R Module Day 3: Statistics

Drew Allen

http://acropora.bio.mq.edu.au/people/andrew-allen/

- Start a new project in a new directory in R Studio
- Download the files we will be using today into this new directory at the web address above:
  - `binary.csv`
  - `gala.txt`
  - `darwin.txt`
  - `cathedral.csv`
  - `rats.csv`

Topics Covered

- Statistical Distributions
- Summary Statistics
- T-tests
- Regression (simple linear, multiple linear)
- Analysis of Variance
  - One-way ANOVA
  - Two-way ANOVA
  - ANCOVA
- Generalised linear models

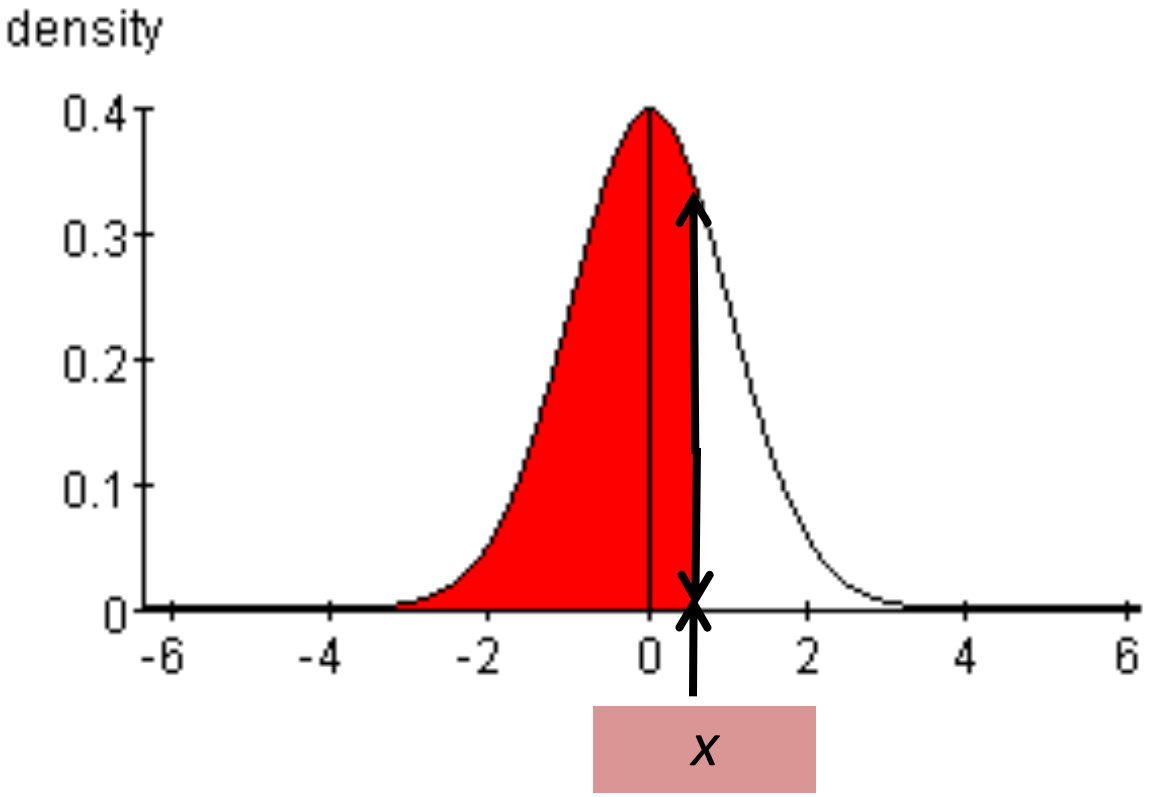
Statistical Distributions

Some Basic Definitions

- **Random Variable** – a variable whose value is not known with certainty, e.g. coin flip
- **Random Variate** – particular outcome of a random variable, e.g. heads
- **Probability** – denotes *relative frequency of occurrence* of particular value, e.g. p(heads) = 0.5
- **Probability distribution** yields the probability of
  - Each value of a random variable (**discrete distribution**)
  - the value of a random falling within a particular interval (**continuous distribution**)

Probability density (i.e. height) at *x*

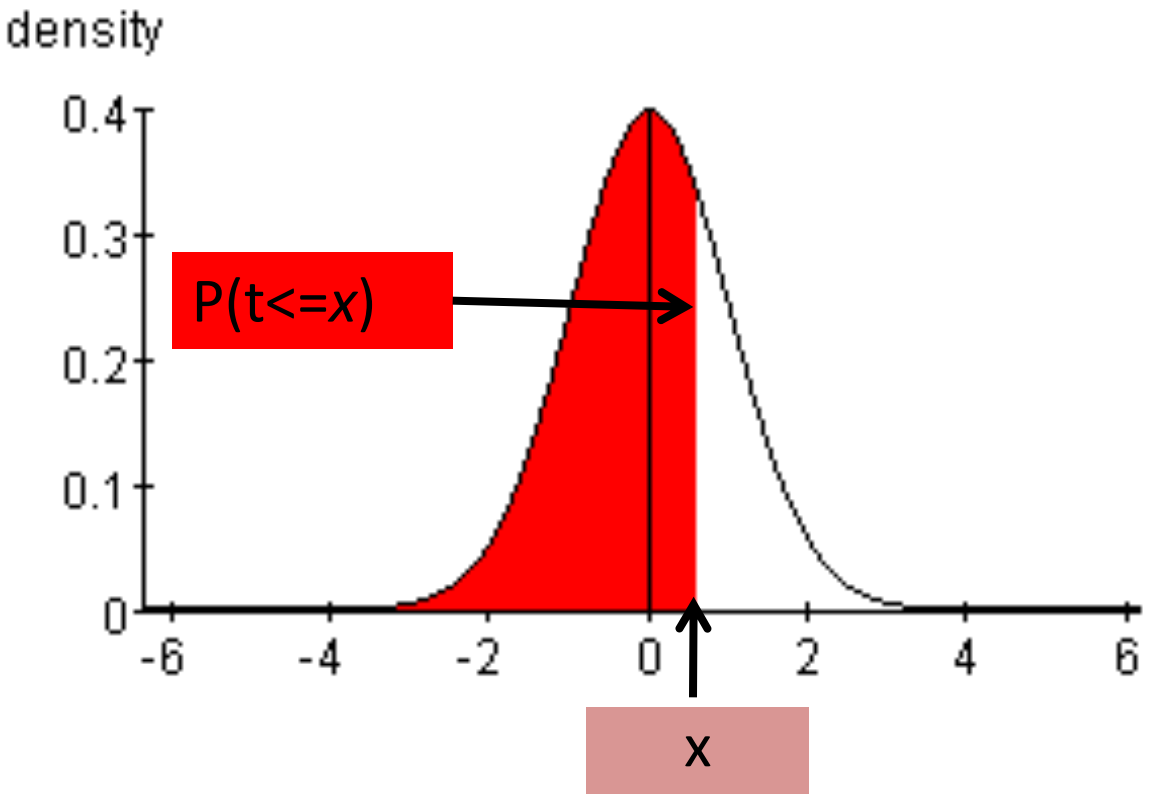
```
dnorm(x,mean=0,sd=1)
```



Probability that variate *t* <= *x*

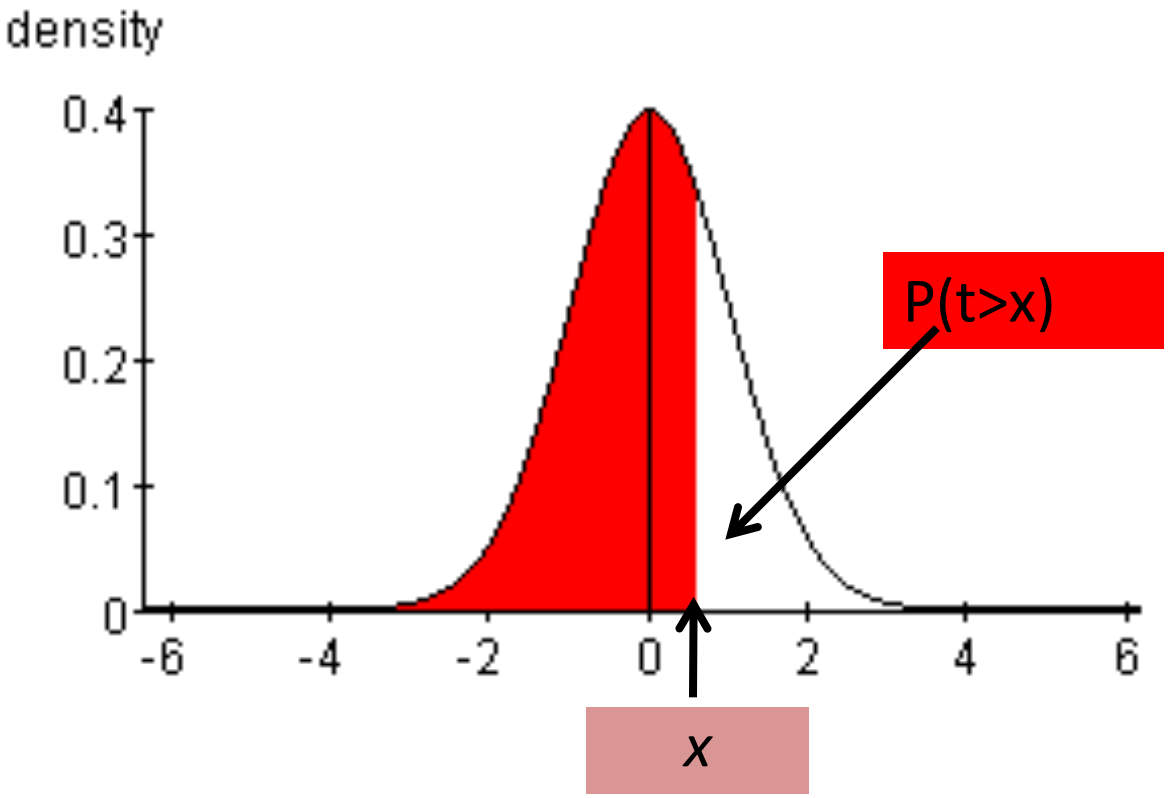
Cumulative Distribution Function, CDF

```
pnorm(x,mean=0,sd=1,lower.tail=TRUE)
```



