninitiated) practitioners are accustomed to, typically relying on probabilistic programming languages to produce samples of the posterior and corresponding sample approximations to means, variances, credible intervals, etc., by whatever methodology this is done. If the entire posterior is known, then we may obtain summaries of independent (not Markov chain) Monte Carlo (MC) samples using, e.g., one or more of R’s ‘r’ functions (e.g., `rnorm`, `rchisq`), but I believe most users would simply specify the numerator of Bayes theorem and be happy with the samples returned by their favorite probabilistic programing language, whether or not the full posterior is available in familiar form.
- **Known Full Conditionals.** Gibbs sampling is a popular and relatively easy type of Markov chain Monte Carlo (MCMC) method most frequently used in practice to sample from posterior distributions whose form we do not know but for which we know **full conditional distributions**. We state a **2-stage Gibbs sampling** algorithm in the context of our 2 full conditionals that we obtained using the indepence prior (§C.7).
- **We Skip Some Theory.** We do not cover the theory that tells us why sampling from the conditionals of the joint distribution will get us a sample from the joint $[\beta, \sigma^2 | \mathbf{y}]$. The seminal reference for Gibbs sampling is [GS90], but, by now, you will find countless other books/articles on Gibbs and related sampling algorithms. (Alan Gelfand—the ‘G’ in [GS90]—was my post-doc advisor, from whom I learned a bit about MCMC.)

For the case of the independence prior of §C.7, we have shown 2 full conditional distributions that we can use in Gibbs sampling to obtain samples from the full posterior of β and σ^2 . For an initially ('iteration' $t = 0$) chosen value of $\sigma^2 = \sigma^{2(0)}$,

1. sample $\beta^{(t+1)} | \sigma^{2(t)}, \mathbf{y} \sim [\beta | \sigma^{2(t)}] = N(\hat{\mathbf{m}}, \sigma^{2(t)} \hat{\Sigma})$
2. sample $\sigma^{2(t+1)} | \beta^{(t+1)}, \mathbf{y} \sim [\sigma^2 | \beta^{(t+1)}] = \text{inv-}\chi^2(\hat{\nu}, \hat{\sigma}^2)$
3. repeat 1 & 2 "to and beyond convergence" (to be discussed)

Note, we don't need an initial value, $\beta^{(0)}$, and, recall, β is inside of $\hat{\sigma}^2$. (Note, we could have started with an initial value of $\beta = \beta^{(0)}$ and switched steps 1 and 2.)

- **Convergence Diagnostics.** Of course, we should not expect to immediately start sampling from the posterior distribution. (Why?) We should perform convergence diagnostics to help us decide when it is appropriate to assume that the samples we're getting from our algorithm have somehow converged to samples from the posterior.
- **Summary & Inference.** We can use the subsequent converged samples—none before convergence—to summarize the posterior and infer about parameters/unknowns.
- **3-Stage Gibbs Sampling Example.** Next section.

C.9 3-Stage Gibbs Sampling: Prostate Data Example with “Combo” Prior

- **Prostate E.g.** We illustrate Gibbs sampling with the prostate data (with

standardized inputs for comparison to previous results and to help us elicit a prior).

- **Leads Naturally to Gibbs Sampling.** Our (Jon Wakefield's) choice of prior leads to **3-stage Gibbs sampling**, which you should see as a natural extension to the 2-stage sampling algorithm given above (once we determine 3 full conditionals, shortly).

C.9.1 Eliciting a Prior

- **First, a Prior Like None Before.** We need a specific form of prior in order to determine and code specific full-conditional distributions for Gibbs sampling (assuming we recognize the full conditionals to be of familiar form...we will). We follow [Wak13, pp. 246-7] to construct a prior, which, as mentioned, may be seen, loosely, as a sort of combination of the independence prior (§C.7) and improper prior (§C.4) that we have discussed.

- **Prior Details.** In particular, we specify our (overall improper) prior as

$$[\boldsymbol{\beta}, \sigma^2] = [\boldsymbol{\beta}][\sigma^2] \propto \left(\prod_{j=0}^k [\beta_j] \right) \sigma^{-2},$$

where

- $[\beta_0] \propto 1$ (improper for the intercept),
- $[\sigma^2] \propto \sigma^{-2}$ (improper for the error variance), but
- $[\beta_j] = N(0, V)$, $j > 0$, are proper, where we (Jon) elicits the prior standard deviation, \sqrt{V} , based on our (Jon's) belief that it is unlikely that any of the **standardized covariates**, over their range of $[0,1]$, will change the median PSA by more than 10 units—equivalently, unlikely to change the mean $\log(\text{PSA})$ by more than $\log(10)$ (the

correspondence of median PSA to mean $\log(\text{PSA})$ results from our assumption that our response, $\log(\text{PSA})$, is normal). In other words, as a (standardized) covariate ranges over $[0,1]$, we expect *a priori* that it is unlikely that $|\beta_j|$ will exceed a change of $\log(10)$. We may incorporate this into our normal prior, with the *a priori* expectation of zero effect (on $\log(\text{PSA})$ scale), by assuming that the maximum value of β_j , as the covariate ranges over $[0,1]$, is $\beta_j = \log(10)$, and by assuming this value occurs at $1.96\sqrt{V}$. Thus, we set $\log(10) = 0 + 1.96\sqrt{V}$, and solve for $V = (\log(10)/1.96)^2$. (Visualize the normal (prior) distribution centered at 0 with $\pm \log(10)$ being 1.96 standard deviations away from 0, thus creating a 95% *a priori* probability interval for the β_j .)

C.9.2 Full Conditionals

- **Want to Avoid FC Details?** [Wak13] uses the method of integrated nested Laplace approximations (INLA), implemented in the R package `rnla`, which does not require users to specify full conditionals. And, high level probabilistic programming languages (e.g., BUGS, WinBUGS, OpenBUGS, JAGS, Stan, NIMBLE), often “know” the standard forms of full conditionals—roughly speaking, those distributions for which we know how to easily generate random value, e.g., from familiar `r` (as in `pdqr` (see §A.12)) functions in R, e.g, `rnorm`, `rt`, `rchisq`, etc., and, again, users need not specify full-conditionals.
- **You Want the Details.** But, we’re informatics students, and we may have to program our own algorithms, which may require us to derive full conditionals.
- **First, Consider A Generic Sampling Problem.** Briefly, we want to sample from a distribution, a posterior in the Bayesian context, of course. For the generic normal linear model, as we know,

$$[\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] \propto [\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta}, \sigma^2].$$

- **Previous FCs.** For some priors, e.g., the conjugate prior, discussed in §C.3 and the common improper prior in §C.4, we know the entire joint posterior, $[\beta, \sigma^2 | \mathbf{y}]$, normal \times inv- χ^2 , which I gave to you, omitting the algebra used to derive the posterior in these cases. In the case of our common independence prior in §C.7, the posterior is not of standard form, but we know full conditional distributions, $[\beta | \sigma^2, \mathbf{y}]$ and $[\sigma^2 | \beta, \mathbf{y}]$, which, again, I gave to you, again omitting algebra to arrive at these results.
- **FCs for Our Current Model.** In our current example, with our somewhat (not terribly) unique prior, we cannot use previous results directly. We need someone or some thing (e.g., Stan, etc.) to help, or we have to program our own algorithm. Of course, we will do Gibbs sampling, which requires standard full conditionals that are easy to sample from. Our particular prior chosen above leads to known full-conditional posterior distributions—usually shortened to ‘full-conditionals’.

•

$$[\beta_0 | \beta, \sigma^2, \mathbf{y}] = N\left(\bar{y}^*, \frac{\sigma^2}{n}\right),$$

where $\bar{y}^* = \sum_{i=1}^n y_i^*/n$, the average of values, y_i^* , defined as $y_i^* = y_i - \mathbf{x}_i^{*t} \beta^*$ (a sort of residuals), where \mathbf{x}_i^* is the i th row of \mathbf{X} **without** its 1 in the first column, and, correspondingly, β^* is β **without** β_0 .

•

$$[\sigma^2 | \beta, \mathbf{y}] = \text{inv-}\chi^2\left(\nu_0 = n, \sigma_0^2 = \frac{SSE + (\beta - \hat{\beta})^t(\mathbf{X}^t\mathbf{X})(\beta - \hat{\beta})}{n}\right),$$

where $SSE = (\mathbf{y} - \mathbf{X}\hat{\beta})^t(\mathbf{y} - \mathbf{X}\hat{\beta})$ (We may have used RSS to denote this previously.)

•

$$\begin{aligned} [\beta^* | \beta_0, \sigma^2, \mathbf{y}] &= N\left((\sigma^{-2}(\mathbf{X}^{*t}\mathbf{X}^*) + V^{-1}\mathbf{I})^{-1}(\sigma^{-2}(\mathbf{X}^{*t}\mathbf{X}^*)\hat{\beta}^{*(t)} + V^{-1}\mathbf{m}_0^*), \right. \\ &\quad \left. (\sigma^{-2}(\mathbf{X}^{*t}\mathbf{X}^*) + V^{-1}\mathbf{I})^{-1}\right), \end{aligned}$$

where \mathbf{m}_0^* is the prior mean of $\boldsymbol{\beta}^*$ (no β_0), which we chose to be $\mathbf{0}$, $\hat{\boldsymbol{\beta}}^* = (\mathbf{X}^{*t}\mathbf{X}^*)^{-1}\mathbf{X}^{*t}y^{**}$, where y^{**} is defined as $y^{**} = y - \mathbf{1}\beta_0$ (not the same as the previously defined y^*), and all other quantities are as defined, above. You can see that it's essentially the same as previous conditional posteriors for effects, with the changes mentioned. Incidentally, the derivation details are very similar to (essentially the same as) manipulations used to derive (full conditional) posterior results shown in previous sections. (Sorry, I have not made available these derivations; mine are in fair shape, but I have to re-write them, nicely, to make them sharable, and doing this is low on my priority list. Besides, as I said, you'll likely find similar derivation memes on the Web.)

- **A Few More Details Towards FCs.** To give you some idea of how these derivations are done, we notice that our full conditional distributions are proportional to the (joint) posterior, just like any conditional distribution is proportional to a joint. For our example,

$$\begin{aligned} [\beta_0 | \boldsymbol{\beta}^*, \sigma^2, \mathbf{y}] &\propto [\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] \\ &\propto [\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta}, \sigma^2] \end{aligned}$$

$$[\sigma^2 | \boldsymbol{\beta}, \mathbf{y}] \propto [\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta}, \sigma^2]$$

$$[\boldsymbol{\beta}^* | \beta_0, \sigma^2, \mathbf{y}] \propto [\mathbf{y} | \boldsymbol{\beta}, \sigma^2][\boldsymbol{\beta}, \sigma^2].$$

- **Look for Factors.** We are looking for standard distributions for each of β_0 , $\boldsymbol{\beta}^*$ and σ^2 , respectively, and we may ignore any multiplicative factors (additive summands on the log scale) that do not depend on the random variable whose distribution we are looking for.

- When looking for $[\beta_0 | \boldsymbol{\beta}^*, \sigma^2, \mathbf{y}]$, we may ignore multiplicative factors involving $\boldsymbol{\beta}^*$, σ^2 or \mathbf{y} (or any other factors not involving β_0).

