

# Introduction to R – Thursday AM

## 1) Linear modelling

- One-way ANOVA
- Two-way ANOVA
- Multiple linear regression
- Combining continuous and categorical variables
- Generalised linear modelling
- Examples and exercises

## 5.5 Simple Linear Regression

- Use to estimate relationships between two variables and to test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model:  
`model <- lm(response ~ X, data=data)`
  2. Obtain information on the model fit:  
`summary(model)` # coefficients, R<sup>2</sup>, p-values  
`abline(model)` # adds prediction following `plot(X, Y)`  
`resid(model)` # returns model residuals  
`plot(model)` # returns diagnostic plots (normality, heteroscedasticity, leverage)

# 7.1 One-way ANOVA

- Use to estimate relationships between a continuous response variable and a *categorical* predictor variable and to test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model:  
`model <- lm(response ~ X, data=data)`
  2. Obtain information on the model fit:  
`summary(model)` # coefficients, R<sup>2</sup>, p-values  
`plot(model)` # returns diagnostic plots (normality, heteroscedasticity, leverage)  
`glht(model, linfct=mcp(X, 'Tukey'))` # performs multiple comparisons tests (in package 'multcomp')

```
> fit1 <- lm(height ~ species, data=cowsub)
> summary(fit1)
```

Call:

```
lm(formula = height ~ species, data = cowsub)
```

Residuals:

Min	1Q	Median	3Q	Max
-11.405	-2.062	0.695	3.299	8.194

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	21.8670	1.5645	13.977	7.02e-14 ***
speciescofl	-15.0695	2.9268	-5.149	2.04e-05 ***
speciesoxar	-5.3620	2.3467	-2.285	0.0304 *
speciesquru	-0.7614	2.2731	-0.335	0.7402

---

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

Residual standard error: 4.947 on 27 degrees of freedom

Multiple R-squared: 0.5326, Adjusted R-squared: 0.4807

F-statistic: 10.26 on 3 and 27 DF, p-value: 0.0001111

```
> anova(fit1)
```

Analysis of Variance Table

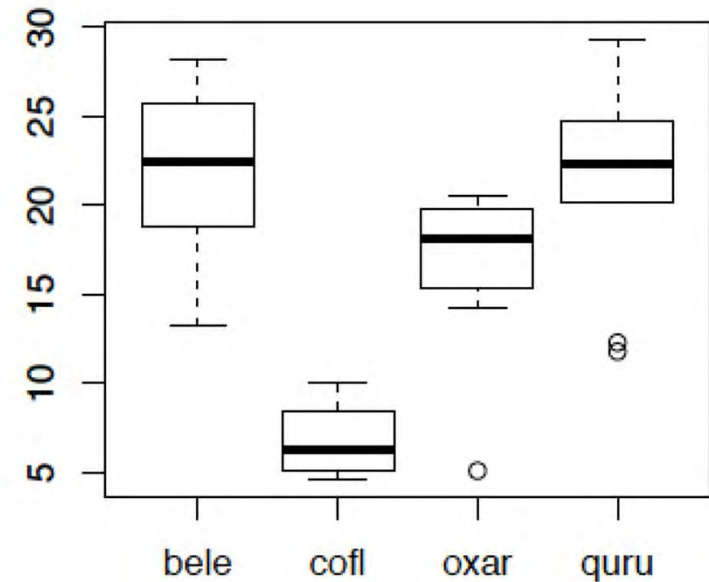
Response: height

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
species	3	753.07	251.024	10.256	0.0001111 ***
Residuals	27	660.83	24.475		

---

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

```
> |
```



## 7.1 One-way ANOVA (multiple comparisons)

```
> library(multcomp)
> tukey_Spec <- glht(fit1, linfct=mcp(species="Tukey"))
> summary(tukey_Spec)
```

### Simultaneous Tests for General Linear Hypotheses

#### Multiple Comparisons of Means: Tukey Contrasts

Fit: `lm(formula = height ~ species, data = cowsub)`

#### Linear Hypotheses:

	Estimate	Std. Error	t value	Pr(> t )	
cofl - bele == 0	-15.0695	2.9268	-5.149	<0.001	***
oxar - bele == 0	-5.3620	2.3467	-2.285	0.1248	
quru - bele == 0	-0.7614	2.2731	-0.335	0.9866	
oxar - cofl == 0	9.7075	3.0295	3.204	0.0167	*
quru - cofl == 0	14.3081	2.9729	4.813	<0.001	***
quru - oxar == 0	4.6006	2.4039	1.914	0.2433	