Probability that variate $t > x$
Complementary CDF

```
pnorm(x,mean=0,sd=1,lower.tail=FALSE)
```

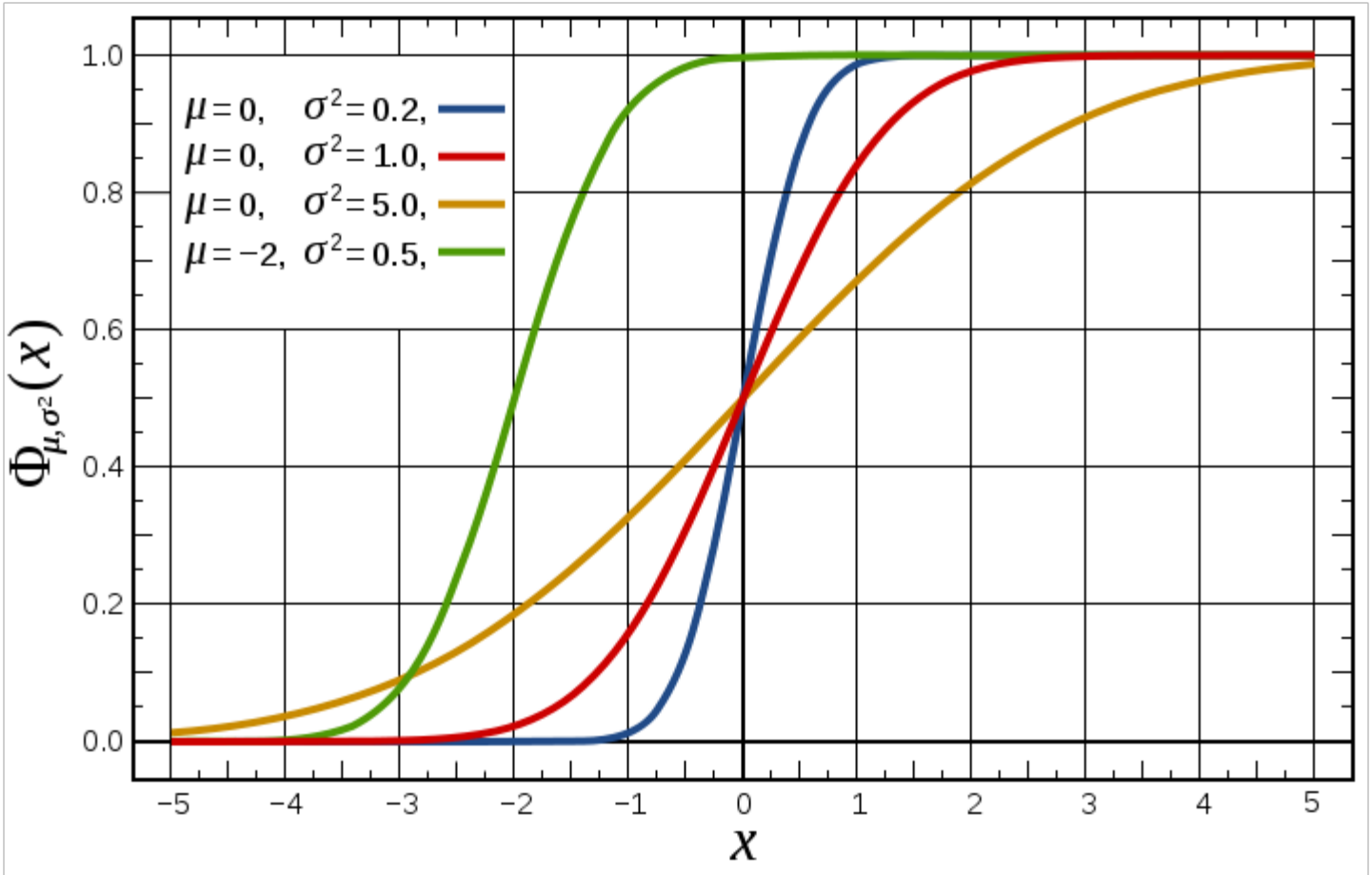


Question: What is this sum?

- **pnorm(x,mean=0,sd=1,lower.tail=TRUE)+**
  pnorm(x,mean=0,sd=1,lower.tail=FALSE)

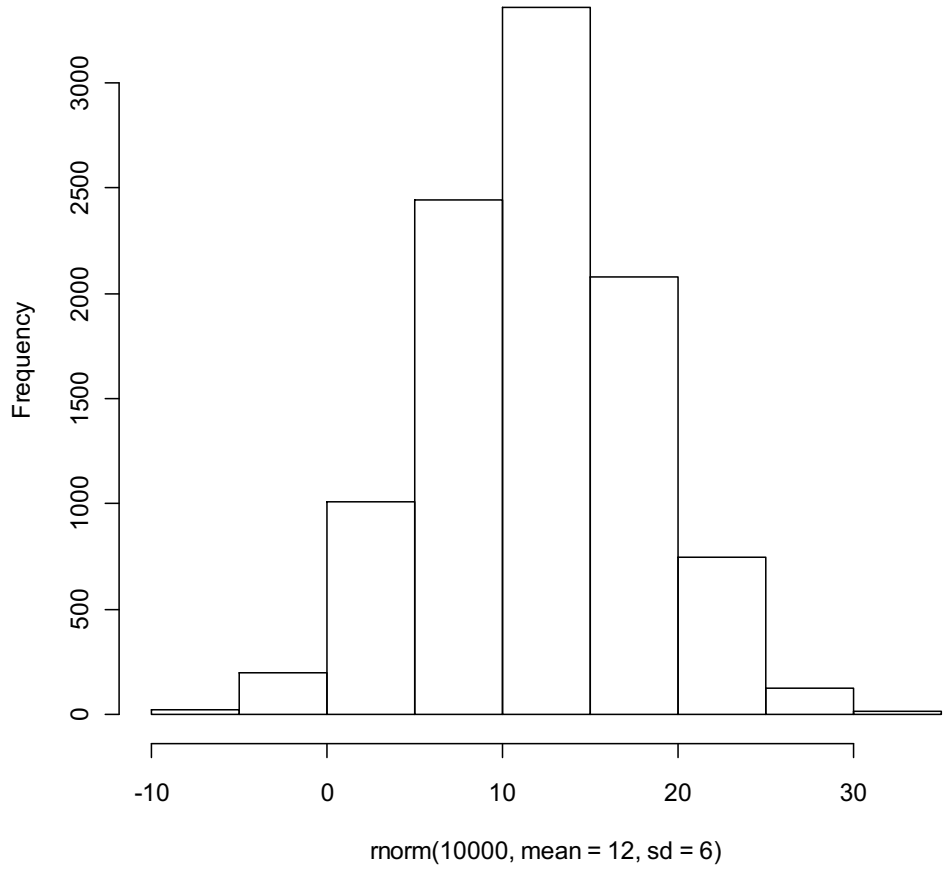
At what value of x is P(*t*<=*x*) = 0.4?

```
qnorm(0.4,mean=-2,sd=sqrt(0.5))
```



Sampling from a distribution

```
hist(rnorm(1000,mean=12,sd=6))
```



Functions have required and optional arguments

- Works fine (no required arguments)
  - `q()#quits R`
- Doesn't work:
  - `rnorm()#missing argument for n, which has no default`
- Does work (caution: computer assigns values for some arguments!)
  - `rnorm(100) #takes default arguments`
- Does work (all arguments specified by user)
  - `rnorm(100,mean=1,sd=4)`
  - `rnorm(mean=1,sd=4,n=100)`

Exercise 1:
Using R as a Statistics Table

- Generate a sample of 1000 variates from a normal distribution of mean 10 and standard deviation 5 using `rnorm`
- For this sample, calculate what fraction of the points take values <5 (hint: use `length`)
- Using `pnorm`, calculate the theoretically predicted fraction of points that should take values < 5

Exercise 1: Answer

- `x <- rnorm(1000,mean=10,sd=5)`
- `length(x[x<5])/length(x)`
- `pnorm(5,mean=10,sd=5)`