- When looking for $[\sigma^2 | \beta, \mathbf{y}]$, we may ignore factors involving β or \mathbf{y} (or any other factors not involving σ^2).
- When looking for $[\beta^* | \beta_0, \sigma^2, \mathbf{y}]$, we may ignore...?
- **Generally.** In general, when looking for the pdf of X , $f(x) = c \times g(x)$, we may look for $g(x)$ and ignore constants of proportionality, c , which do not depend on x . The catch is that we must recognize $f(x)$ from $g(x)$, which we have called the **kernel** of $f(x)$ (in the context of looking for the entire posterior in §C.1.2, above), i.e., we must recognize the kernel. (e.g., “Hey, I recognize that as part of a normal pdf for X , aside from a few missing multiplicative constants that don’t involve x .”)
- **Examples (Sort of).** For β_0 and β^* , you will see quadratic forms inside the \exp function, which suggest normal distributions, which should motivate a bit of algebra to find the parameter expressions (shown above) of these normals. For σ^2 , you should essentially immediately see the kernel of an $\text{inv-}\chi^2$ (or inverse gamma), without much algebra (not much beyond that needed for the re-expression of $[\mathbf{y} | \beta, \sigma^2]$ that I showed, anyway).
- **Enough.** For time, we omit further detail on the fascinating world of deriving full-conditionals and now return to our regularly scheduled informatics program.

C.9.3 Gibbs Sampling Algorithm

For the initially chosen values of $\sigma^2 = \sigma^{2(t=0)}$ and $\beta^* = \beta^{*(t=0)}$

1. sample

$$\begin{aligned} \beta^{*(t+1)} | \beta_0^{(t)}, \sigma^{2(t)}, \mathbf{y} &\sim [\beta^* | \beta_0^{(t)}, \sigma^{2(t)}, \mathbf{y}] \\ &= \mathcal{N} \left((\sigma^{-2(t)}(\mathbf{X}^{*t} \mathbf{X}^*) + V^{-1} \mathbf{I})^{-1} (\sigma^{-2(t)}(\mathbf{X}^{*t} \mathbf{X}^*) \hat{\beta}^* + V^{-1} \mathbf{m}_0^*), \right. \\ &\quad \left. (\sigma^{-2(t)}(\mathbf{X}^{*t} \mathbf{X}^*) + V^{-1} \mathbf{I})^{-1} \right), \end{aligned}$$

2. sample

$$\begin{aligned}\beta_0^{(t+1)} | \boldsymbol{\beta}^{*(t+1)}, \sigma^{2(t)}, \mathbf{y} &\sim [\beta_0 | \boldsymbol{\beta}^{*(t+1)}, \sigma^{2(t)}, \mathbf{y}] \\ &= \mathcal{N}\left(\bar{y}^{*(t+1)}, \frac{\sigma^{2(t)}}{n}\right)\end{aligned}$$

3. sample

$$\begin{aligned}\sigma^{2(t+1)} | \boldsymbol{\beta}^{(t+1)}, \mathbf{y} &\sim [\sigma^2 | \boldsymbol{\beta}^{(t+1)}] \\ &= \text{inv-}\chi^2\left(n, \frac{SSE + (\boldsymbol{\beta}^{(t+1)} - \hat{\boldsymbol{\beta}})^t (\mathbf{X}^t \mathbf{X}) (\boldsymbol{\beta}^{(t+1)} - \hat{\boldsymbol{\beta}})}{n}\right)\end{aligned}$$

4. repeat 1,2 & 3 “to convergence and beyond”

All quantities are as defined in §C.9.2, above.

C.9.4 Gibbs Sampling Code

- **To the Code!** For now, we go directly to my code, to be discussed in class. Note that I have attempted to choose variable names corresponding to those used in the presentation of the full-conditionals, above (perhaps some of the variables given only in omitted details of my derivation of the FCs).

COMPUTATIONAL DISCLAIMER: I do not claim that the following code is somehow computationally efficient or numerically stable. But, it contains the sort of computations that, hopefully, illustrate conceptually, at least, our typeset material.

PROPERTY WARNING: Improper priors may lead to improper posteriors, even if the full conditionals are proper! With that said, we don't have to worry about posterior propriety with our particular improper prior used here. In general, however, you need to verify posterior propriety,

which is an exercise in convergence of series or integrals (integral calculus), which we will not cover. Almost always, in practice, people use proper priors so that posterior propriety is not in question, or use improper priors that are well known to lead to proper posteriors.

```
> ## (not run during knitting)
> y<- zprostate.lm$model[,1]
> X<- model.matrix(zprostate.lm)
> Xstar<- X[,-1]
> (n<- dim(X)[1])
> (p<- dim(X)[2])
> k<- p - 1
> XtX<- crossprod(X)
> XstartXstar<- XtX[-1,-1]
> bhat<- solve(XtX) * %t(X) * %*% y
> prebstarhat<- solve(XstartXstar) * %*% t(Xstar)
> m0star<- rep(0,k) ## betastar prior mean
>
> ## Prior variance for beta1-betak (i.e., for betastar not for beta0)
> V<- (log(10)/1.96)^2
> ## Prior precision for betastar
> Vinv<- V^{-1}
> B= Vinv * diag(k)
>
> ## FC df for sigma2
> nuhat<- n
> ## Intermediate computation for sigma2 FC
> SSE <- sum((y - X * %*% bhat)^2)
>
> nChain<- 3 ## number of chains (to be discussed in class)
> M<- 10000 ## number of MCMC iterations
> sigma2 <- matrix(NA,nChain,M+1) ## to store sigma2 samples
> beta<- array(NA,c(nChain,M+1,p)) ## to store beta values
>
> ## 3 (dispersed) sigma2 starting values, one to start each chain:
> sigma2[1,1]<- summary(zprostate.lm)$sigma^2 / 10 ## initial sigma20
> sigma2[2,1]<- summary(zprostate.lm)$sigma^2 ## initial sigma20
> sigma2[3,1]<- summary(zprostate.lm)$sigma^2 * 10 ## initial sigma20
>
```

```

> ## 3 beta0 starting values, one to start each chain:
> set.seed(5551212 + 90210)
> beta[1,1,1] <- rnorm(n=1, m=bhat[1], sd=sqrt(sigma2[1,1] / n))
> beta[2,1,1] <- rnorm(n=1, m=bhat[1], sd=sqrt(sigma2[2,1] / n))
> beta[3,1,1] <- rnorm(n=1, m=bhat[1], sd=sqrt(sigma2[3,1] / n))
>
> ## betastar starting values not required (see algorithm)
>
> library(mvtnorm)
>
> for (chain in 1:nChain){ ## chain loop
+   for (i in 1:M){ ## iteration loop
+     if(i %% 1000 == 0) print(paste0("Chain = ", chain,
+                                     ". Iteration = ", i, "."))
+
+     ##### FC for bstar (beta w/o beta0)
+     #### variance
+     bstarvar <- solve(sigma2[chain, i] ^ {-1} * XstartXstar + B)
+     #### mean
+     bstarhat <- prebstarhat %*% (y - beta[chain,i,1])
+     bstarmean <- bstarvar %*% (sigma2[chain,i] ^ {-1} *
+                                 XstartXstar %*% bstarhat +
+                                 Vinv * m0star)
+     #### sample bstar (beta w/o beta0)
+     beta[chain,i+1,-1] <- as.vector(rmvnrm(n=1, mean=bstarmean,
+                                             sigma=bstarvar))
+
+     ##### FC for beta0
+     #### mean
+     ystarbar <- mean(y - Xstar %*% beta[chain,i+1,-1])
+     ## sample beta0
+     beta[chain,i+1,1] <- rnorm(n=1, mean=ystarbar,
+                                 sd=sqrt(sigma2[chain,i] / n))
+
+     ##### FC for sigma2
+     ## See BDA 3 appendix A for generating scaled inv-chi2 (see also
+     ## Wakefield's expression (5.45)...typo?):
+     sigma2hat <- (SSE + crossprod(X %*% (beta[chain,i+1,] - bhat))) / nuhat
+     sigma2[chain, i+1] <- nuhat * sigma2hat / rchisq(n=1, df=nuhat)
+   } ## end iterations
+ } ## end chains

```

```
>  
> ## Good idea to save the fit to a file for later use:  
> save(list=c("beta", "sigma2"), file="zprostate.fit.RData")  
>  
> detach(package:mvtnorm)  
> rm(y,X,Xstar,n,p,k,XtX,XstartXstar,bhat,prebstarhat,m0star,V,Vinv,B,nuhat,SSE)
```

C.9.5 Posterior Convergence Diagnostics with coda

- **History & Corresponding Marginal Density Plots.** We compute history plots to diagnose how the chains of sampled values **converge and mix** (or not) throughout the history of iterations. We also show corresponding density ('smooth histogram') plots. We may use basic R code for such plots, but I choose to use the **coda** (COndvergence DiAgnostics) package, first converting our Gibbs sampling output to a `coda::mcmc.list` object to exploit coda method functions.

I see that, perhaps, we should have dispersed our starting values (omitted from the chains) a bit more, if only to see subsequent iterations' values converge to a common range of values, all three chains finding the posterior after some “period” of **burn-in** (or **warm-up**); again, convergence appears almost instantly here with no discernable warm-up period.

- **Potential Scale Reduction Factor (psrf) or Brooks-Gelman-Rubin (BGR) Statistic.** The Brooks-Gelman-Rubin (BGR) potential scale reduction factor (psrf) is a more quantitative measure of convergence. Loosely speaking, it compares the variability between chains to the variability within chains. If chains are far apart relative to the within chain variability, we get a large (> 1) psrf indicating lack of convergence. If chains are relatively close together, i.e., “mixing well,” then the psrf will be close to 1. A rule-of-thumb is that convergence has occurred if psrf $<$ about 1.1 or 1.2.

Our psrf values are all well below the rule-of-thumb values of 1.1 to 1.2, confirming the convergence seen in the history plots. (details omitted)

If you have parameters that are (linear) deterministic functions of one another (or nearly so) or if a parameter is set at some constant, then you may see an error when computing the multivariate psrf; see code.