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
(Adjusted p values reported -- single-step method)

## 7.2 Two-way ANOVA

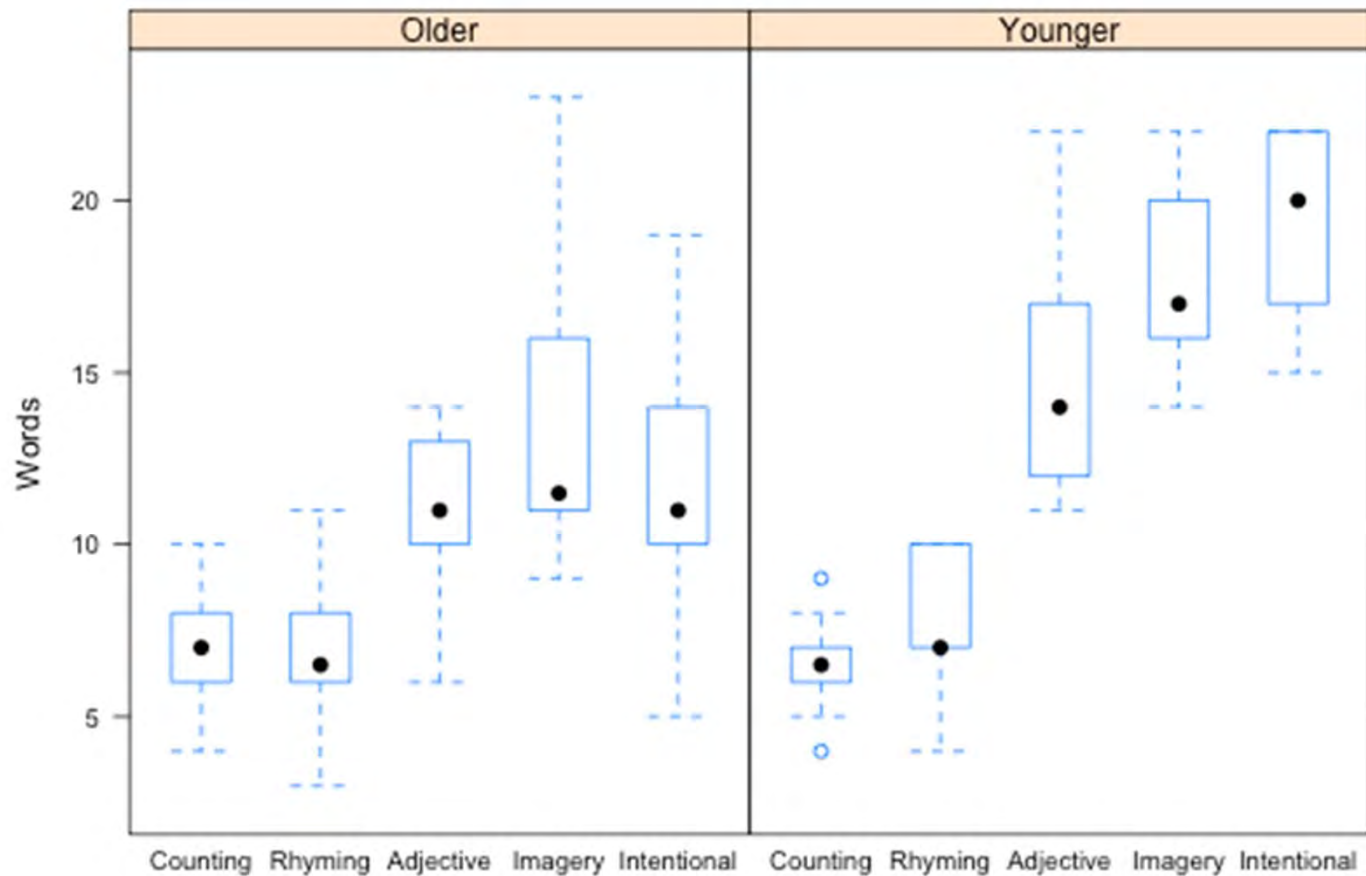
- Use to estimate relationships between a continuous response variable and multiple *categorical* predictor variables and to test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model:

```
model <- lm(response ~ X + Y + X:Y, data=data) # or  
model <- lm(response ~ X * Y, data=data)
```
  2. Obtain information on the model fit:

```
summary(model) # coefficients, R2, p-values  
anova(model) # generates ANOVA table  
drop1(model) # drops individual terms from model to  
test effect on model fit
```