Built-in Probability Distributions:
for the list, type `?Distributions`

Continuous distributions

- Normal (`dnorm`)
- T (`dt`)
- Chi-squared (`dchisq`)
- F (`df`)
- Exponential (`dexp`)
- Uniform (`dunif`)
- Beta (`dbeta`)
- Cauchy (`dcauchy`)
- Logistic (`dlogis`)
- Lognormal (`dlnorm`)
- Gamma (`dgamma`)
- Weibull (`dweibull`)

Discrete distributions

- Binomial (`dbinom`)
- Poisson (`dpois`)
- Geometric (`dgeom`)
- Hypergeometric (`dhyper`)
- Negative binomial (`dnbinom`)

Other Distributions Use Similar Syntax

NORMAL DISTRIBUTION

- `dnorm(x, mean = 0, sd = 1, log = FALSE)`
- `pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)`
- `qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)`
- `rnorm(n, mean = 0, sd = 1)`

UNIFORM DISTRIBUTION

- `dunif(x, min=0, max=1, log = FALSE)`
- `punif(q, min=0, max=1, lower.tail = TRUE, log.p = FALSE)`
- `qunif(p, min=0, max=1, lower.tail = TRUE, log.p = FALSE)`
- `runif(n, min=0, max=1)`

Exercises 2 and 3:
Using R as a Statistics Table

- What is the probability that a random variate from a gamma distribution with a shape parameter = 3 and scale parameter = 1 is > 0.68? [use `pgamma`]
- What is the probability that a random variate from an exponential distribution with rate = 0.05 lies between 1 and 10? [use `pexp`]

Exercises 2 and 3: Answers

- 1- pgamma(0.68,shape=3, scale = 1)

- pexp(10,rate=0.05) - pexp(1,rate=0.05)

Exercise 4:
Using R as a Statistics Table

- What is the probability that a random sample of 15 people has 2 people with the same birthday? [Hint: ?pbirthday]

- What is the probability that a random sample of 25 martians includes 2 martians with the same birthday? [Hint: a year on Mars is 687 days]

Exercise 4 Answer:

R functions arguments can be matched positionally or by name

- pbirthday(n = 15, classes = 365, coincident = 2)
- pbirthday(15, 365, 2)
- pbirthday(classes = 365, 15, 2)
- pbirthday(15, coincident = 2)
- pbirthday(25, coincident = 2) #wrong answer for martians
- pbirthday(25,687,2) #right answer for martians

Statistical distributions provide a means to perform simulations

- `#using r for simulation of 1D random walker`
- `steps<-rnorm(n=10000,mean=0,sd=1)`
- `distance.from.origin <- cumsum(steps)`
- `plot(distance.from.origin,type='l')`

Use of `set.seed()` for reproducible random results

- `#using r for simulation of 1D random walker`
- `set.seed(1)`
- `steps<-rnorm(n=10000,mean=0,sd=1)`
- `distance.from.origin <- cumsum(steps)`
- `plot(distance.from.origin,type='l')`

Summary Statistics

Some Functions for Calculating Summary Statistics

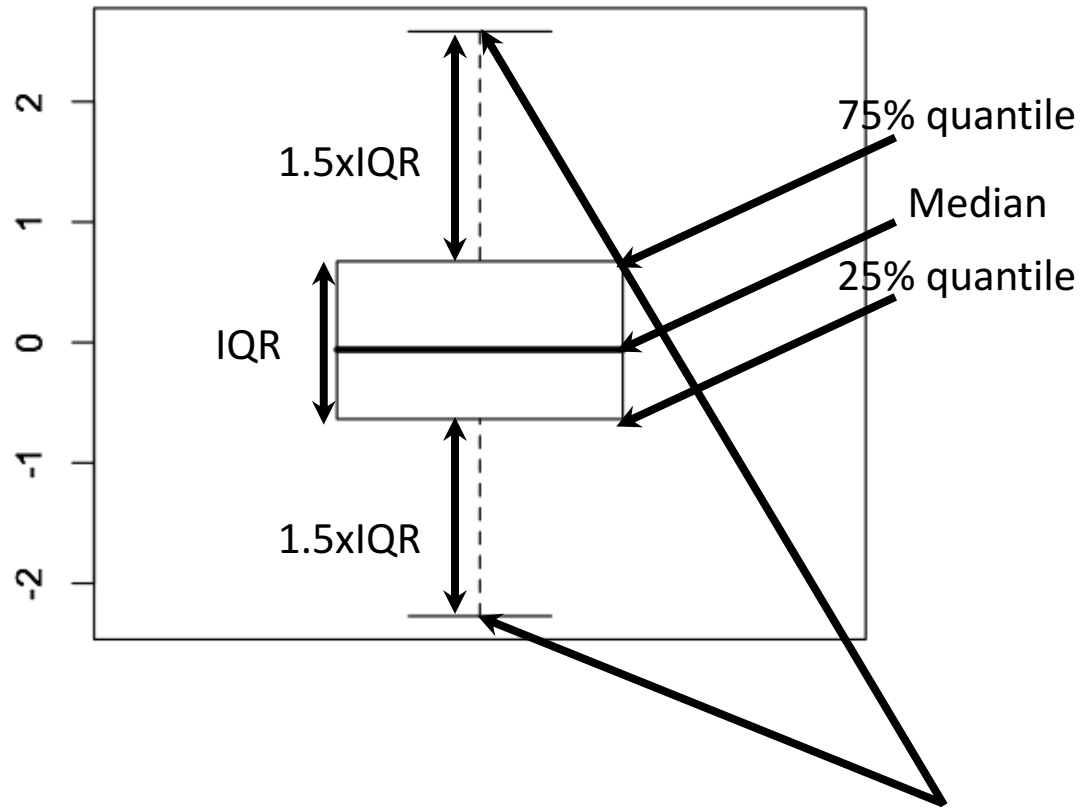
- Minimum: `min()`
- Maximum: `max()`
- Range (Minimum and Maximum): `range()`
- Mean: `mean()`
- Median: `median()`
- Quantiles: `quantile()`
- Interquartile range: `IQR()`
- Variance: `var()`
- Standard Deviation: `sd()`
- Summary: `summary()`
- Stem & Leaf Plot: `stem()`

- Boxplot: `boxplot()`
- QQ Plot: `qqnorm()`, `qqline()`

Functions for Calculating Summary Statistics

```
>x<-rnorm(100)

>boxplot(x)
```



IQR= 75% quantile -25% quantile= Inter Quantile Range

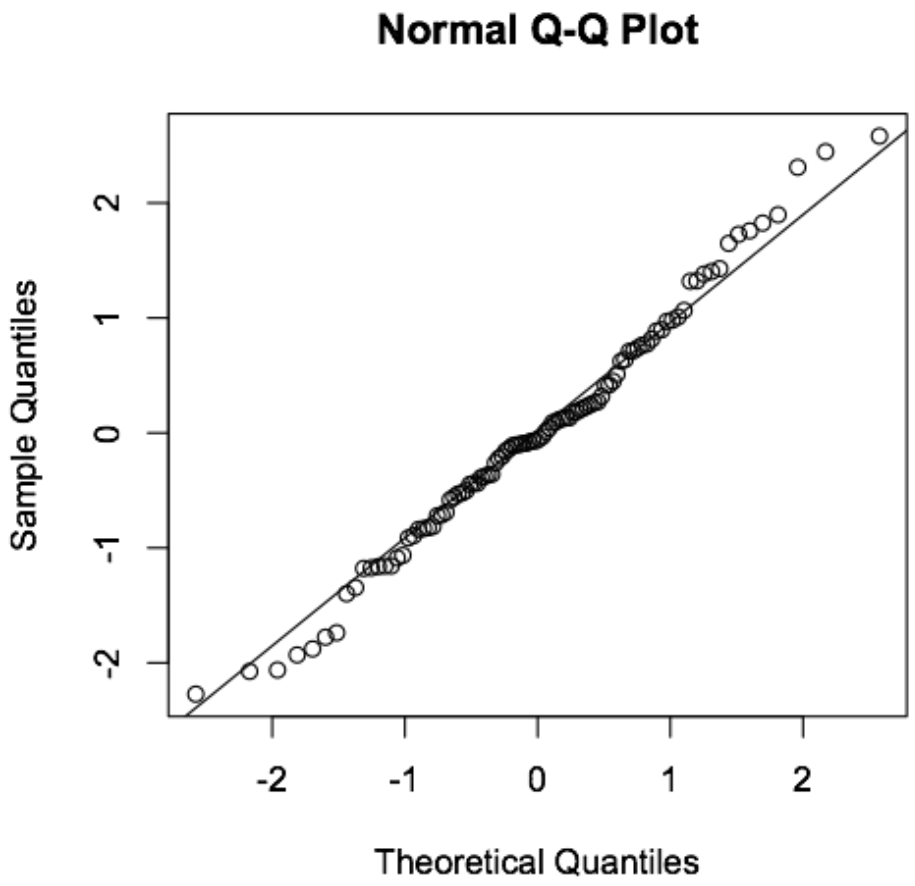
Everything above or below are considered outliers

QQ Plot

- Many statistical methods make some assumption about the distribution of the data (e.g. Normal)
- The quantile-quantile plot provides a way to visually verify such assumptions
- The QQ-plot shows the theoretical quantiles versus the empirical quantiles. If the distribution assumed (theoretical one) is indeed the correct one, we should observe a straight line.