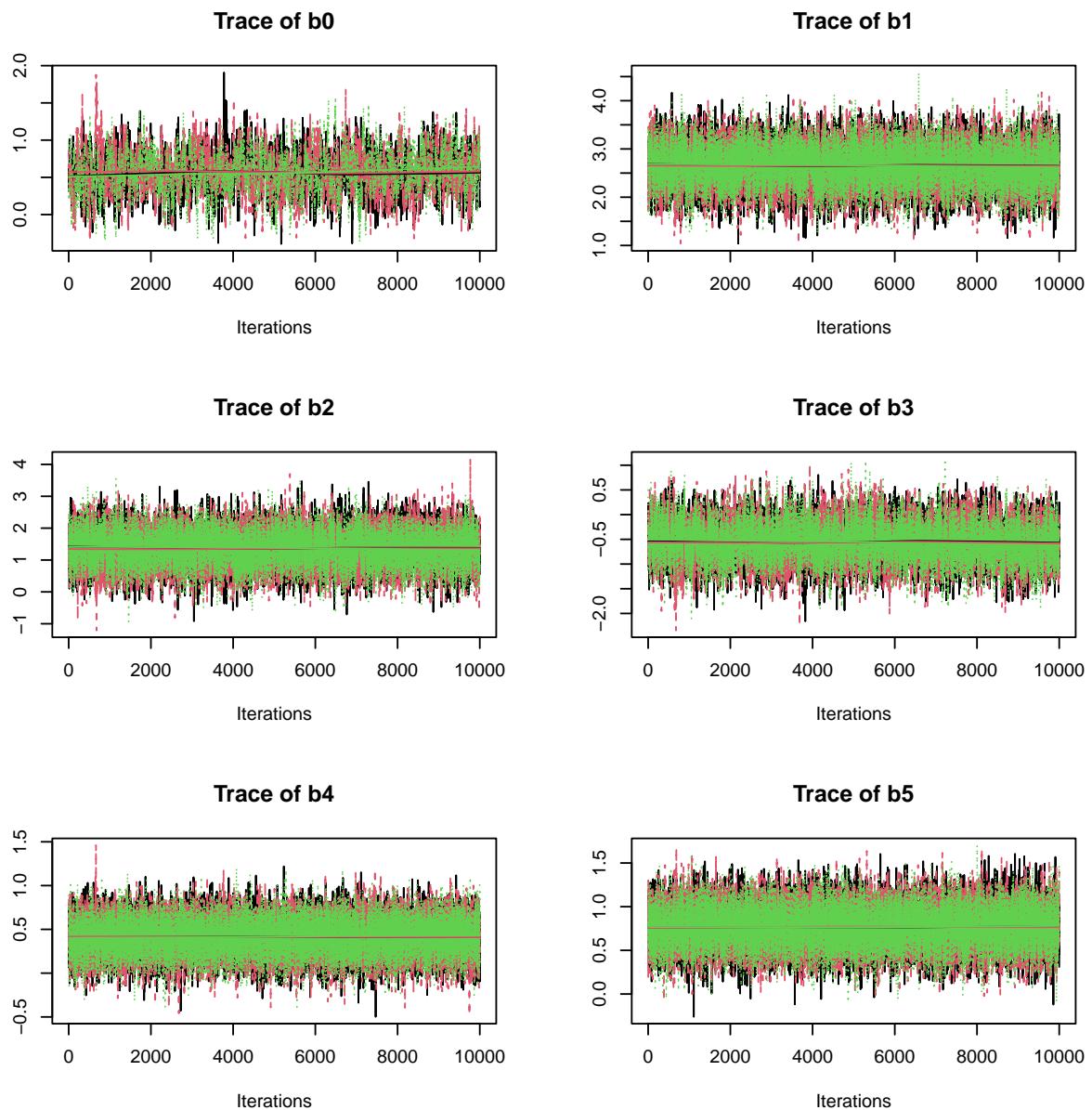
```
> load(file="zprostate.fit.RData")
> ## I remove initial values (or NAs) (using -1 index)
> c1<- cbind(beta[1,-1],sigma2[1,-1]) ## chain 1
> c2<- cbind(beta[2,-1],sigma2[2,-1]) ## chain 2
> c3<- cbind(beta[3,-1],sigma2[3,-1]) ## chain 3
> library(coda)
> zpGibbs<- mcmc.list(as.mcmc(c1),as.mcmc(c2),as.mcmc(c3))
> nchain(zpGibbs)

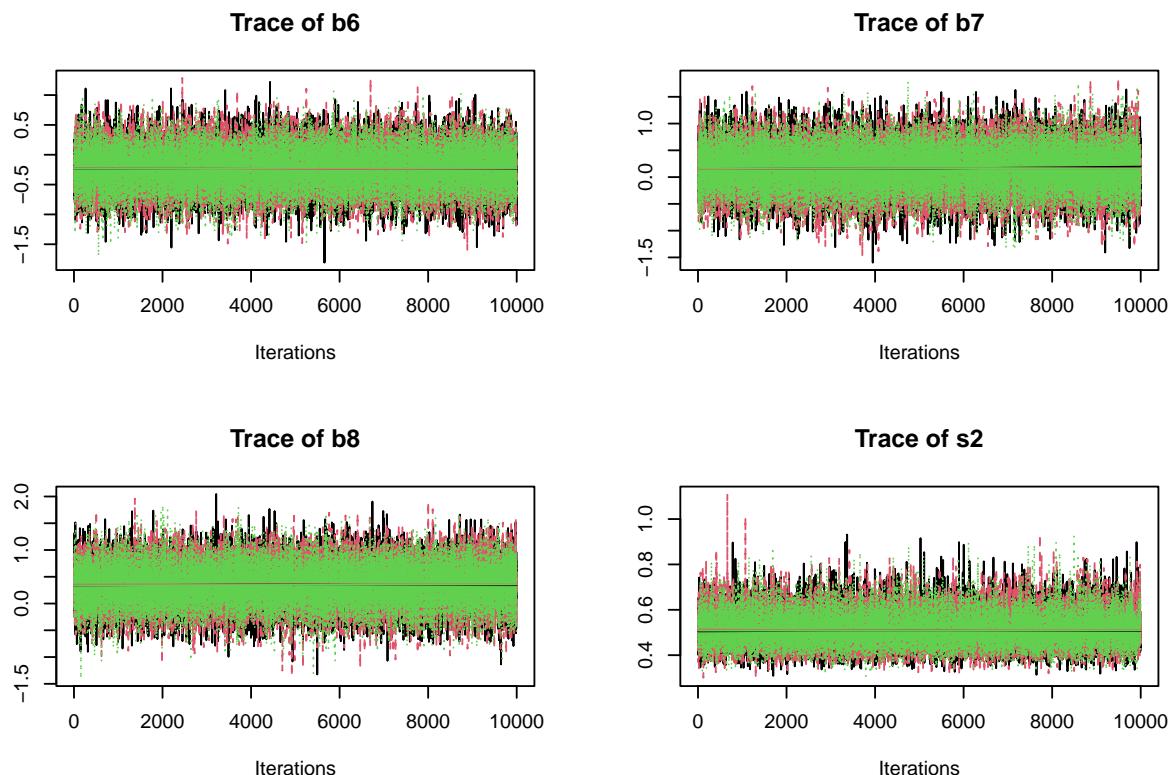
[1] 3

> nvar(zpGibbs)

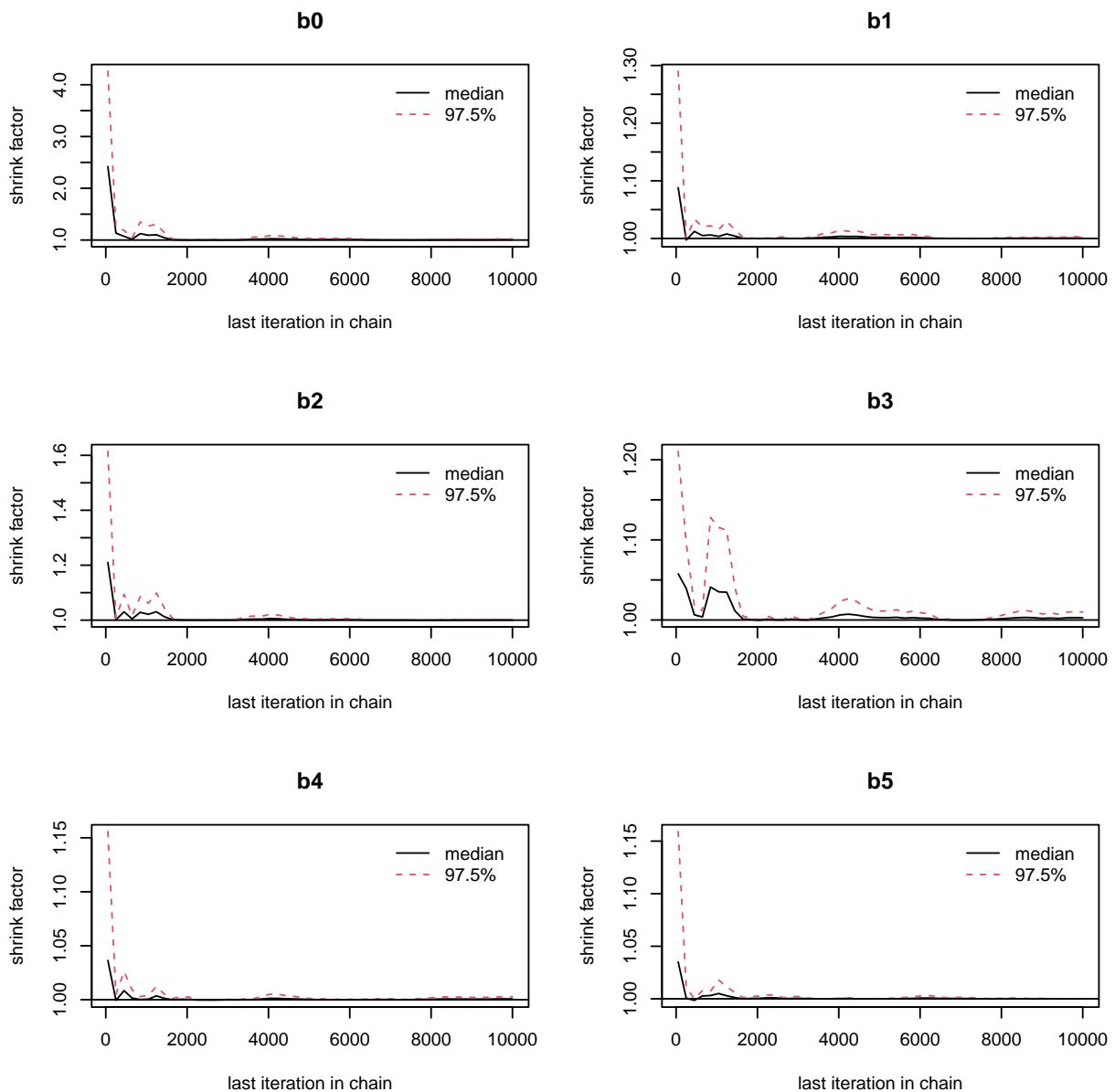
[1] 10

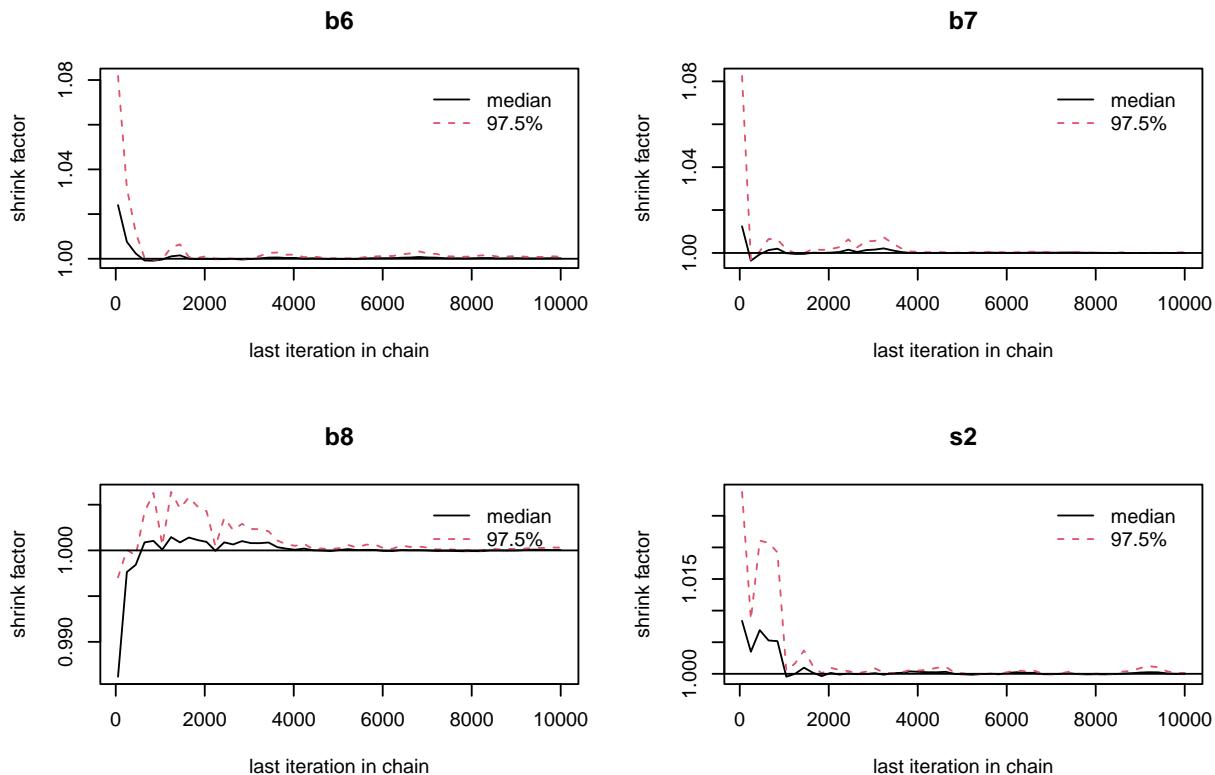
> varnames(zpGibbs)<- c(paste0("b",0:8), "s2")
> plot(zpGibbs, density=FALSE)
```





```
> gelman.plot(zpGibbs, ask=FALSE)
```





```
> gelman.diag(zpGibbs)
```