## 7.2 Two-way ANOVA

```
> library(lattice)  
> bwplot(Words ~ Process | Age, data=memory)
```





## 7.2 Two-way ANOVA

```
# Two equivalent ways of specifying a linear model that includes all main effects
# and interactions:
fit3 <- lm(Words~Age+Process+Age:Process, data=memory)

# Is the same as:
fit3.2 <- lm(Words~Age*Process, data=memory)
anova(fit3.2)

## Analysis of Variance Table
##
## Response: Words
##
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)	
## Age	1	240	240	29.94	4e-07	***
## Process	4	1515	379	47.19	< 2e-16	***
## Age:Process	4	190	48	5.93	0.00028	***
## Residuals	90	722	8			

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



## Source

Console ~/RProjects/gitRepositories/hiermanual/data/

```
> summary(fit3.2)
```

Call:

```
lm(formula = Words ~ Age * Process, data = memory)
```

Residuals:

Min	1Q	Median	3Q	Max
-7.0	-1.6	-0.5	2.0	9.6

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	7.0000	0.8959	7.814	9.85e-12	***
AgeYounger	-0.5000	1.2669	-0.395	0.694031	
ProcessRhyming	-0.1000	1.2669	-0.079	0.937263	
ProcessAdjective	4.0000	1.2669	3.157	0.002168	**
ProcessImagery	6.4000	1.2669	5.052	2.28e-06	***
ProcessIntentional	5.0000	1.2669	3.947	0.000157	***
AgeYounger:ProcessRhyming	1.2000	1.7917	0.670	0.504731	
AgeYounger:ProcessAdjective	4.3000	1.7917	2.400	0.018458	*
AgeYounger:ProcessImagery	4.7000	1.7917	2.623	0.010231	*
AgeYounger:ProcessIntentional	7.8000	1.7917	4.353	3.54e-05	***

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

Residual standard error: 2.833 on 90 degrees of freedom

Multiple R-squared: 0.7293, Adjusted R-squared: 0.7022

F-statistic: 26.93 on 9 and 90 DF, p-value: < 2.2e-16

```
> visreg(fit3.2, 'Process', 'Age', overlay=T)
```

```
>
```

## Environment

## History

## Git

## Files

## Plots

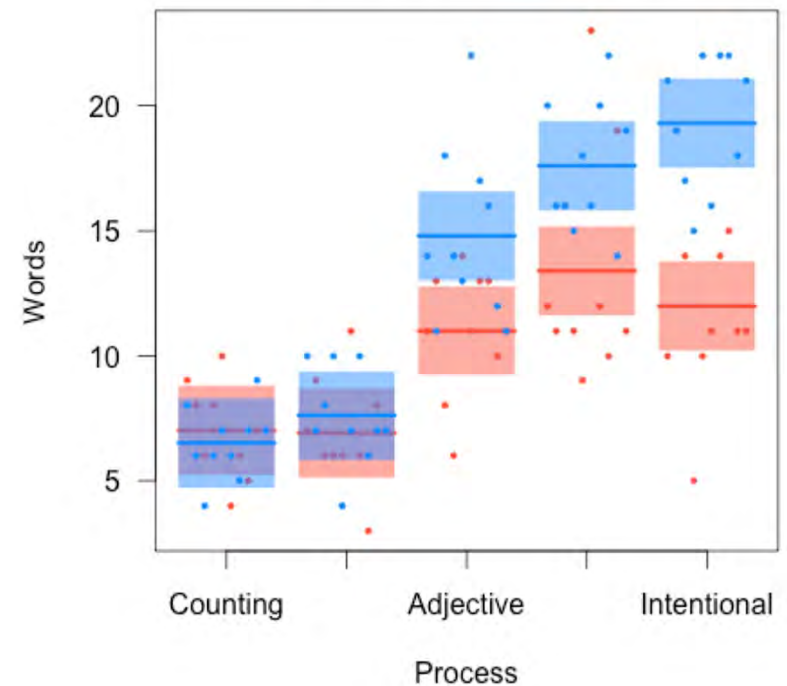
## Packages

## Help

## Viewer



— Older — Younger



## 7.3 Multiple linear regression

- Use to estimate relationships between a continuous response variable and multiple *continuous* predictor variables and to test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model (e.g.):