QQ Plot

- `x<-rnorm(100)`
- `qqnorm(x)`
- `qqline(x)`



Functions for Calculating Summary Statistics

- Two functions are extremely useful for calculating summary statistics for subsets of data:
  - `apply()` (calculates function on a column-by – column or row-by-row basis)
  - `tapply()` (groups data in one column based on values in another column)

T test

What does Student's t distribution have to do with Guinness Stout?

VOLUME VI MARCH, 1908 No. 1

BIOMETRIKA.

THE PROBABLE ERROR OF A MEAN.

BY STUDENT.

Introduction.

ANY experiment may be regarded as forming an individual of a "population"

T distribution

- The t distribution was introduced by William Gosset, a chemist working for Guinness brewery in Ireland
- He published his work under the pen name “Student” because Guinness regarded the fact that they were using statistics to help with brewing to be a trade secret

T test Example:
Darwin's Plant Growth Data

- Data are from Darwin's study of cross- and self-fertilization.
- Pairs of seedlings of the same age, one produced by cross-fertilization and the other by self-fertilization, were grown together so that the members of each pair were reared under nearly identical conditions.
- The data are the final heights of each plant after a fixed period of time, in inches.
- Darwin consulted the famous 19th century statistician Francis Galton about the analysis of these data

Exercise 5:
Darwin's Plant Growth Data

- Import `darwin.txt`
- Conduct a paired T test using the function `t.test()`
  - Type `?t.test` for some help
- Answer the following questions:
  - What is the mean difference, *m*, between the treatments?
  - What is the standard deviation, *s*, of the paired differences?
  - According to the t test, is the difference significant at the P = 0.05 level for the two-tailed test?
  - According to the non-parametric analogue of the t test (Mann-Whitney U), is the difference significant at the P = 0.05 level for the two-tailed test? **[Use `wilcox.test`]**

Exercise 5 Answers

- `darwin <- read.table('darwin.txt',header=TRUE)`
- `m<-mean(darwin$crossfertilized-darwin$selffertilized)`
- `s<-sd(darwin$crossfertilized-darwin$selffertilized)`
- `t.test(darwin$crossfertilized,darwin$selffertilized,paired=TRUE)`
- `wilcox.test(darwin$crossfertilized,darwin$selffertilized,paired=TRUE)`

More on T tests

- `#one-sample t test`
- `t.test(darwin$crossfertilized - darwin$selffertilized,mu=0)`

- `#Welch two-sample t test`
- `t.test(darwin$crossfertilized,darwin$selffertilized)`

- `#Student two-sample t test`
- `t.test(darwin$crossfertilized,darwin$selffertilized,var.equal=TRUE)`

Mann-Whitney U Test

- This technique is non-parametric , meaning that it does not rely on assumptions that the data are drawn from a particular probability distribution.
- Non-parametric methods are particularly suited to data that are not normally distributed.
- Assumptions Mann-Whitney U Test include:
    - random samples from populations
    - independence within samples and mutual independence between samples
    - measurement scale is at least ordinal

Power Analysis

- **A very important part of planning research**

- **Power** is the conditional probability of rejecting the null hypothesis given that it is really false

- 1- Power = Type II error

Packages Allow You To Increase the Functionality of R

R has lots of statistical capabilities

- Full list of packages:
  - http://cran.r-project.org/web/packages/available_packages_by_name.html
- Task views are helpful:
  - http://cran.r-project.org/web/views/

Please add the following packages

- Please add the following packages and make them available
  - **`pwr`**: for performing power analysis

- `install.packages('pwr')`
- `library(pwr)`

Exercise 6:
Darwin's Plant Growth Data

- Install the library `pwr`
- Calculate the estimated effect size as $d = m / s$ for the `darwin.txt` data
- In the command window, learn how to conduct a power analysis using `?pwr.t.test`
- Using this function, calculate the statistical power of the test that Darwin conducted
- Now use this function to determine how large a sample size would be required to reject the null hypothesis at a significance level of 0.05 with 80% power

Exercise 6 Answer

- `m<-mean(darwin$crossfertilized-darwin$selffertilized)`
- `s<-sd(darwin$crossfertilized-darwin$selffertilized)`

- `pwr.t.test(n=16,d=m/s,sig.level=0.05,type='paired')`

- `pwr.t.test(d=m/s,sig.level=0.05,power=0.8,type='paired')`

Linear Regression

Linear Regression

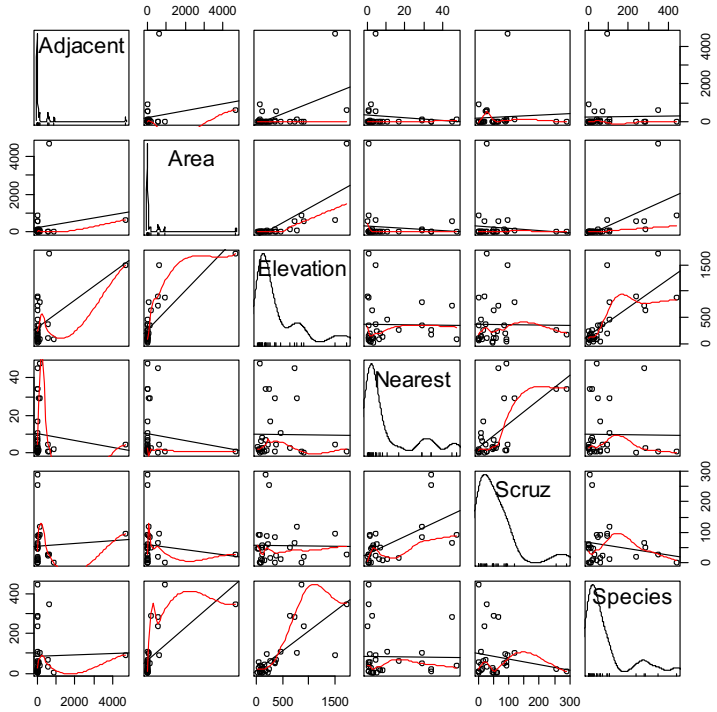
- Use `gala <- read.table(...,header=TRUE,row.names=1)` to import the dataset `gala`
- View the dataset using `head(gala)`