Potential scale reduction factors:

	Point est.	Upper C.I.
b0	1.01	1.02
b1	1.00	1.00
b2	1.00	1.00
b3	1.00	1.01

```
b4      1.00    1.00  
b5      1.00    1.00  
b6      1.00    1.00  
b7      1.00    1.00  
b8      1.00    1.00  
s2      1.00    1.00
```

Multivariate psrf

1.01

C.9.6 Posterior Summaries with coda

- **Discard Burn-in or Warm-up Samples.** Based on our convergence diagnostics of our Gibbs sampling, convergence, or mixing, occurs almost immediately, so we might discard only, say, the first 100 iterations from each chain before using the remaining samples to summarize the posterior. But, we will repeat sampling from this posterior, shortly, with Stan (using Hamiltonian Monte Carlo (HMC) behind the scenes), and we will see that we must omit the first 5000 iterations from Stan's chains, for reasons to be discussed when we get there. So, for comparison to Stan results, we omit the first 5000 iterations from each of our Gibbs chains, leaving

$$3 \times 5000 = 15000$$

iterations with which to summarize the posterior, the same number as we will use from Stan. We use the generic `stats:::window` function (which calls the appropriate method function) to summarize the appropriate iterations contained in our `coda::mcmc.list` object, created above.

- **coda.** Again, the results are simply random samples from a distribution, and we may use basic summarizing functions in R (e.g., `mean`, `sd`, `quantile`, etc.), but, again, we use `coda`, not only just for convergence diagnostics, as above, but also for summarizing the posterior, here.

- **Commentary on Bayesian and Frequentist Results.** The effect signs are the same between the Bayesian (mean and median) and frequentist LS estimates, and intervals are qualitatively comparable, with the Bayesian credible intervals for all effects covering or not covering zero in the same manner as the frequentist confidence intervals, with the exception of the intercept. For the Bayesian analysis, some effects (means) appear to be shrunk toward the zero prior mean from their LS counterpart (b1 lcavol; b2 lweight; b3 age; b6 lcp; b8 pgg45) while others are similar (b5 svi) or slightly larger (b4 lbph; b7 gleason) in absolute value than their frequentist estimates.

```
> ## Obtain the last 5000 iterations of each chain (using zpGibbs mcmc.list
> ## object created previously):
> zpGibbs5to10k<- window(zpGibbs, start=5001, end=10000)
>
> ## The window result is also an mcmc.list (coda) object, from which we easily
> ## obtain a summary of the posterior (sample thereof anyway).
> class(zpGibbs5to10k)

[1] "mcmc.list"

> ## Bayes summary:
> (bayessum<- summary(zpGibbs5to10k))
```

Iterations = 5001:10000
 Thinning interval = 1
 Number of chains = 3
 Sample size per chain = 5000

1. Empirical mean and standard deviation for each variable, plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE	SE
b0	0.563	0.2884	0.002355	0.012901	
b1	2.653	0.4227	0.003452	0.008011	
b2	1.369	0.5647	0.004611	0.010055	
b3	-0.558	0.4014	0.003278	0.009660	

```
b4  0.415 0.2126 0.001736      0.001812
b5  0.768 0.2384 0.001947      0.002196
b6 -0.226 0.3650 0.002981      0.003004
b7  0.168 0.4300 0.003511      0.003480
b8  0.352 0.4084 0.003335      0.003346
s2  0.514 0.0788 0.000644      0.000703
```

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
b0	0.01114	0.3635	0.557	0.7619	1.127
b1	1.80231	2.3729	2.656	2.9395	3.468
b2	0.25928	0.9879	1.372	1.7515	2.456
b3	-1.34086	-0.8304	-0.559	-0.2887	0.241
b4	-0.00269	0.2704	0.415	0.5601	0.831
b5	0.30675	0.6077	0.770	0.9259	1.238
b6	-0.93442	-0.4763	-0.225	0.0207	0.492
b7	-0.68437	-0.1130	0.169	0.4589	1.015
b8	-0.45634	0.0779	0.352	0.6282	1.152
s2	0.38338	0.4580	0.507	0.5614	0.688

```
> ## Compare to the LS freq summary:
> cbind("Bmean"=bayessum$statistics[, "Mean"] ,
+        "LS"=c(coef(zprostate.lm),
+               summary(zprostate.lm)$sigma^2))
```

	Bmean	LS
b0	0.56293	0.42142
b1	2.65348	3.03377
b2	1.36914	1.69639
b3	-0.55759	-0.74621
b4	0.41519	0.39745
b5	0.76783	0.76616
b6	-0.22627	-0.45253
b7	0.16840	0.13542
b8	0.35200	0.45252
s2	0.51412	0.50185

```
> cbind(confint(zprostate.lm), bayessum$quantiles[-10,c("2.5%","97.5%")])
```

	2.5 %	97.5 %	2.5%	97.5%
(Intercept)	-0.16638	1.009229	0.0111376	1.12724
z.lcavol	2.13079	3.936757	1.8023085	3.46802

z.lweight	0.43525	2.957535	0.2592751	2.45567
z.age	-1.58994	0.097518	-1.3408642	0.24122
z.lbph	-0.03379	0.828687	-0.0026903	0.83097
z.svi	0.28064	1.251670	0.3067486	1.23828
z.lcp	-1.22855	0.323483	-0.9344241	0.49234
z.gleason	-0.80336	1.074208	-0.6843657	1.01454
z.pgg45	-0.42609	1.331139	-0.4563416	1.15234

C.9.7 Regression Function and Posterior Predictive

- **Regression Function.** We may want to estimate the (linear model of the) regression function, $E(Y^* | \mathbf{x}^*) = \mathbf{x}^{*t} \boldsymbol{\beta}$, which is the **mean** $\log(psa)$ value for a population of individuals with characteristics \mathbf{x}^* . As Bayesians, we want the posterior,

$$[\mathbf{x}^{*t} \boldsymbol{\beta} | \mathbf{y}].$$

(We are still implicitly conditioning on covariates \mathbf{x}^* and \mathbf{X} .) We simply use the posterior sample values of $\boldsymbol{\beta} | \mathbf{y}$ to compute (deterministically) $\mathbf{x}^{*t} \boldsymbol{\beta} | \mathbf{y}$ in a one-for-one fashion, thus obtaining a sample from the posterior of $\mathbf{x}^{*t} \boldsymbol{\beta} | \mathbf{y}$, summarizing as we see fit, e.g., credible interval. That's easy!

- **Posterior Predictive.** Or, we may wish to predict the $\log(psa)$ value, $Y^* | \mathbf{x}^*$, of an individual from this population. As Bayesians, we want the posterior predictive distribution,

$$[y^* | \mathbf{y}] = \int [y^* | \boldsymbol{\beta}, \sigma^2, \mathbf{y}] [\boldsymbol{\beta}, \sigma^2 | \mathbf{y}] d\boldsymbol{\beta} d\sigma^2$$

(now omitting notationally the conditioning on \mathbf{x}^* as we have been omitting it in the Bayesian context).

- **Composition Sampling.** For this posterior predictive, we use composition sampling (§C.1.4). We use the Gibbs samples of $\boldsymbol{\beta}, \sigma^2 | \mathbf{y}$ from $[\boldsymbol{\beta}, \sigma^2 | \mathbf{y}]$, then, for each of these sample values, we generate a value of $Y^* | \boldsymbol{\beta}, \sigma^2, \mathbf{y}$ from the conditional posterior predictive, $[y^* | \boldsymbol{\beta}, \sigma^2, \mathbf{y}]$,

which is just our normal regression model with mean $\mathbf{x}^* \boldsymbol{\beta}$ (samples already obtained as described in the previous item) and error variance σ^2 and which actually does not depend on the observed outputs \mathbf{y} because our model says outputs, observed or not, are conditionally independent (given $\boldsymbol{\beta}$ and σ^2). Thus, we will have sample values from the joint posterior of $Y^*, \mathbf{x}^* \boldsymbol{\beta}, \boldsymbol{\beta}, \sigma^2 | \mathbf{y}$, thus, we will have a sample of our desired marginal posteriors, $\mathbf{x}^* \boldsymbol{\beta} | \mathbf{y}$ and $Y^* | \mathbf{x}^*, \mathbf{y}$ which we may summarize as we wish. (We are essentially using Monte Carlo integration to “avoid” the integral denoted in the posterior predictive distribution, above.)

- **Lecture 4.** We already know how to obtain a frequentist confidence interval and prediction interval, respectively, for these targets; see Lecture 4 and elsewhere.
- **Question.** Which of $\mathbf{x}^* \boldsymbol{\beta}$ or $Y^* | \mathbf{y}$ would a doctor use to diagnose an individual patient? (We should know this from our frequentist estimation/prediction interval work (Lecture 4), but the question/answer is the same in the Bayesian context.)
- **Commentary on Bayesian and Frequentist Results.** The Bayesian and frequentist intervals for the regression function linear model, $\mathbf{x}^* \boldsymbol{\beta}$, are very comparable (lines `xstarbetaBayes` and `xstarbeta`, respectively) as are the Bayesian and frequentist and intervals for $Y^* | \mathbf{x}$ (lines `ystarBayes` and `ystar`, respectively).