```
model <- lm(response ~ X + Y + X:Y, data=data) # or  
model <- lm(response ~ X * Y, data=data)
```
  2. Obtain information on the model fit:

```
summary(model) # coefficients, R2, p-values  
drop1(model) # drops individual terms from model to  
test effect on model fit
```

## 7.3 Multiple linear regression

```
fit5 <- lm(logLA~logD+logH, data=allom)
summary(fit5)

##
## Call:
## lm(formula = logLA ~ logD + logH, data = allom)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0723 -0.1701  0.0176  0.1735  0.4521
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.413     0.155   -2.66   0.010 *
## logD           2.190     0.304    7.21  1.1e-09 ***
## logH          -0.734     0.322   -2.28   0.026 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.253 on 60 degrees of freedom
## Multiple R-squared:  0.79, Adjusted R-squared:  0.783
## F-statistic: 113 on 2 and 60 DF, p-value: <2e-16
```

## 7.4 Factors and continuous predictors

- Use to estimate relationships between a continuous response variable and multiple predictor variables, both *continuous and categorical*, and to test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model (e.g.):

```
model <- lm(response ~ X + Y + Z + X:Y + X:Z, data=data)
model <- lm(response ~ X * Y + X * Z, data=data)
```
  2. Obtain information on the model fit:

```
summary(model) # coefficients, R2, p-values
anova(model)   # generates ANOVA table
drop1(model)   # drops individual terms from model to
                test effect on model fit
```

## 7.4 Factors and continuous predictors

```
# A linear model that includes an interaction between a factor and a continuous variable
fit8 <- lm(logLA~species*logD+species*logH, data=allom)
summary(fit8)

##
## Call:
## lm(formula = logLA ~ species * logD + species * logH, data = allom)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0554 -0.0881 -0.0075  0.1148  0.3412
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.370     0.253   -1.46   0.1498
## speciesPIP0    -0.407     0.348   -1.17   0.2473
## speciesPSME     0.225     0.331    0.68   0.4997
## logD           1.398     0.496    2.82   0.0067 **
## logH           0.161     0.526    0.31   0.7606
## speciesPIP0:logD 1.503     0.650    2.31   0.0246 *
## speciesPSME:logD 1.723     0.718    2.40   0.0200 *
## speciesPIP0:logH -1.524     0.674   -2.26   0.0278 *
## speciesPSME:logH -2.089     0.790   -2.64   0.0107 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