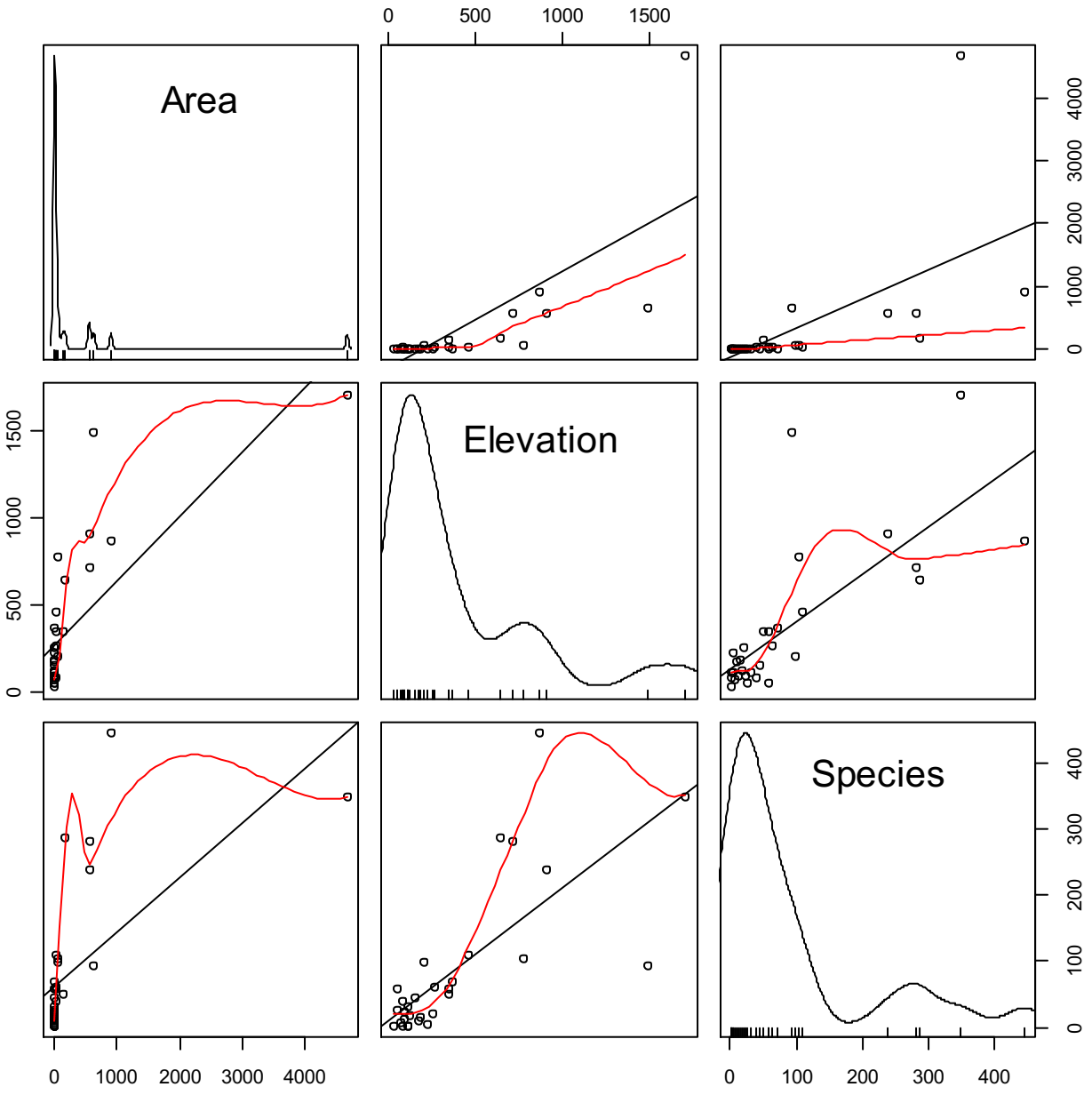
gala

- Source
  - M. P. Johnson and P. H. Raven (1973) "Species number and endemism: The Galapagos Archipelago revisited" Science, 179, 893-895
- Variables
  - **Species** the number of plant species found on the island
  - **Endemics** the number of endemic species
  - **Area** the area of the island (km^2)
  - **Elevation** the highest elevation of the island (m)
  - **Nearest** the distance from the nearest island (km)
  - **Scruz** the distance from Santa Cruz island (km)
  - **Adjacent** the area of the adjacent island (square km)

Investigate Distributions of Variables and Their Relationships

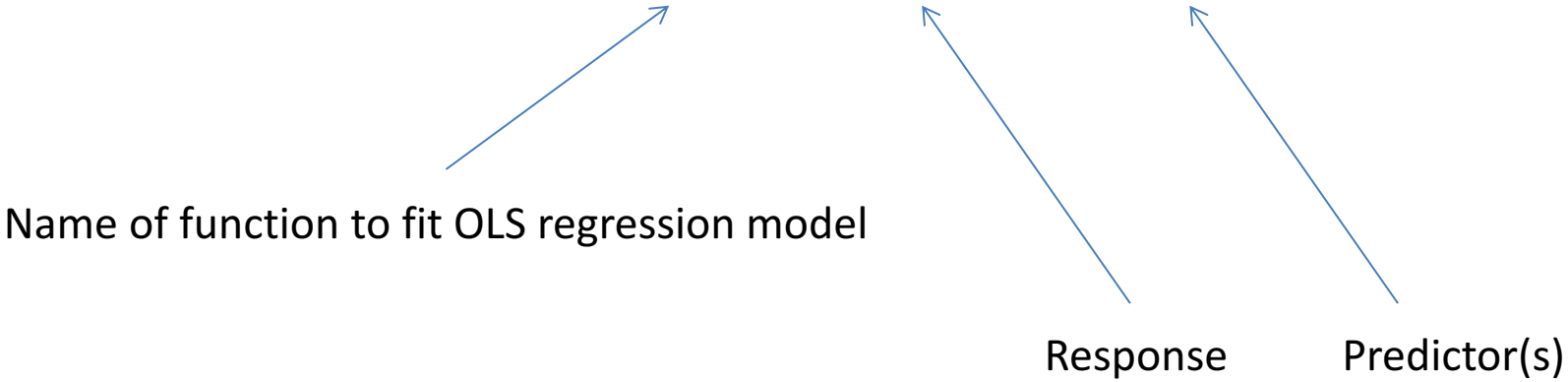
- Generate a plot similar to the one below by typing `plot(gala)`




Area
Elevation
Species
0
500
1000
1500
0
1000
2000
3000
4000
0
100
200
300
400

Ignore these issues and fit a linear model

- Now fit a linear regression model by typing:
  - `gala.model<-lm(Species~Area, data=gala)`



- Let's look at the attributes of this object:
  - `str(gala.model)`

Extractor functions allow you to get information on `lm` objects

- `coef(gala.model)`
- `residuals(gala.model)`
- `fitted.values(gala.model)`
- `cooks.distance(gala.model)`
- `summary(gala.model)`
- `anova(gala.model)`

Assumptions of Linear Regression

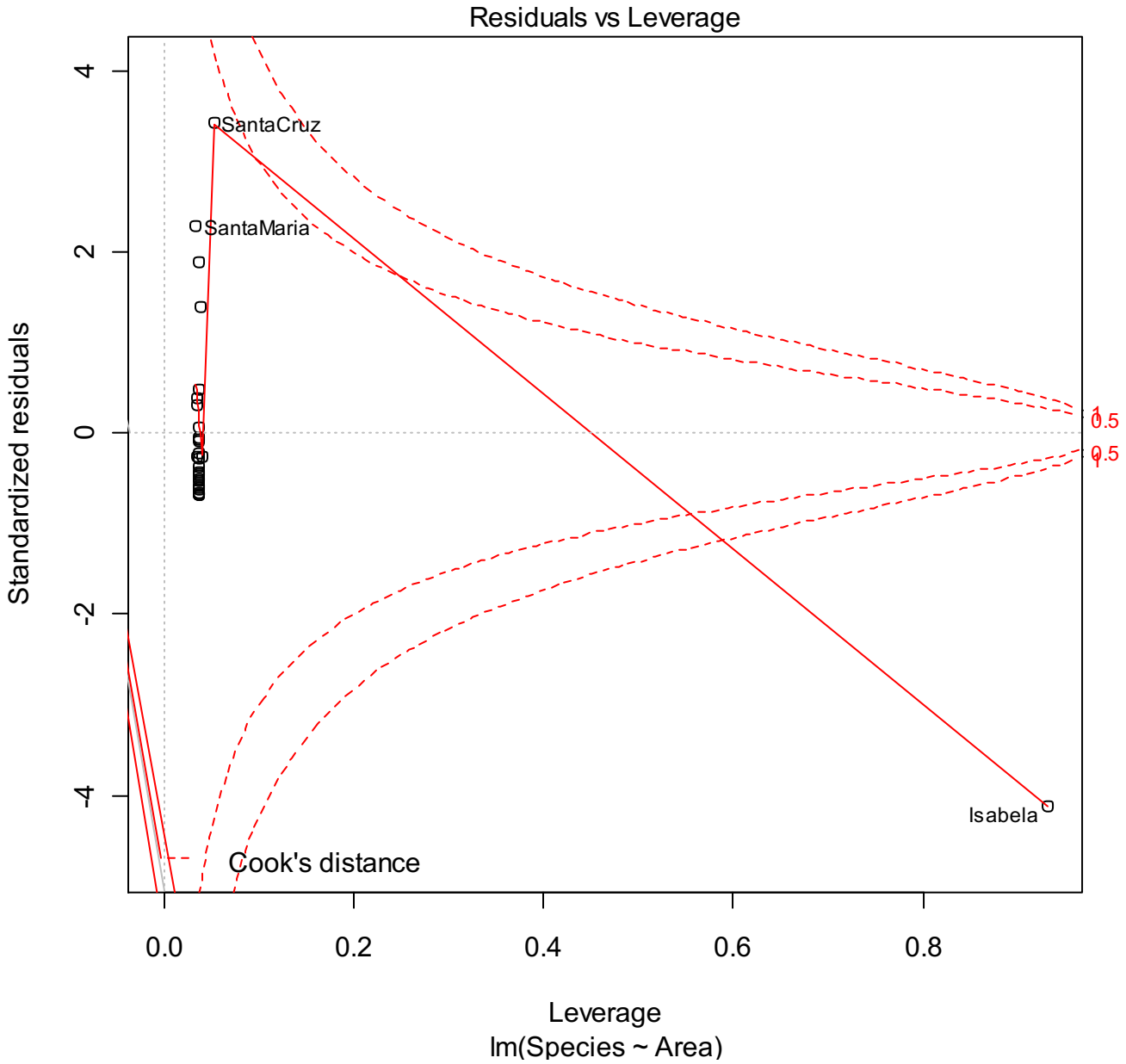
- **Linearity** of the relationship between dependent and independent variables
- **Independence** of the errors (no serial correlation)
- **homoscedasticity** (constant variance) of the errors
- **normality** of the error distribution

Let's evaluate these assumptions

- To evaluate assumptions type:
  - `plot(gala.model)`
- Theory:
  - Leverage is a measure of how far an independent variable deviates from its mean
  - Cook's distance
    - measures the influence of an observation on the overall model:

$$D_i = \frac{\sum_{j=1}^{n} (\hat{Y}_j - \hat{Y}_{j(i)})^2}{p \ \mathrm{MSE}}.$$

    - Y_j is the prediction from the full regression model for observation *j*
    - $Y_{j\,(i)}$ is the prediction for observation *j* from a refitted regression model in which observation *i* has been omitted
  - Frequently proposed rules of thumb include focusing on points with distances $D_i > 1$ or $> 4/n$


Residuals vs Leverage
Standardized residuals
Leverage
lm(Species ~ Area)
SantaCruz
SantaMaria
Isabela
Cook's distance
0.5
0.5
4
2
0
-2
-4
0.0
0.2
0.4
0.6
0.8