We see, like the frequentist intervals we've seen before (Lecture 4), we estimate the mean (regression function) $\mathbf{x}^* \boldsymbol{\beta}$ more precisely than we do an individual unobserved value $Y^* | \mathbf{x}$.

```
> X<- model.matrix(zprostate.lm)
> (xstar<- apply(X,2,median))  ## <- typical (median) patient

(Intercept)    z.lcavol   z.lweight      z.age     z.lbph
  1.00000     0.54063    0.33437     0.63158    0.45424
  z.svi        z.lcp     z.gleason    z.pgg45
  0.00000     0.13700    0.33333     0.15000
```

```

> varnames(zpGibbs5to10k)
[1] "b0" "b1" "b2" "b3" "b4" "b5" "b6" "b7" "b8" "s2"

> tmp<- as.matrix(zpGibbs5to10k)
> ppred<- t(apply(tmp, 1, function(theta, xstar) {
+   xstarbeta<- sum(theta[1:9] * xstar) ##  $x^{\top} \theta$  / y
+   ystar<- rnorm(n=1, xstarbeta, sd=sqrt(theta[10])) ##  $y^* / \sigma$ 
+   c(xstarbeta=xstarbeta, ystar=ystar)
+ },
+ xstar=xstar
+ ))
>
> xstarbetaCI<- quantile(ppred[, "xstarbeta"], c(0.025, 0.975))
> ystarCI<- quantile(ppred[, "ystar"], c(0.025, 0.975))
>
> freqCI<- predict(zprostate.lm, newdata=as.data.frame(t(xstar)),
+                     interval="confidence")
> freqPI<- predict(zprostate.lm, newdata=as.data.frame(t(xstar)),
+                     interval="prediction")
>
> rbind(xstarbetaBayes=xstarbetaCI, ystarBayes=ystarCI,
+        xstarbeta = freqCI[-1], ystar=freqPI[-1])

```

	2.5%	97.5%
xstarbetaBayes	2.15858	2.5772
ystarBayes	0.95582	3.7940
xstarbeta	2.17243	2.6057
ystar	0.96465	3.8134

C.10 HMC in Stan: Prostate Data Example with “Combo” Prior

Above, we implemented Gibbs sampling ([Wak13, §3.8.4]) “by hand.” Gibbs sampling is a special case of the Metropolis-Hastings algorithm ([Wak13, §3.8.2]). Both fall under the guise of Markov chain Monte Carlo sampling methods (MCMC; [Wak13, §3.8]). Here, we use another type of MCMC sampling known as Hamiltonian Monte Carlo (HMC) (sometimes “hybrid Monte Carlo,” but not in the same sense as [Wak13, §3.8.5]). We do not discuss the details of HMC, but

merely use Stan, via the RStan interface, to implement HMC for the current running prior as we have been following in [Wak13, §5.12]. Of course, with large enough MCMC sample size, our Gibbs sampling results should be practically the same as the HMC results; we're sampling from the same posterior, after all. But, you may find the relatively high level interface of Stan to be more convenient compared to programming the HMC algorithm or other MCMC (e.g., Gibbs) yourself.

I do not discuss the preliminary details of installing Stan or the `rstan` package (<https://mc-stan.org/users/interfaces/rstan.html>). You should be able to do that. The `rstan` package comes with its own documentation, as most any other R package. Also, the Stan language comes with its own set of documentation (<https://mc-stan.org/users/documentation/>). I use this documentation a lot. Incidentally, there are other interfaces to the Stan language besides RStan, e.g., PyStan, for Python enthusiasts (<https://mc-stan.org/users/interfaces/>). Importantly, it seems that the `cmdstan` and `cmdstanr` interfaces are the first to incorporate new changes in the Stan language, which may take a long time to be implemented in the other interfaces.

In the following subsections, we implement the current running Bayesian model example in Stan's **program blocks**. Together, these blocks constitute a Stan language program, which will be parsed and translated to C++, then compiled and linked into object code, ready to be loaded and run. More discussion in class.

C.10.1 Functions Block

```
functions {
    // nothing for this example
}
```

C.10.2 Data Block

```
data{
    int<lower=1> N;
    int<lower=1> p;
    matrix[N,p] X;
```

```
    vector[N] y;
    vector[p-1] m0;
    vector[p] xstar;
}
```

C.10.3 Transformed Data Block

```
transformed data{
    // Cheap illustration transformed data using prior sd of beta's:
    real rootV=log(10)/1.96;
}
```

C.10.4 Parameters Block

```
parameters{
    vector[p] beta;
    real lnsigma2;
}
```

C.10.5 Transformed Parameters Block

```
transformed parameters{
    real<lower=0> sigma2 = exp(lnsigma2);
}
```

C.10.6 Model Block

As I mentioned before, we are merely specifying the joint model in the numerator of Bayes theorem and let Stan do the work. No derivations required. (Still, probabilistic programming languages may not warn you if a posterior is improper, so be carefull when using improper priors that you are not sure lead to proper posteriors.)

```

model{
  y ~ normal(X*beta, sqrt(sigma2));
  // Prior for beta1-betak (excluding intercept beta0):
  for(j in 2:p) beta[j]~ normal(m0[j-1], rootV);;

  *****
}

```

Other parameters (in parameters block), without an explicit prior specification, here, will receive a flat prior over their implied support. For us, this means [beta0] will be proportional to 1 over (-inf,inf), improper as desired, and that [lnsigma2] is proportional to 1 on (-inf, inf) (improper), which corresponds to [sigma2] proportional to 1/sigma2, improper as desired (details omitted).

```
*****/
```

```
}
```

C.10.7 Generated Quantities Block

```

generated quantities{
  // Composition sampling. (no longer MCMC at this point)
  real xstarbeta = dot_product(xstar, beta);
  real ystar = normal_rng(xstarbeta, sqrt(sigma2));
}

```

C.10.8 Altogether for Stan

I have put all of the above program block code into one ASCII (text) file called zprostate1.stan. The next chunk reads and displays the file contents, with all of the program blocks together in a working Stan program.

```

> writeLines(readLines("./Stan/zprostate1.stan"))

functions {
  // nothing for this example
}

```

```

data{
    int<lower=1> N;
    int<lower=1> p;
    matrix[N,p] X;
    vector[N] y;
    vector[p-1] m0;
    vector[p] xstar;
}

transformed data{
    // Cheap illustration of transformed data using prior sd of beta's:
    real rootV=log(10)/1.96;
}

parameters{
    vector[p] beta;
    real lnsigma2;
}

transformed parameters{
    real<lower=0> sigma2 = exp(lnsigma2);
}

model{
    y ~ normal(X*beta, sqrt(sigma2));
    // Prior for beta1-betak (excluding intercept beta0):
    for(j in 2:p) beta[j]~ normal(m0[j-1], rootV);

    *****
    Other parameters (in parameters block), without an explicit prior
    specification, here, will receive a flat prior over their
    implied support. For us, this means [beta0] will be proportional
    to 1 over (-inf,inf), improper as desired, and that
    [lnsigma2] is proportional to 1 on (-inf, inf) (improper), which
    corresponds to [sigma2] proportional to 1/sigma2, improper as
    desired (details omitted).
}

****/
}

generated quantities{

```

```
// Composition sampling. (no longer MCMC at this point)
real xstarbeta = dot_product(xstar, beta);
real ystar = normal_rng(xstarbeta, sqrt(sigma2));
}
```

C.10.9 Translate Stan to C++ with `stanc`

The next chunk uses the `rstan` function `stanc` to parse and translate Stan code in `zprostate1.stan` into a C++ file, checking for Stan syntax errors. Address any errors reported by editing the Stan code in `zprostate1.stan` until `stanc` returns no errors/warnings.

```
> library(rstan,quietly=TRUE)

rstan (Version 2.21.5, GitRev: 2e1f913d3ca3)
For execution on a local, multicore CPU with excess RAM we recommend calling
options(mc.cores = parallel::detectCores()).
To avoid recompilation of unchanged Stan programs, we recommend calling
rstan_options(auto_write = TRUE)

Attaching package: 'rstan'
The following object is masked from 'package:coda':
  traceplot

> options(mc.cores = parallel::detectCores())
> rstan_options(auto_write = TRUE)
> zprostate1.stanc<- stanc(file="./Stan/zprostate1.stan")
```

C.10.10 Make an Executable Stan Model with `stan_model`

The next chunk compiles the C++ code in the object returned by `stanc`, above, into a `stanmodel` object, which references compiled/linked object code ready to be (re)used in sampling, below, with the `sampling` function in `rstan`.

```
> zprostate1.stanmod<- stan_model(stanc_ret=zprostate1.stanc)
```

C.10.11 Data List for Stan

Now we are ready to create data and initial value lists to pass to our Stan model to obtain samples from the posterior distribution. If we do not specify initial values, then Stan generates these, which may work fine. In the case when Stan's initial values cause sampling to fail, then you should pass (better) initial values to Stan in a list.

We use a previous `lm` object to help us along.

```
> ## Using objects computed previously.
> X<- model.matrix(zprostate.lm)
> zprostate1.data<- list(
+   N=dim(X)[1],
+   p=dim(X)[2],
+   X=X,
+   y=zprostate.lm$model[,1],
+   m0=rep(0, dim(X)[2]-1),
+   xstar=xstar
+ )
> rm(X,xstar)
```

C.10.12 List of Initial Value Lists for Stan

```
> library(mvtnorm)
> set.seed(20500 + 5150 + 24601)
> binit<- rmvnorm(n=3,mean=coef(zprostate.lm),sigma=3*vcov(zprostate.lm))
> attach(zprostate1.data)
> sigma2inits<- NULL
> sigma2inits<- c(sigma2inits, sum((y - X%*%binit[1,])^2)/(N-p))
> sigma2inits<- c(sigma2inits, sum((y - X%*%binit[2,])^2)/(N-p))
> sigma2inits<- c(sigma2inits, sum((y - X%*%binit[3,])^2)/(N-p))
> detach(zprostate1.data)
> zprostate1.init<- list(
+   list(beta=binit[1,], lnsigma2=log(sigma2inits[1])),
+   list(beta=binit[2,], lnsigma2=log(sigma2inits[2])),
+   list(beta=binit[3,], lnsigma2=log(sigma2inits[3])))
> detach(package:mvtnorm)
> rm(binit, sigma2inits)
```

C.10.13 Executing a Stan Model with sampling

```
> zprostate1.fit<- sampling(zprostate1.stanmod,
+                               data=zprostate1.data,
+                               pars=c("beta", "sigma2", "xstarbeta", "ystar"),
+                               chains=3,
+                               iter=10000,
+                               warmup=5000,
+                               init=zprostate1.init,
+                               refresh=1000)
> ## Good idea to save the fit to a file for later use:
> save(list=c("zprostate1.stanc", "zprostate1.stanmod",
+                   "zprostate1.fit"), file="Stan/zprostate1.fit.RData")
```

C.10.14 Posterior Convergence Diagnostics with coda

NOTE: **rstan** has its own posterior summary functions. But, because (at one time) these functions do (did) not work on my computer, and because **coda** (COvergence DiAgnostics) is a well-known and popular package to summarize posteriors, I use **coda** instead, as we used for our previous Gibbs sampling. Also, **rstan::As.mcmc.list** automatically creates a **coda mcmc.list** object from a **rstan** object, but **rstan::As.mcmc.list** omits the warmup iterations! I use **rstan::extract**, instead, so we can see the warmup, too, at the expense of a bit of extra code to get the result into a **coda mcmc.list** object.