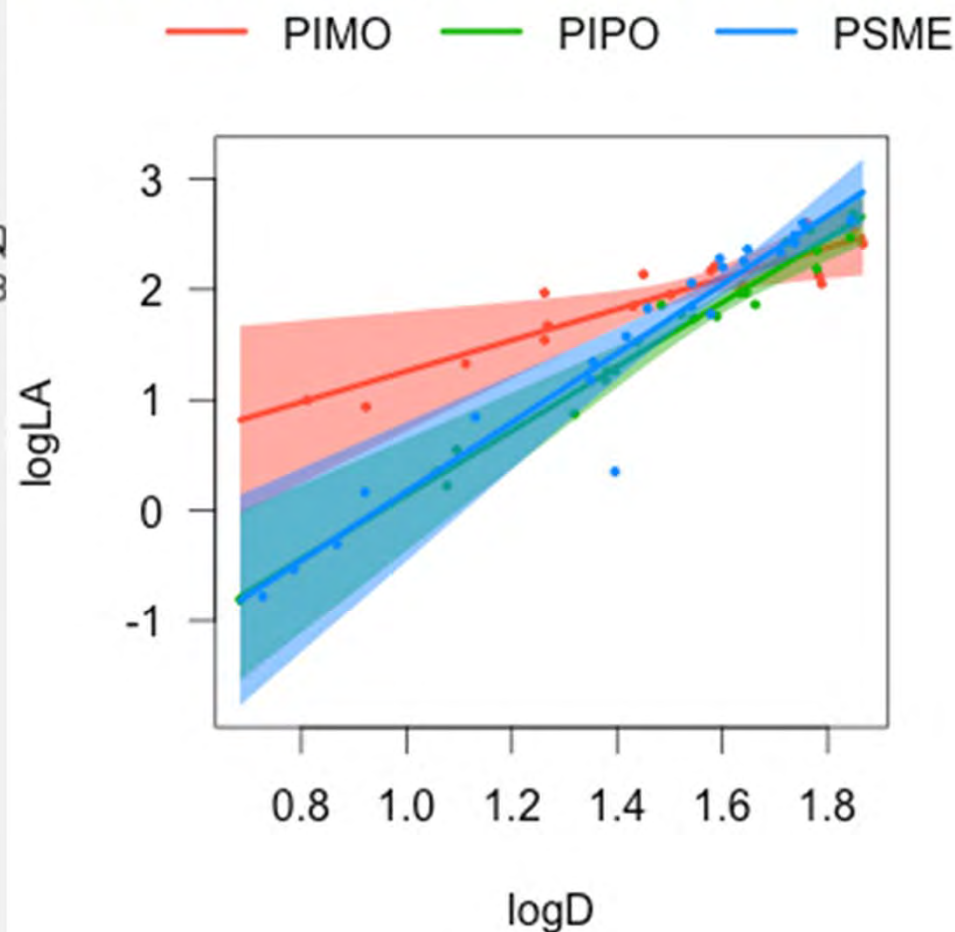


## 7.4 Factors and continuous predictors

```
# A linear model that includes an interaction between a factor and a continuous variable
fit8 <- lm(logLA~species*logD+species*logH, data=allom)
summary(fit8)
```

```
##
## Call:
## lm(formula = logLA ~ species *
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0554 -0.0881 -0.0075  0.1148  0.8112
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    Signif. codes:  0 '***' 0.001 '***' 0.01 '***' 0.05 '***' 0.1 '***' 1
## (Intercept)      -0.370      0.181   -2.045  0.041 ***
## speciesPIPO       -0.407      0.181   -2.248  0.027 ***
## speciesPSME        0.225      0.181    1.243  0.218
## logD              1.398      0.181    7.721 <0.001 ***
## logH              0.161      0.181    0.889  0.374
## speciesPIPO:logD  1.503      0.181    8.301 <0.001 ***
## speciesPSME:logD  1.723      0.181    9.518 <0.001 ***
## speciesPIPO:logH -1.524      0.181   -8.421 <0.001 ***
## speciesPSME:logH -2.089      0.181  -11.541 <0.001 ***
## ---
## Signif. codes:  0 '***' 0.001 '***' 0.01 '***' 0.05 '***' 0.1 '***' 1
```

```
> visreg(fit8, 'logD', 'species', overlay=T)
```