Exercise 7:

Independent analysis of `gala` data

- Transform species and area using the log10 transformation, e.g.
- Refit the linear model using the log transformed data and assess whether model assumptions are upheld
- Plot the data and model together using the functions `plot()` and `abline()`
- Inspect the coefficients using `summary()`

Exercise 7: Answer

- gala$log.species<-log10(gala$Species)
- gala$log.area<-log10(gala$Area)
- gala.model<-lm(log.species~log.area, data=gala)
- plot(log.species~log.area,gala)
- abline(gala.model)

Fit of simple linear regression model

- `summary(gala.model)`

```
Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.26106    0.06822  18.484  < 2e-16 ***
log.area     0.38860    0.04160   9.342 4.23e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3406 on 28 degrees of freedom
Multiple R-squared: 0.7571,   Adjusted R-squared: 0.7484
F-statistic: 87.27 on 1 and 28 DF,  p-value: 4.23e-10
```

- 95% confidence interval for fitted slope:
  - lower CI: `0.38860 + qt(.025,28)* 0.04160`
  - Upper CI: `0.38860 - qt(.025,28)* 0.04160`
  - `confint(gala.model)`

Multiple linear regression

- Extending analyses to multiple linear regression is straightforward using `lm()`:
  - lm(y~x1 + x2,data)
- Notation used for formulas (VERY general, applies to many statistical procedures in R):
  - Intercept only
    - `lm(y~1,data)`
  - Force-fit y versus x1 relationship through origin
    - `lm(y~x1-1,data)`
  - Include all variables in data.frame:
    - `lm(y~.,data)`
  - Include all variables in data.frame but x7:
    - `lm(y~.- x7,data)`
  - Include x1, x2 and their interactions:
    - `lm(y~x1*x2,data)`
    - `lm(y~x1+x2+x1:x2)`
    - `Lm(y~x1*x2*x3 - x1:x2:x3) #drop 3-way interaction`

Exercise 8

Formally test for effects of `log.elevation` after accounting for `log.area`

- Fit a new model that includes both `log.elevation` and `log.area`
- Null hypothesis: after account for the effects of area, elevation is not significant
- How do we test this null hypothesis?

- R knows what to do. Just type:
  - `anova(lm1,lm2)`

Exercise 8 Answer

- `gala$log.elevation <- log10(gala$Elevation)`
- `gala.model2 <- lm(log.species~log.area+log.elevation,data=gala)`
- `anova(gala.model,gala.model2)`

Automated Model Selection

- Several methods available:
  - Best subset selection
  - Stepwise selection

- Fit using multiple criteria:
  - Statistical significance [`logLik(lm1)-logLik(lm2)`]
  - AIC [`AIC(lm1) - AIC(lm2)`]
- Key issue: need to first specify a full model

- Controversial among statisticians due to multiple comparisons problem, but still useful for exploration

R Code for BE using `step()`

- Use R function `step`
- Need to define an *initial model* (the full model in this case, as produced by the R function lm) and a *scope* (a formula defining the full model)
- `ffa.lm <- lm(ffa~., data=ffa.df)`
- `step(ffa.lm,direction="backward")`

Forward Selection (FS) using `step()`

- Start with a null model
- Fit all one-variable models in turn. Pick the model with the best (i.e. lowest) AIC
- Then, fit all two variable models that contain the variable selected in 2. Pick the one for which the added variable gives the best AIC
- Continue in this way until adding further variables does not reduce the AIC

R Code for FS using `step()`

- Use R function `step`
- As before, we need to define an *initial model* (the null model in this case and a *scope* (a formula defining the full model)
- **`# R code: first make null model:`**
- **`ffa.lm = lm(ffa~., data=ffa.df)`**
- **`null.lm = lm(ffa~1, data=ffa.df)# then do FS`**
- **`step(null.lm, scope=formula(ffa.lm),`**
- **`direction="forward")`**

R Code Output (1 of 2)

```
> step(null.lm, scope=formula(ffa.lm),
direction="forward")
Start:  AIC=-49.16
ffa ~ 1

           Df Sum of Sq     RSS     AIC
+ weight    1   0.63906 0.91007 -57.799
+ age       1   0.20503 1.34410 -50.000
<none>                  1.54913 -49.161
+ skinfold  1   0.00145 1.54768 -47.179
```

Starts with constant term only

Results of all possible 1 (& 0) variable models. Pick weight (smallest AIC)

R Code Output (2 of 2)

```
Step:  AIC=-57.8
ffa ~ weight

           Df Sum of Sq     RSS     AIC
+ age       1  0.115900 0.79417 -58.524
<none>                  0.91007 -57.799
+ skinfold  1  0.007778 0.90230 -55.971

Step:  AIC= -58.52
 ffa ~ weight + age

           Df Sum of Sq     RSS     AIC
<none>                   0.794 -58.524
+ skinfold  1     0.003   0.791 -56.601
```

Exercise 9:

Choosing the best predictor of richness

- Using BE and function `step()`, determine the "best" model of species richness using the following potential predictors:
  - log.area
  - log.elevation
  - log.nearest
  - log.scruz [note: use log10(x+1) transform]
  - log.adjacent
- Recall:
  - `y.lm <- lm(y~., data=data)`
  - `step(y.lm,direction='backward')`

Exercise 9 Answer

- gala$log.nearest <- log10(gala$Nearest)
- gala$log.scruz <- log10(gala$Scruz+1)
- gala$log.adjacent <- log10(gala$Adjacent)

- gala.full <- lm(log.species~log.area+log.elevation+log.nearest+log.scruz +log.adjacent,gala)
- gala.step <- step(gala.full,direction='backward')

ANOVA and ANCOVA

Factor Variable Type

- `ssize <- sample(0:2,40,replace=TRUE)`
- `ssize`
- `is.factor(ssize)`
- `ssize.f <- factor(ssize,labels=c('s','m', 'l'))`
- `is.factor(ssize.f)`
- `is.ordered(ssize.f)`
- `ssize.f <- factor(ssize,labels=c('s','m', 'l'),ordered=TRUE)`
- `is.ordered(ssize.f)`
- `ssize.f[41] <- 'x'`
- `levels(ssize.f) <- c('s','m','l','x')`
- `ssize.f[41] <- 'x'`

http://www.ats.ucla.edu/stat/r/modules/factor_variables.htm

One-way ANOVA using `mtcars`

- ?mtcars
- summary(mtcars)
- str(mtcars)

Exercise 10: One-way ANOVA using `mtcars`

- Using `factor()`, create a new variable (`cyl.f`) in the data.frame `mtcars` that treats the number of cylinders (`cyl`) as a factor variable
- Using `lm()`, fit an lm model that predicts mileage (`mpg`) based on the number of cylinders (`cyl.f`). Call it `lm1`.
- Using `lm()`, fit a regression model that predicts mileage based on engine horsepower (`hp`). Call it `lm2`.
- Compare the two models using `AIC()`
- Which is “better”

Exercise 10 Answer

- mtcars$cyl.f <- factor(mtcars$cyl)

- lm1 <- lm(mpg ~ cyl.f,mtcars)
- summary(lm1) #estimates of coefficients
- anova(lm1) #overall effects of cyl.f

- lm2 <- lm(mpg ~ hp,mtcars)
- summary(lm2)
- anova(lm2) #overall effects

- AIC(lm1,lm2)

Changing reference level in ANOVA

- contrasts(mtcars$cyl.f)
- mtcars$cyl.fa <- relevel(mtcars$cyl.f,ref='8')
- contrasts(mtcars$cyl.fa)
- lm1a <- lm(mpg ~ cyl.fa,mtcars)
- summary(lm1a)

http://www.ats.ucla.edu/stat/r/library/contrast_coding.htm

Other Stuff....

- #formal analysis of variance
- anova(lm1)

- #post hoc test
- TukeyHSD(aov(lm1))
- plot(mtcars$mpg~mtcars$cyl.f)

- ?aov() #alternative way of fitting anova models, allows for error strata

Two-way ANOVA using Rat data

- rats <- read.csv('rats.csv')
- plot(time ~ treat + poison, data=rats)
- interaction.plot(rats$treat,rats$poison,rats$time)
- interaction.plot(rats$poison,rats$treat,rats$time)

Rat Data

- g <- lm(time ~ poison*treat, rats)
- anova(g)
- qqnorm(g$res)
- qqline(g$res)

Exercise 11

- Transform the rat response to `1/time`
- Refit the model using `lm`
- Undertake diagnostic residual plots to assess deviations from normality
- Assess the significance of the interaction term by calling the function `anova`

- Do treatments vary in effectiveness?
- Do poisons vary in toxicity?
- Does the success of treatment vary by poison?