Discussion in class, of course.

```
> load(file=".~/Stan/zprostate1.fit.RData")
> ## Transform Stan model fit (stanfit class) to a coda mcmc.list object
> ## for use in coda. rstan::As.mcmc.list() omits warmup! I want it.
> ## So, I use rstan::extract() instead. (Note capital A)
> ## zprostate1.mcmc<- rstan::As.mcmc.list(zprostate1.fit)
> class(zprostate1.fit)

[1] "stanfit"
attr("package")
[1] "rstan"

> zprostate1.array<- rstan::extract(zprostate1.fit, permuted=FALSE,
+                                    inc_warmup=TRUE)
> dim(zprostate1.array)
```

```
[1] 10000      3      13

> nchains<- dim(zprostate1.array)[2]
> zprostate1.mcmc.list<- vector("list", nchains)
> for(chain in 1:nchains){
+   zprostate1.mcmc.list[[chain]]<- coda::as.mcmc(zprostate1.array[,chain,])
+ }
> zprostate1.mcmc<- coda::as.mcmc.list(zprostate1.mcmc.list)
> class(zprostate1.mcmc)

[1] "mcmc.list"

> coda::nchain(zprostate1.mcmc)

[1] 3

> coda::niter(zprostate1.mcmc)

[1] 10000

> coda::nvar(zprostate1.mcmc)

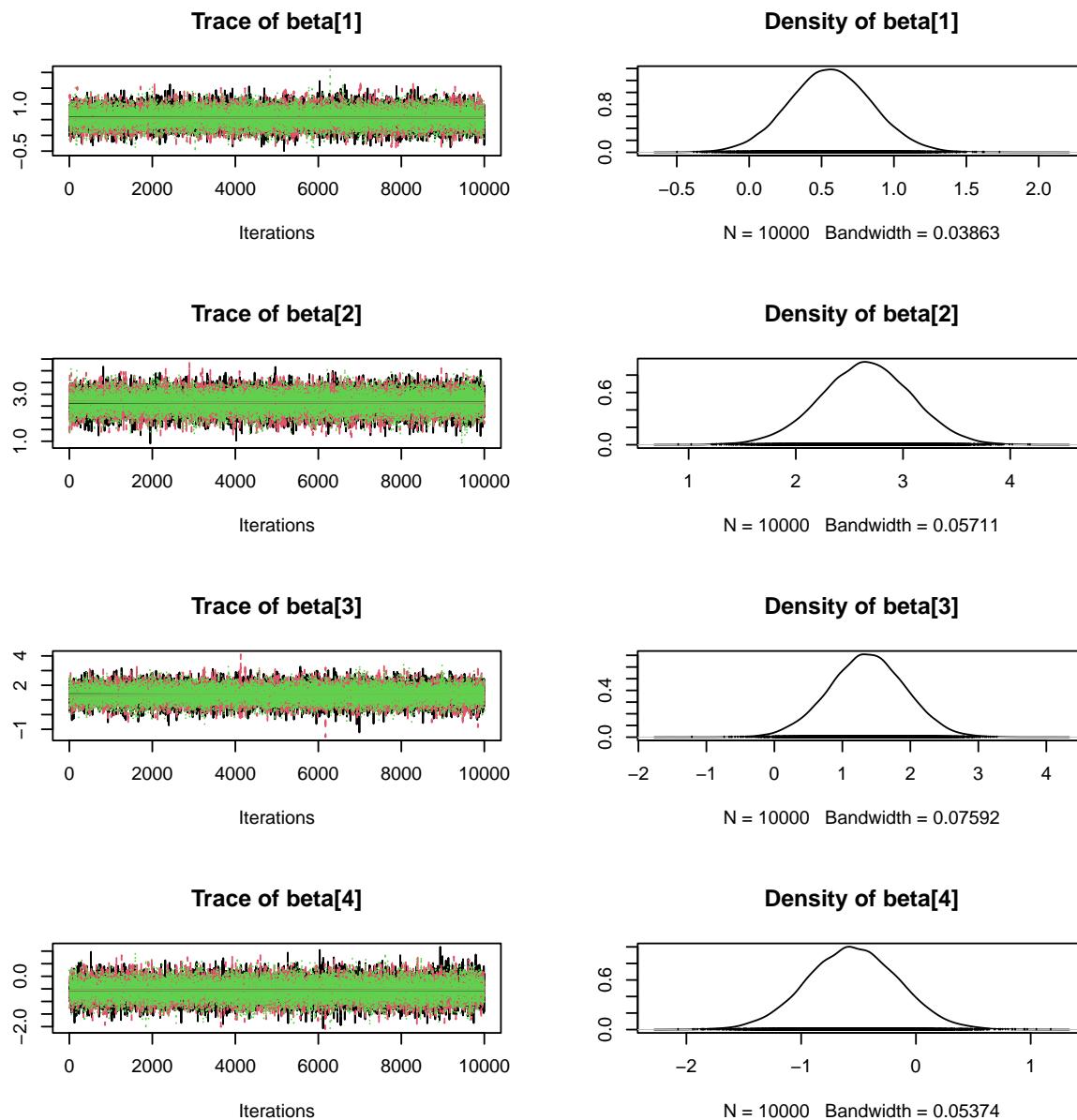
[1] 13

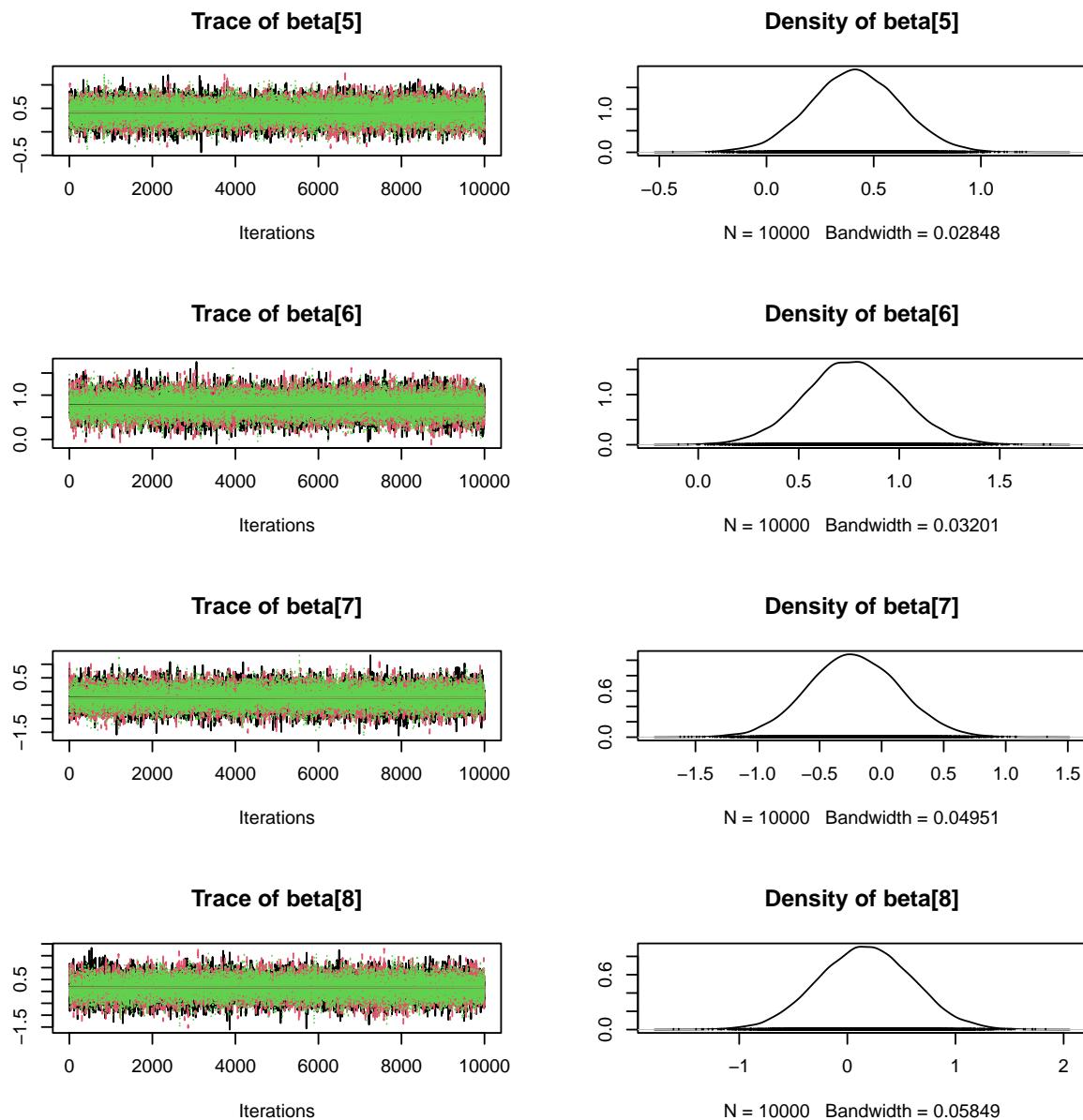
> coda::varnames(zprostate1.mcmc)

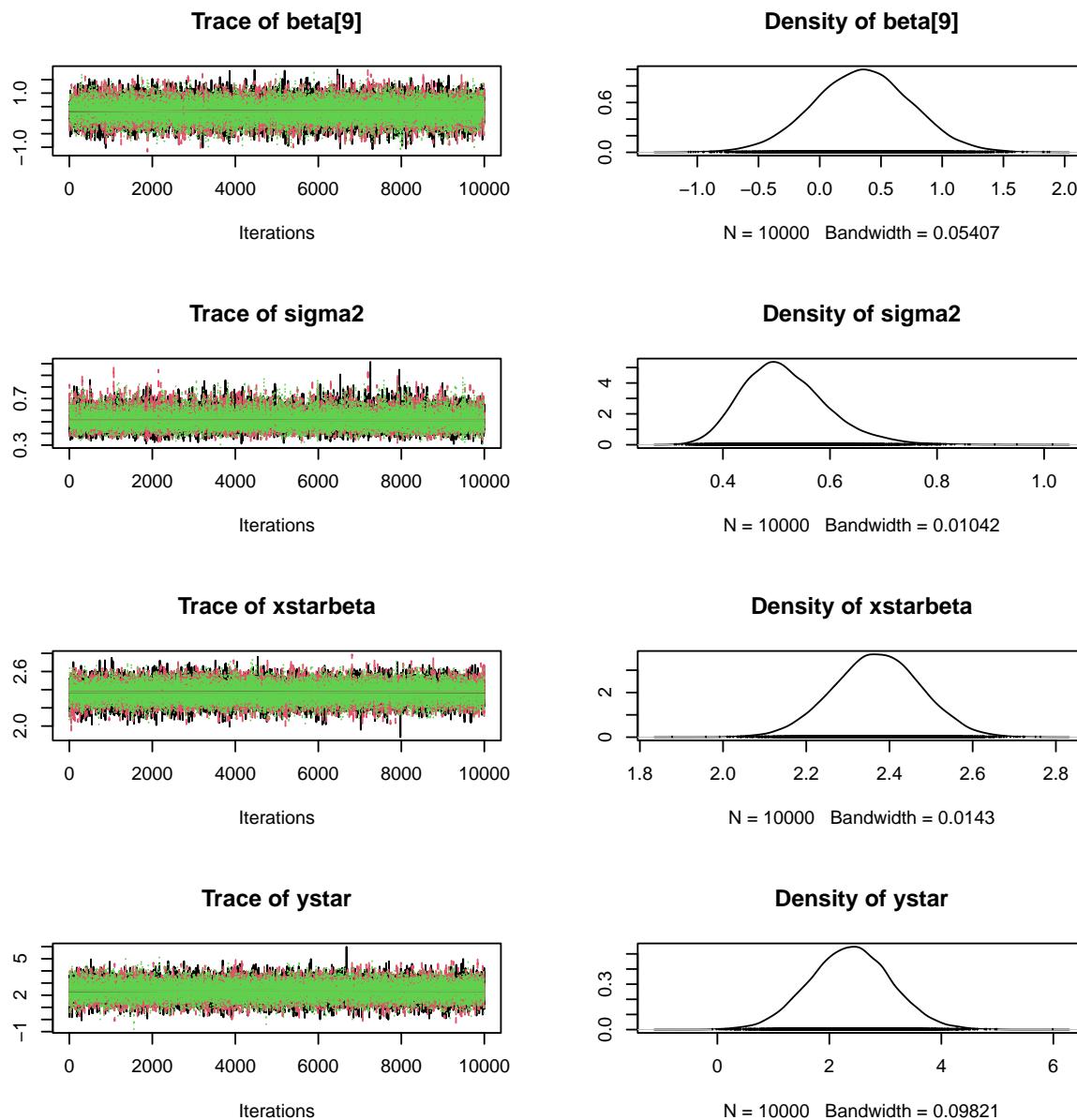
[1] "beta[1]"    "beta[2]"    "beta[3]"    "beta[4]"    "beta[5]"
[6] "beta[6]"    "beta[7]"    "beta[8]"    "beta[9]"    "sigma2"
[11] "xstarbeta" "ystar"     "lp__"
```