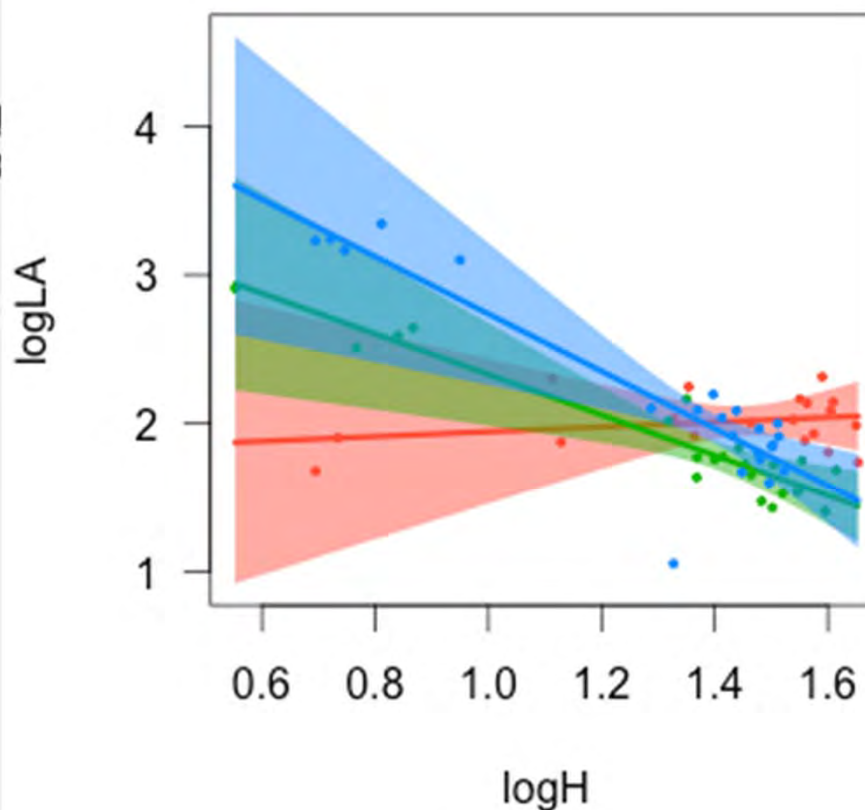
## 7.4 Factors and continuous predictors

```
# A linear model that includes an interaction between a factor and a continuous variable
fit8 <- lm(logLA~species*logD+species*logH, data=allom)
summary(fit8)
```

```
##
## Call:
## lm(formula = logLA ~ species *
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0554 -0.0881 -0.0075  0.1148  0.7000
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    Signif. codes: 0 '***' 0.001 '
## (Intercept)      -0.370      0.151   -2.450  0.0154 ***
## speciesPIPO       -0.407      0.151   -2.700  0.0084 ***
## speciesPSME        0.225      0.151    1.490  0.1368
## logD              1.398      0.151    9.250  <0.001 ***
## logH              0.161      0.151    1.060  0.2918
## speciesPIPO:logD  1.503      0.151    9.950  <0.001 ***
## speciesPSME:logD  1.723      0.151   11.400  <0.001 ***
## speciesPIPO:logH -1.524      0.151   -10.000  <0.001 ***
## speciesPSME:logH -2.089      0.151   -13.800  <0.001 ***
## ---
## Signif. codes: 0 '***' 0.001 '***' 0.01 '***' 0.05 '***' 0.1 '***' 1
```

```
> visreg(fit8, 'logH', 'species', overlay=T)
```

— PIMO — PIPO — PSME





## 7.4 Factors and continuous predictors

```
anova(fit8)

## Analysis of Variance Table
##
## Response: logLA
##           Df Sum Sq Mean Sq F value   Pr(>F)
## species      2   0.54    0.27    5.84 0.00507 **
## logD          1  14.20   14.20  305.64 < 2e-16 ***
## logH          1   0.58    0.58   12.47 0.00085 ***
## species:logD  2   0.06    0.03    0.61 0.54494
## species:logH  2   0.38    0.19    4.04 0.02322 *
## Residuals    54   2.51    0.05
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

drop1(fit8, test="F")

## Single term deletions
##
## Model:
## logLA ~ species * logD + species * logH
##           Df Sum of Sq  RSS   AIC F value Pr(>F)
## <none>                 2.51 -185
## species:logD  2      0.338 2.85 -181   3.64 0.033 *
## species:logH  2      0.375 2.88 -180   4.04 0.023 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## Specify logD before logH

```
> anova(fit8)
```

Analysis of Variance Table

Response: logLA

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
species	2	0.5424	0.2712	5.8367	0.0050712 **
logD	1	14.2003	14.2003	305.6356	< 2.2e-16 ***
logH	1	0.5794	0.5794	12.4707	0.0008548 ***
species:logD	2	0.0571	0.0285	0.6140	0.5449374
species:logH	2	0.3752	0.1876	4.0378	0.0232150 *
Residuals	54	2.5089	0.0465		

---

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

```
> library(car)
```

```
> Anova(fit8)
```

Anova Table (Type II tests)

Response: logLA

	Sum Sq	Df	F value	Pr(>F)
species	0.8855	2	9.5294	0.0002854 ***
logD	3.9165	1	84.2945	1.286e-12 ***
logH	0.6102	1	13.1325	0.0006425 ***
species:logD	0.3382	2	3.6394	0.0329039 *
species:logH	0.3752	2	4.0378	0.0232150 *
Residuals	2.5089	54		

---

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