Exercise 11 Answers

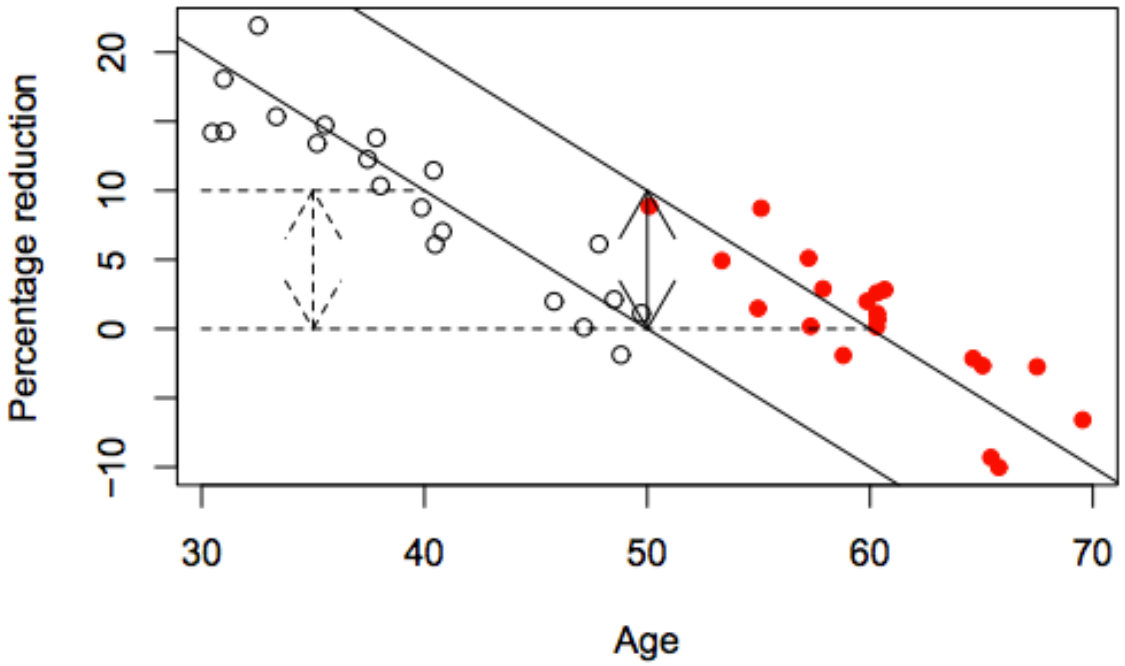
- g <- lm(1/time ~ poison*treat,rats)

- plot(g$fitted,g$res,xlab="Fitted",ylab="Residuals",main="Reciprocal response")
- qqnorm(g$res)
- qqline(g$res)

- anova(g)

ANCOVA

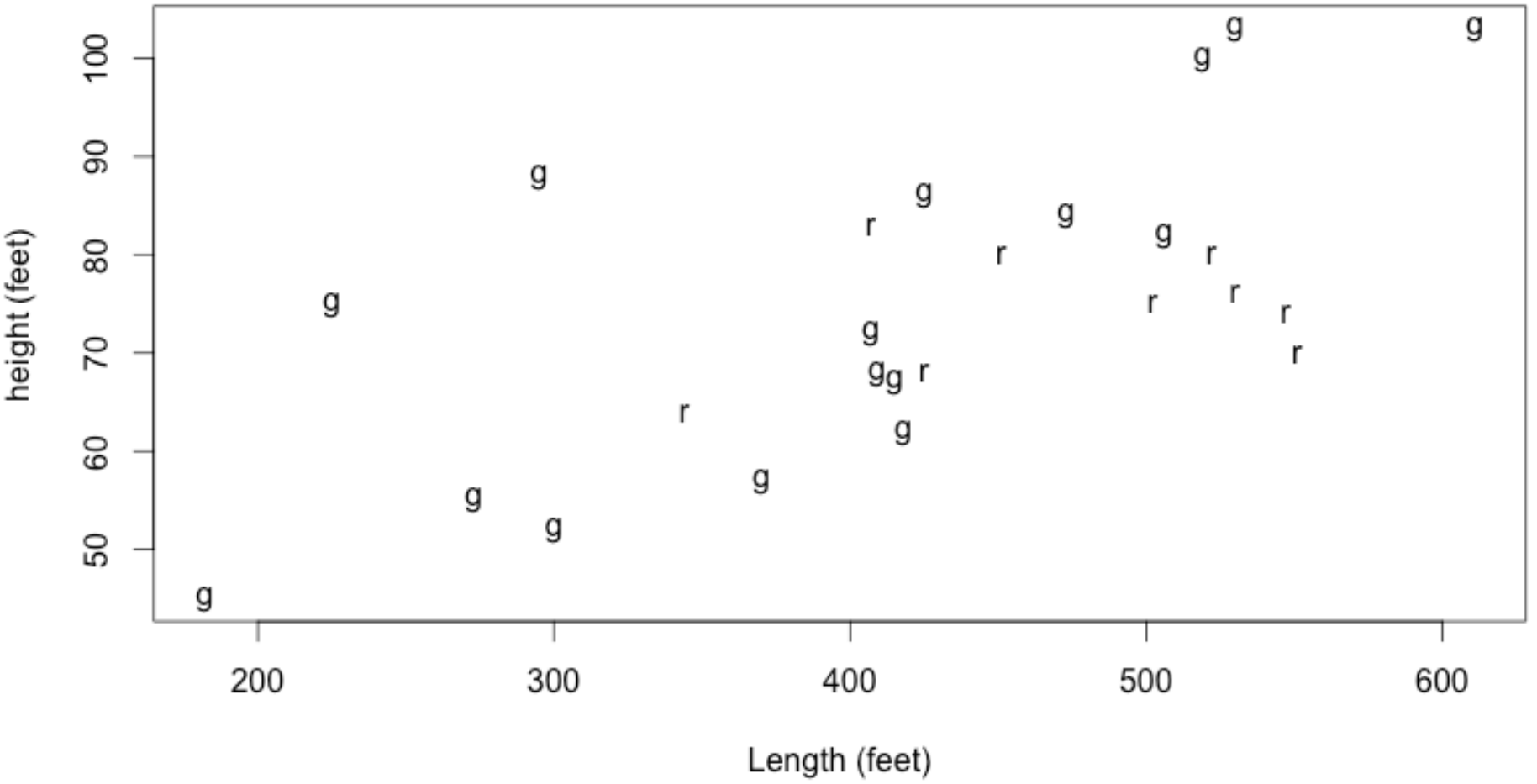
- Refers to regression problems where there is a mixture of quantitative and qualitative predictors



Cathedral Dataset

- cathedral <- read.csv('cathedral.csv')
- names(cathedral)[2:3] <- c('height','length')
- plot(cathedral$length,cathedral$height,type="n", xlab=”length (feet)",ylab="height (feet)")
- text(cathedral$length,cathedral$height,as.character(cathedral$style))

Cathedral Dataset



Exercise 12

- Import the dataset `cathedral`
- Perform a homogeneity of slopes test by fitting a model of the form `lm(height~length+style+length:style,data=cathedral)` and evaluating significance of the `length:style` term using the function `summary`
- If the slope difference is not significant, refit the model assuming a constant slope for both groups. Do the cathedral types differ in height after controlling for length?
- Harder: plot the final fitted model

Exercise 12 Answers

- summary(lm(height ~ length + style:length, cathedral))
- summary(lm(height~length + style ,cathedral))
- plot(cathedral$length,cathedral$height,type="n",xlab="length (feet)",ylab="height (feet)")
- text(cathedral$length,cathedral$height,as.character(cathedral$style))
- abline(34.96916,0.10058)
- abline(34.96916-8.34535,0.10058 ,lty=2)
- legend('topleft',legend=c('Gothic','Romanesque'),lty=c(1,2))

Further Information on ANOVA

- http://goanna.cs.rmit.edu.au/~fscholer/anova.php

- Provides details on how to partition variance, particularly with unbalanced designs

- My recommendation: if your design is unbalanced, and you have two (or more factors), consider using `Anova()` function in `car` package

GLM

Many response variables are inherently non-normal

- Counts (Integers >= 0; e.g. # of chicks)
- Non-negative continuous variables (>=0; e.g. times between foraging bouts)
- Proportions (0 <= P <= 1; e.g. proportion protein in the diet)
- Binary (integer 0/1 for failure/success; e.g. prey capture during predation event; presence-absence of species)

Modeling counts

- **Poisson regression** – simplest method; there are a number of extensions useful for count models (e.g. quasi-poisson)
- **Negative binomial regression** – for over-dispersed count data, meaning that the conditional variance exceeds the conditional mean

http://cran.r-project.org/web/packages/pscl/vignettes/countreg.pdf

Modeling non-negative continuous variables

- **Exponential regression** – assumes conditional distribution of response variable is exponentially distributed
- **Gamma regression** – assumes conditional distribution of response variable is gamma distributed

Modeling proportions

- **Beta regression** – Assumes conditional distribution of response variable is beta distributed

- Unlike the other types of regression, beta regression can't be conducted using glm()

http://cran.r-project.org/web/packages/betareg/vignettes/betareg.pdf

Modeling binary data

- **Logistic Regression** – standard method, involves modeling binary data using the logit link function

- **Probit Regression** – another frequently used method, involves modeling binary data using the probit link function