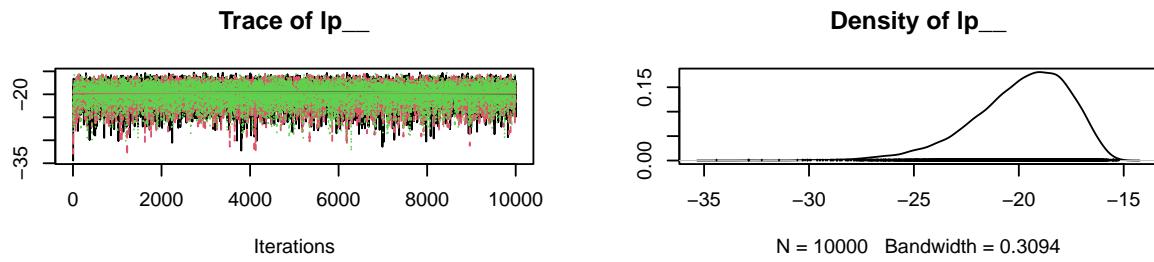
Typical graphical convergence diagnostics: trace plots and density plots.

```
> coda:::plot.mcmc.list(zprostate1.mcmc)
```







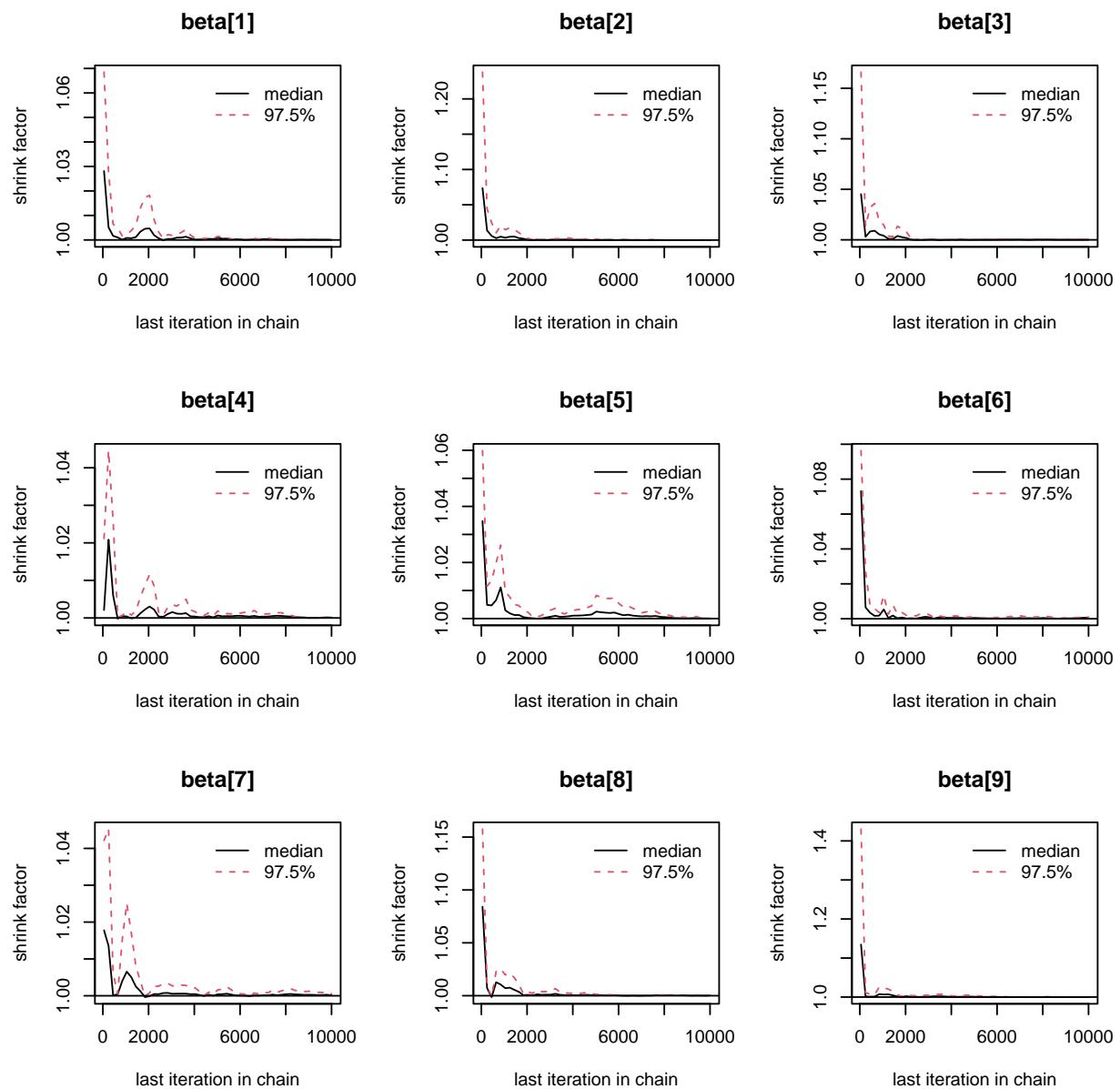


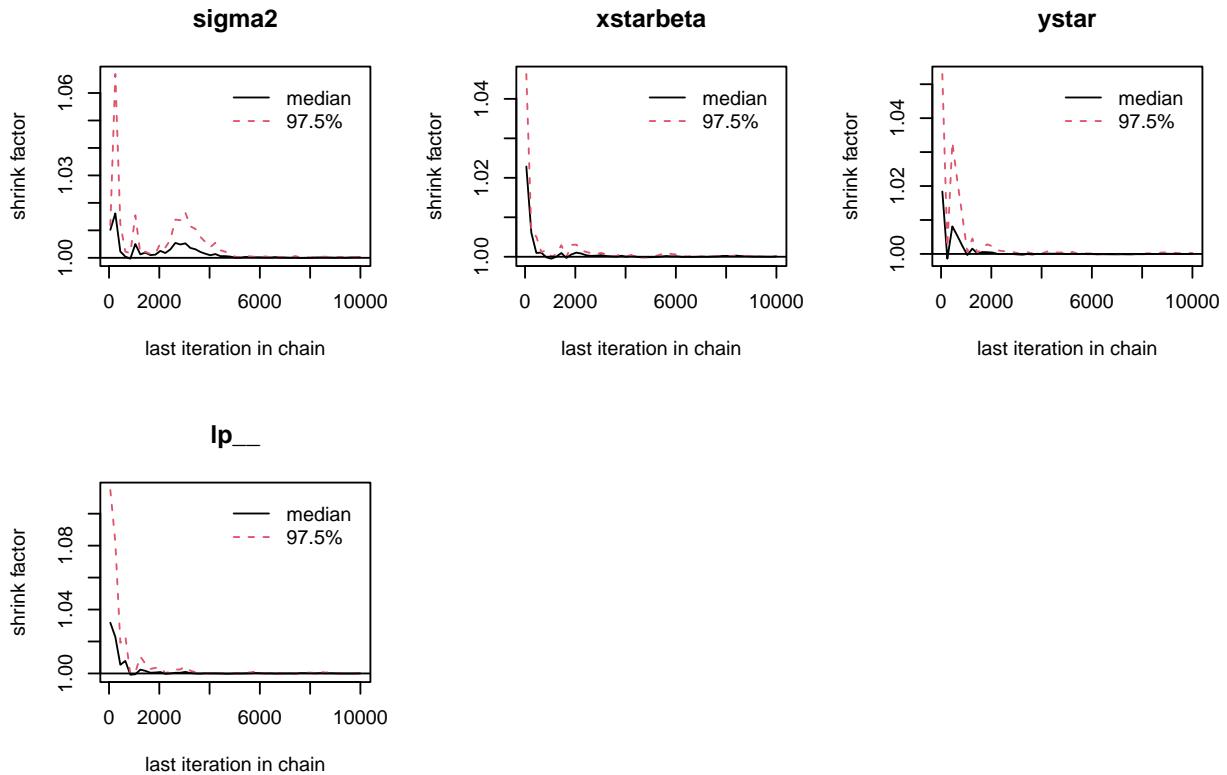
We introduced the Brooks-Gelman-Rubin (BGR) potential scale reduction factor (psrf) when diagnosing our Gibbs sampling. Again, loosely speaking, it compares the variability between chains to the variability within chains. If chains are far apart relative to the within chain variability, we get a large (> 1) psrf indicating lack of convergence. If chains are relatively close together,

i.e., “mixing well,” then the psrf we be close to 1. A rule-of-thumb is that convergence has occurred if psrf < about 1.1 or 1.2.

Together with the history plots, the psrf leaves little doubt: convergence is quick.

```
> ## Graphical display of next chunk:
> coda::gelman.plot(zprostate1.mcmc, ask=FALSE)
```





```
> ## Numerical display of previous plot:
> ## (deterministic function xstarbeta causes error by default)
> ## coda::gelman.diag(zprostate1.mcmc)
>
> ## No error if omit deterministic functions of posterior
> coda::gelman.diag(zprostate1.mcmc[,-11])
```

Potential scale reduction factors:

```
      Point est. Upper C.I.  
beta[1]      1      1  
beta[2]      1      1  
beta[3]      1      1  
beta[4]      1      1  
beta[5]      1      1  
beta[6]      1      1  
beta[7]      1      1  
beta[8]      1      1  
beta[9]      1      1  
sigma2       1      1  
ystar        1      1  
lp__         1      1  
  
Multivariate psrf  
  
1  
  
> ## Or, no error if omit the multivariate psrf  
> coda::gelman.diag(zprostate1.mcmc, multivariate=FALSE)  
  
Potential scale reduction factors:  
  
      Point est. Upper C.I.  
beta[1]      1      1  
beta[2]      1      1  
beta[3]      1      1  
beta[4]      1      1  
beta[5]      1      1  
beta[6]      1      1  
beta[7]      1      1  
beta[8]      1      1  
beta[9]      1      1  
sigma2       1      1  
xstarbeta   1      1  
ystar        1      1  
lp__         1      1
```

C.10.15 Posterior Summaries with coda

Once we have determined convergence, we may summarize the posterior, as we did with our Gibbs samples. We **do not** use any iterations before convergence has occurred. Why? Also, we **do not** use any iterations during which the sampling procedure is somehow **adapted**. There was no adaption in our Gibbs sampling, so we merely had to decide when convergence occurred, as guided by history plots and the BGR psrf. However, Stan adapts sampling during its “warmup” iterations, so we must omit these iterations, even if they somehow look good, and, if necessary, omit more if convergence has not yet occurred after the adaption period. Again, as we say before, we use the **stats::window** (generic) function to omit iterations.