```
> |
```

≠

=

## Specify logH before logD

```
> anova(fit9)
```

Analysis of Variance Table

Response: logLA

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
species	2	0.5424	0.2712	5.8367	0.005071 **
logH	1	10.9145	10.9145	234.9142	< 2.2e-16 ***
logD	1	3.8652	3.8652	83.1920	1.6e-12 ***
species:logH	2	0.0941	0.0470	1.0123	0.370160
species:logD	2	0.3382	0.1691	3.6394	0.032904 *
Residuals	54	2.5089	0.0465		

---

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

```
>
```

```
> Anova(fit9)
```

Anova Table (Type II tests)

Response: logLA

	Sum Sq	Df	F value	Pr(>F)
species	0.8855	2	9.5294	0.0002854 ***
logH	0.6102	1	13.1325	0.0006425 ***
logD	3.9165	1	84.2945	1.286e-12 ***
species:logH	0.3752	2	4.0378	0.0232150 *
species:logD	0.3382	2	3.6394	0.0329039 *
Residuals	2.5089	54		

---

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

```
> |
```

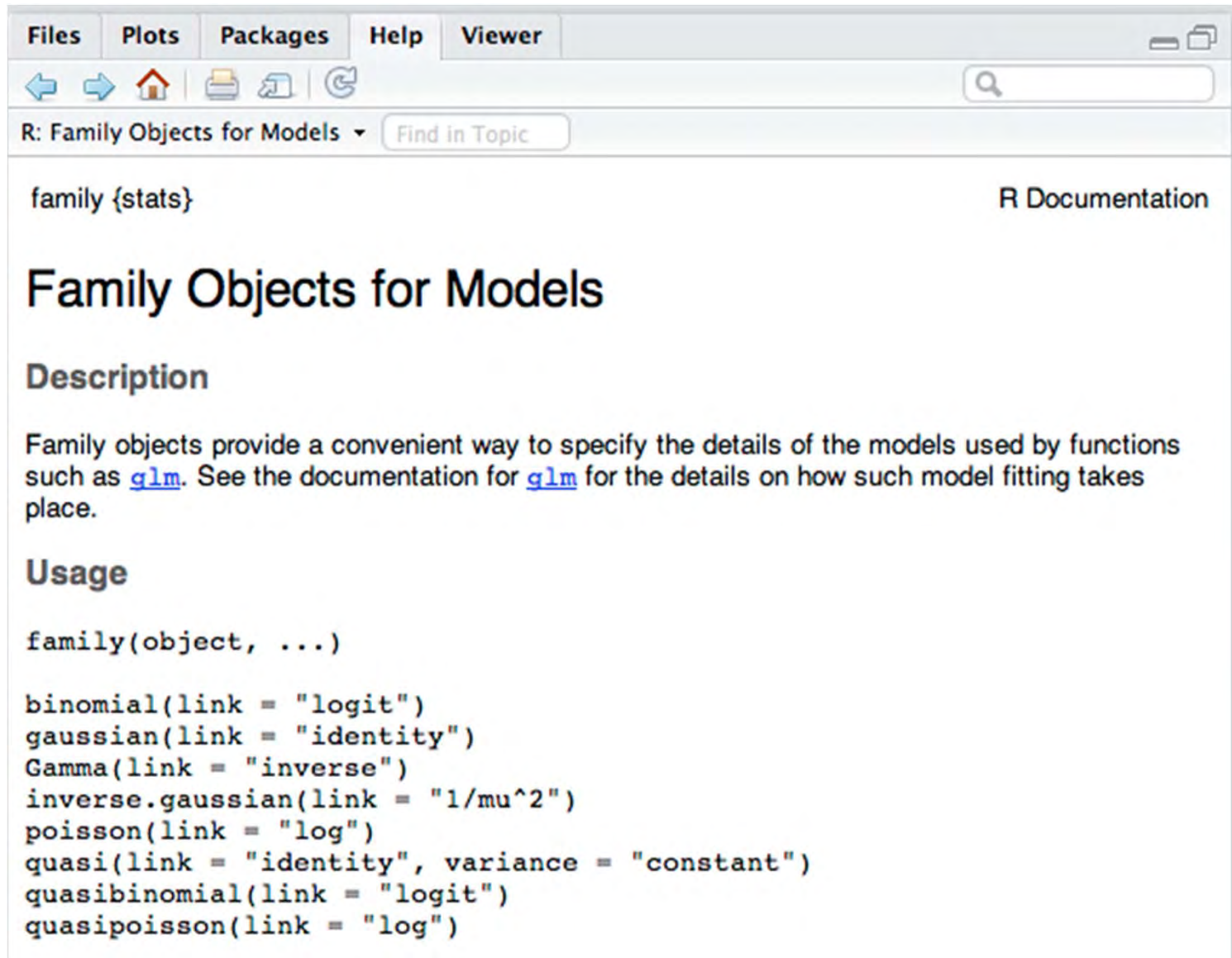
## 7.5/6 Generalised linear models

- Use to estimate relationships between non-normally distributed response variables (*continuous, categorical*) and one or more predictor variables (*continuous, categorical*), and test hypotheses regarding these relationships
- Two-step process (see example in manual):
  1. Fit the model (e.g.):