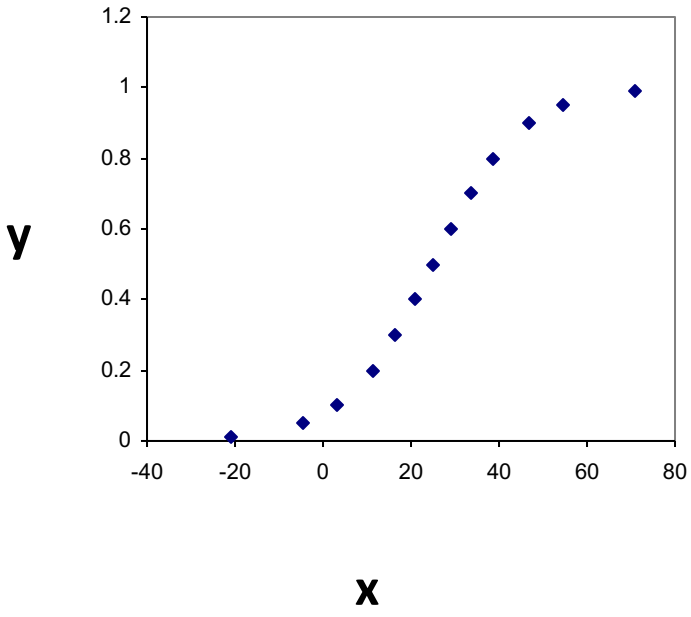
All of these different types of regression are GLM

- Conditional distributions differ:
  - Poisson regression
  - Negative binomial regression
  - Logistic regression
  - Exponential Regression
- Only link functions differ (both assume binomial distribution)
  - Logistic regression
  - Probit regression

Logistic regression

- Old way: arcsine transformation proportion and try OLS regression

- New (better) way: use logit (or probit) link with binomial errors



Source: www.zoology.ubc.ca/~srivast/zool502/Wkshp3-R.ppt

Logistic regression

p = proportion of successes

If $p = e^{ax+b} / (1+ e^{ax+b})$ calculate log(odds):

$$\log_e(p/1-p)$$

Source: www.zoology.ubc.ca/~srivast/zool502/Wkshp3-R.ppt

Logistic regression

Output from logistic regression with logit link:
predicted $\log_e (p/1-p) = a+bx$

To obtain any expected values of p, need to input a and b in original equation:

$p = e^{ax+b} / (1+ e^{ax+b})$

Source: www.zoology.ubc.ca/~srivast/zool502/Wkshp3-R.ppt

Logistic regression analysis

- Import the following file:
  - `binary.csv`
- During import, make sure you specify the separator as comma
- Recode **rank** from numeric to factor
- View the dataset

Logistic regression analysis

- Attributes of data:
  - This dataset has a binary response (outcome, dependent) variable called **admit**.
  - There are three predictor variables: **gre**, **gpa** and **rank**. We will treat the variables **gre** and **gpa** as continuous.
  - The variable **rank** takes on the values 1 through 4. Institutions with a rank of 1 have the highest prestige, while those with a rank of 4 have the lowest.

Logistic regression analysis

- Code:
  - `glm(formula = admit ~ gre + gpa + rank, family = binomial(logit), data = admit)`
- Predictors:
  - `gpa + gre + rank`
- Response:
  - `admit`
- Form:
  - Binomial response
  - Logit link

Logistic regression analysis: Summary

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.989979   1.139951  -3.500 0.000465 ***
gpa          0.804038   0.331819   2.423 0.015388 *
gre          0.002264   0.001094   2.070 0.038465 *
rank[T.2]   -0.675443   0.316490  -2.134 0.032829 *
rank[T.3]   -1.340204   0.345306  -3.881 0.000104 ***
rank[T.4]   -1.551464   0.417832  -3.713 0.000205 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 499.98  on 399  degrees of freedom
Residual deviance: 458.52  on 394  degrees of freedom
AIC: 470.52

Number of Fisher Scoring iterations: 4
```

Exercise 13

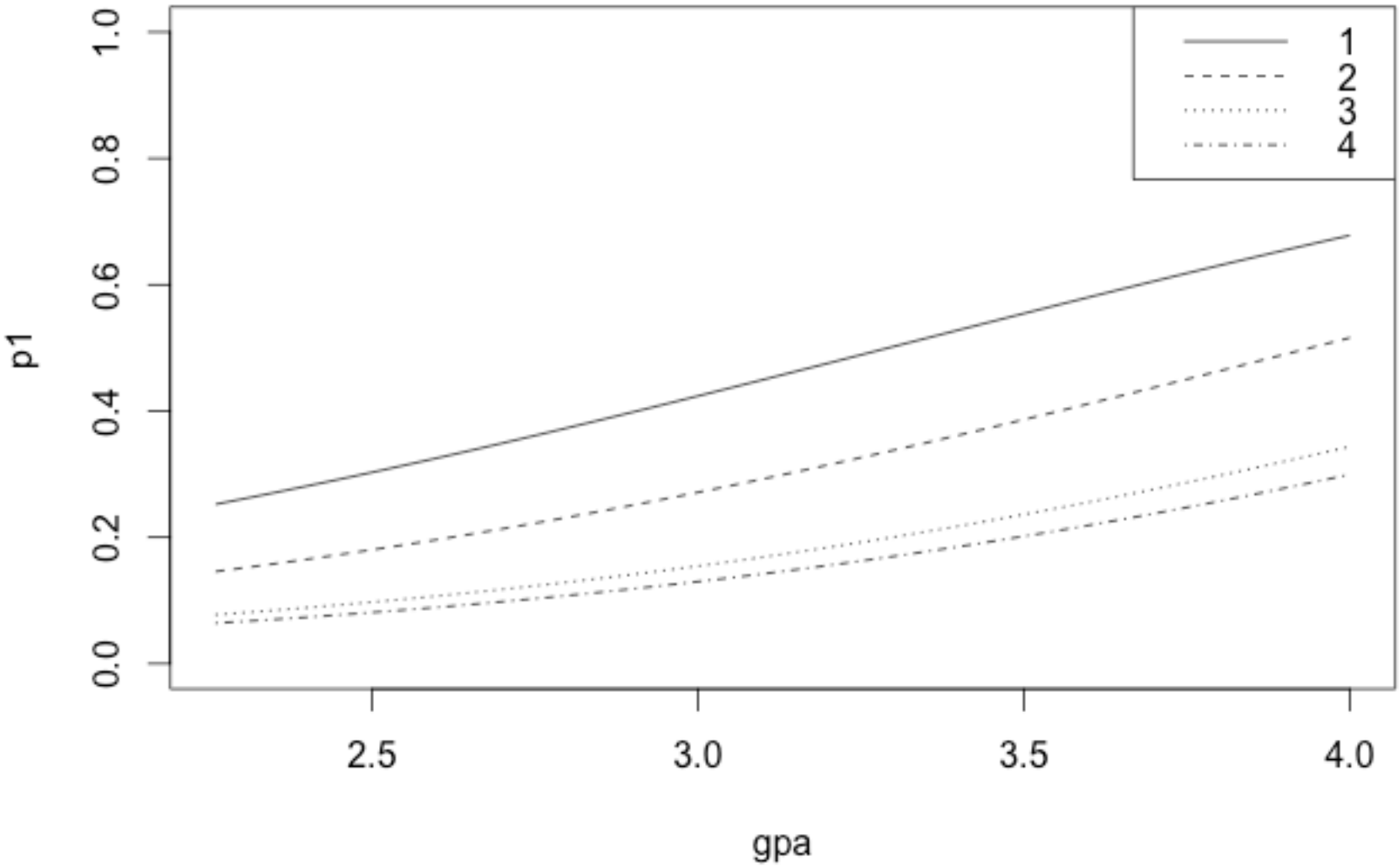
- Fit the full model, including `gpa`, `gre`, and `rank`, using `glm`
- Assess the significance of each term in the model using `drop1(…, ,test='Chisq')`
- Refit the model after dropping the term of lowest significance
- Harder: plot the predicted **P values** of the model for a range of predictor variables

Exercise 13 Answers

- `g <- glm(formula = admit ~ gre + rank + gpa, family = binomial(logit), data = binary)`
- `drop1(g, test='Chisq')`
- `g <- glm(formula = admit ~ gpa + rank, family = binomial(logit), data = binary)`

Plot of Model

- `gpa <- seq(2.26,4,length=100)`
- `p1 <- exp(1.0521*gpa-3.4636)/(1+exp(1.0521*gpa-3.4636))`
- `p2 <- exp(1.0521*gpa-3.4636-0.6810)/(1+exp(1.0521*gpa-3.4636-0.6810))`
- `p3 <- exp(1.0521*gpa-3.4636-1.3919)/(1+exp(1.0521*gpa-3.4636-1.3919))`
- `p4 <- exp(1.0521*gpa-3.4636-1.5943)/(1+exp(1.0521*gpa-3.4636-1.5943))`
- `plot(gpa,p1,ylim=c(0,1),lty=1,type='l')`
- `points(gpa,p2,ylim=c(0,1),lty=2,type='l')`
- `points(gpa,p3,ylim=c(0,1),lty=3,type='l')`
- `points(gpa,p4,ylim=c(0,1),lty=4,type='l')`
- `legend('topright',legend=1:4,lty=1:4)`


p1
0.0
0.2
0.4
0.6
0.8
1.0
2.5
3.0
3.5
4.0
gpa
1
2
3
4