Note, we often hear of **burn-in**, **warm-up** or **adaption phase**, and they may mean more or less the same thing: the algorithm has to burn in or warm up or adapt. **Adaptive phase** refers particularly to those iterations where sampling is somehow adapted; as we said, **do not** use these iterations, and pure Gibbs sampling, as we saw previously, does not adapt. **Burn-in** or **warm-up** generally refers to the number of iterations at the beginning of chains before convergence, during which adaption may or may not have occurred. Perhaps confusingly, Stan adapts sampling during its “**warmup**” iterations, but, as we should understand, convergence may occur much later (not in our case). We did not adapt our Gibbs sampling, and convergence occurred very early, and we omitted 5000 Gibbs iterations only for comparison with Stan output, from which we also omitted the initial 5000 iterations from each chain because of adaption.

```
> ## Stan adapted for 5000 iterations, so we omit these.  
> (zprostate1.psum<- summary(window(zprostate1.mcmc, start=5001)))
```

```
Iterations = 5001:10000  
Thinning interval = 1  
Number of chains = 3  
Sample size per chain = 5000
```

1. Empirical mean and standard deviation for each variable,
plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
beta[1]	0.560	0.2896	0.002364	0.002638
beta[2]	2.653	0.4247	0.003468	0.003555
beta[3]	1.369	0.5686	0.004643	0.004577
beta[4]	-0.546	0.4021	0.003283	0.003423
beta[5]	0.410	0.2116	0.001728	0.001682
beta[6]	0.770	0.2371	0.001936	0.001865
beta[7]	-0.225	0.3667	0.002994	0.003108
beta[8]	0.166	0.4321	0.003528	0.003526
beta[9]	0.348	0.4000	0.003266	0.003406
sigma2	0.516	0.0795	0.000649	0.000640
xstarbeta	2.371	0.1048	0.000855	0.000797
ystar	2.374	0.7264	0.005931	0.005931
lp__	-19.907	2.3159	0.018910	0.029388

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta[1]	-0.00970	0.3663	0.559	0.7527	1.134
beta[2]	1.79770	2.3720	2.657	2.9347	3.484
beta[3]	0.23506	0.9908	1.373	1.7533	2.462
beta[4]	-1.34167	-0.8152	-0.547	-0.2772	0.241
beta[5]	-0.00496	0.2691	0.410	0.5537	0.825
beta[6]	0.30699	0.6119	0.768	0.9275	1.241
beta[7]	-0.94342	-0.4740	-0.230	0.0255	0.496
beta[8]	-0.68437	-0.1285	0.165	0.4569	1.025
beta[9]	-0.43818	0.0792	0.347	0.6181	1.130
sigma2	0.38236	0.4589	0.508	0.5631	0.694
xstarbeta	2.16436	2.3003	2.371	2.4410	2.572
ystar	0.96088	1.8844	2.380	2.8646	3.791
lp__	-25.35061	-21.2406	-19.584	-18.1952	-16.394

C.11 Prostate Data Example Summary

Here, we compare the results from

- the improper prior analysis (§C.4 & C.6) wherein we get the same *numerical* results as previous frequentist analysis;

2. from Gibbs sampling (§C.9), with the “combo” prior (not the same prior used in 1) and from
3. HMC, obtained just above, using Stan (same prior and posterior as in 2, but different sampling algorithm—results from 2 and 3 will be the same up to MC sampling error),

to get our version of [Wak13, Fig. 5.11], a display of Bayesian credible intervals for each analysis. Again, up to MCMC error, we expect the Gibbs results to be the same as the HMC results as both methods sample from the same posterior but use different sampling algorithms; this appears to be the case here, with any differences plausibly attributable to MC error, which we can reduce simply by running more iterations in each case of Gibbs or HMC (Stan). We see that the improper prior leads to wider credible intervals due to bringing less information via the prior. Relatedly, note the **shrinkage** toward zero of the posteriors obtained from the informative prior (with mean zero).

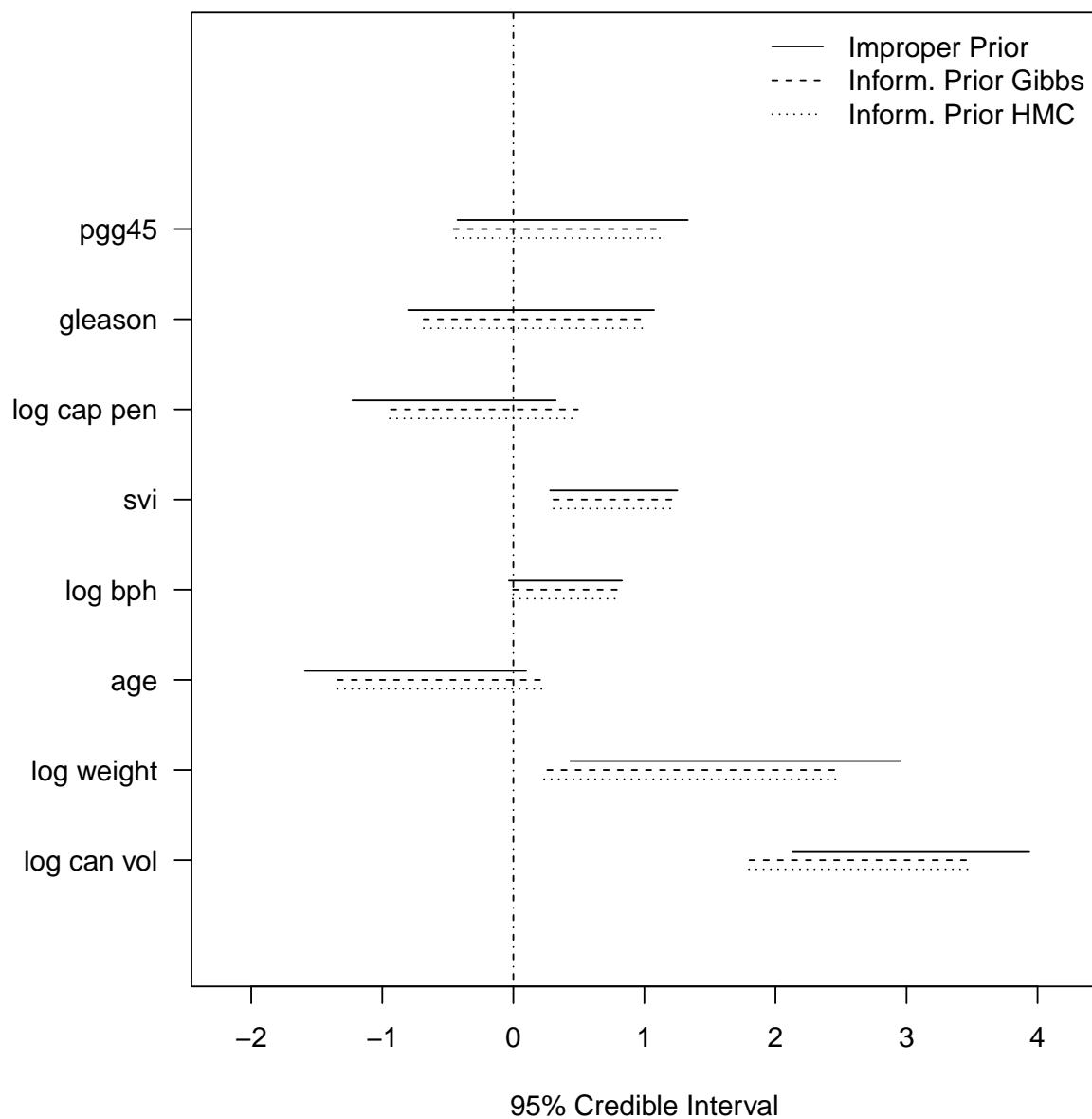
```
> ## Use the previously created objects from our improper prior analysis
> ## (which matched our familiar frequentist results):
> p1lo<- tpost25lb[2:9]
> p1hi<- tpost975ub[2:9]
>
> ## Omit the initial value and first 5000 samples, i.e., omit
> ## ``burn-in'' or ``warm-up'' iterations from Gibbs:
> p2lo<- bayessum$quantiles[-c(1,10),c("2.5%")]
> p2hi<- bayessum$quantiles[-c(1,10),c("97.5%")]
>
> ## From Stan HMC (already omitted first 5000 iterations from each of
> ## three chains):
> names(zprostate1.psum)

[1] "statistics" "quantiles"  "start"       "end"        "thin"
[6] "nchain"

> p3lo<- zprostate1.psum$quantiles[2:9,c("2.5%")]
> p3hi<- zprostate1.psum$quantiles[2:9,c("97.5%")]

> ## Similar to Wakefield's BFRM Fig 5.11:
> par(mar=c(5,7,1,1)+.1)
```

```
> plot(p1lo,p1hi,xlim=c(-2.2,4.2),type="n",xlab="95% Credible Interval",
+       ylab="",axes="F",ylim=c(0,10))
> box()
> axis(1)
> axis(2,at=seq(1,8),labels=c("log can vol","log weight","age","log bph","svi",
+                               "log cap pen","gleason","pgg45"),las=1)
> for (i in 1:8){
+   lines(y=c(i+.1,i+.1),x=c(p1lo[i],p1hi[i]))
+   lines(y=c(i,i),x=c(p2lo[i],p2hi[i]),lty=2)
+   lines(y=c(i-.1,i-.1),x=c(p3lo[i],p3hi[i]),lty=3)
+
+ }
> abline(v=0,lty=4)
> legend("topright",
+         legend=c("Improper Prior","Inform. Prior Gibbs", "Inform. Prior HMC"),
+         lty=1:3,bty="n")
```



```
> detach(package:coda)
> detach(package:rstan)
```

C.12 Other Priors

Generally speaking, for priors different from what we've seen, we will typically not recognize the posterior or full conditionals, in which case we have to appeal to other methods, MCMC again being very popular, e.g., (Metropolis-) Hastings or more specialized algorithms such as HMC as implemented in Stan. See [Wak13, §3.8].

Lecture References

- [Ber85] James O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer, 2nd edition, 1985. ISBN 0387960988.
- [Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [Box76] George E.P Box. Science and statistics. *Journal of the American Statistics Association*, 71(356):791–799, 1976.
- [CB02] George Casella and Roger L. Berger. *Statistical Inference*. Duxbury, Pacific Grove, 2 edition, 2002.
- [Chr02] Ronald Christensen. *Plane answers to complex questions: the theory of linear models*. Springer–Verlag, 3rd edition, 2002.
- [Coh88] Jacob Cohen. *Statistical power analysis for the behavioral sciences*. Routledge, 1988.
- [Efr20] Bradley Efron. Prediction, estimation, and attribution. *Journal of the American Statistical Association*, 115(530):636–655, 2020.
- [Far14] Julian James Faraway. *Linear models with R*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press, 2 edition, 2014.
- [GCS⁺14] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Chapman & Hall/CRC, Boca Raton, 3rd edition, 2014.
- [GS90] Alan E. Gelfand and Adrian F. M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990.

- [Har97] David A. Harville. *Matrix Algebra from a Statiscian's Perspective*. Springer–Verlag, 1997.
- [HTF01] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.
- [KNNL05] Michael H. Kutner, Christopher J. Nachtsheim, John Neter, and William Li. *Applied Linear Statistical Models*. McGraw–Hill/Irwin, New York, 5th edition, 2005.
- [Mur12] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.
- [Mur21] Paul Murrell. *R Graphics*. The R Series. Chapman & Hall/CRC Press, 3rd edition, 2021.
- [Pea09] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition, 2009.
- [PM18] Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic Books, New York, 2018.
- [Rob01] Christian P. Robert. *The Bayesian Choice: From Decision–Theoretic Foundations to Computational Implementation*. Springer, New York, 2001. ISBN 0–387–95231–4.
- [RS13] Fred L. Ramsey and Daniel W. Schafer. *The Statistical Sleuth: A Course in Methods of Data Analysis*. Brooks Cole, Boston, 3rd edition, 2013.
- [Wak13] Jon Wakefield. *Bayesian and Frequentist Regression Methods*. Springer, New York, 2013.