```
model <- glm(response ~ X * Y * Z, data=data,  
             family=*family*)
```
  2. Obtain information on the model fit:

```
summary(model) # coefficients, p-values  
anova(model, test=*test*) # generates ANOVA table  
drop1(model) # drops individual terms from model to  
               test effect on model fit
```

# 7.5/6 Generalised linear models



The screenshot shows the R documentation interface. At the top, there are tabs for 'Files', 'Plots', 'Packages', 'Help', and 'Viewer'. Below these is a toolbar with navigation icons and a search bar. The main content area is titled 'R: Family Objects for Models' and includes a 'Find in Topic' search box. The text 'family {stats}' is in the top left, and 'R Documentation' is in the top right. The main heading is 'Family Objects for Models'. Below this is a 'Description' section with a paragraph explaining that family objects provide a convenient way to specify model details, with links to [glm](#) documentation. The 'Usage' section follows, showing the `family(object, ...)` function and a list of model families: `binomial(link = "logit")`, `gaussian(link = "identity")`, `Gamma(link = "inverse")`, `inverse.gaussian(link = "1/mu^2")`, `poisson(link = "log")`, `quasi(link = "identity", variance = "constant")`, `quasibinomial(link = "logit")`, and `quasipoisson(link = "log")`.

Files Plots Packages Help Viewer

R: Family Objects for Models Find in Topic

family {stats} R Documentation

## Family Objects for Models

### Description

Family objects provide a convenient way to specify the details of the models used by functions such as [glm](#). See the documentation for [glm](#) for the details on how such model fitting takes place.

### Usage

```
family(object, ...)
```

```
binomial(link = "logit")
gaussian(link = "identity")
Gamma(link = "inverse")
inverse.gaussian(link = "1/mu^2")
poisson(link = "log")
quasi(link = "identity", variance = "constant")
quasibinomial(link = "logit")
quasipoisson(link = "log")
```



## 7.5/6 Generalised linear models

```
# Fit a generalized linear model
fit13 <- glm(Words~Age*Process, data=memory, family=poisson)
summary(fit13)

##
## Call:
## glm(formula = Words ~ Age * Process, family = poisson, data = memory)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.290  -0.492  -0.199   0.562   2.377
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      1.9459    0.1195  16.28 < 2e-16 ***
## AgeYounger     -0.0741    0.1723  -0.43  0.66703
## ProcessRhyming  -0.0144    0.1696  -0.08  0.93241
## ProcessAdjective  0.4520    0.1529   2.96  0.00311 **
## ProcessImagery   0.6493    0.1475   4.40  1.1e-05 ***
## ProcessIntentional 0.5390    0.1504   3.58  0.00034 ***
## AgeYounger:ProcessRhyming 0.1707    0.2394   0.71  0.47577
## AgeYounger:ProcessAdjective 0.3708    0.2133   1.74  0.08218 .
## AgeYounger:ProcessImagery 0.3468    0.2069   1.68  0.09378 .
## AgeYounger:ProcessIntentional 0.5493    0.2078   2.64  0.00821 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

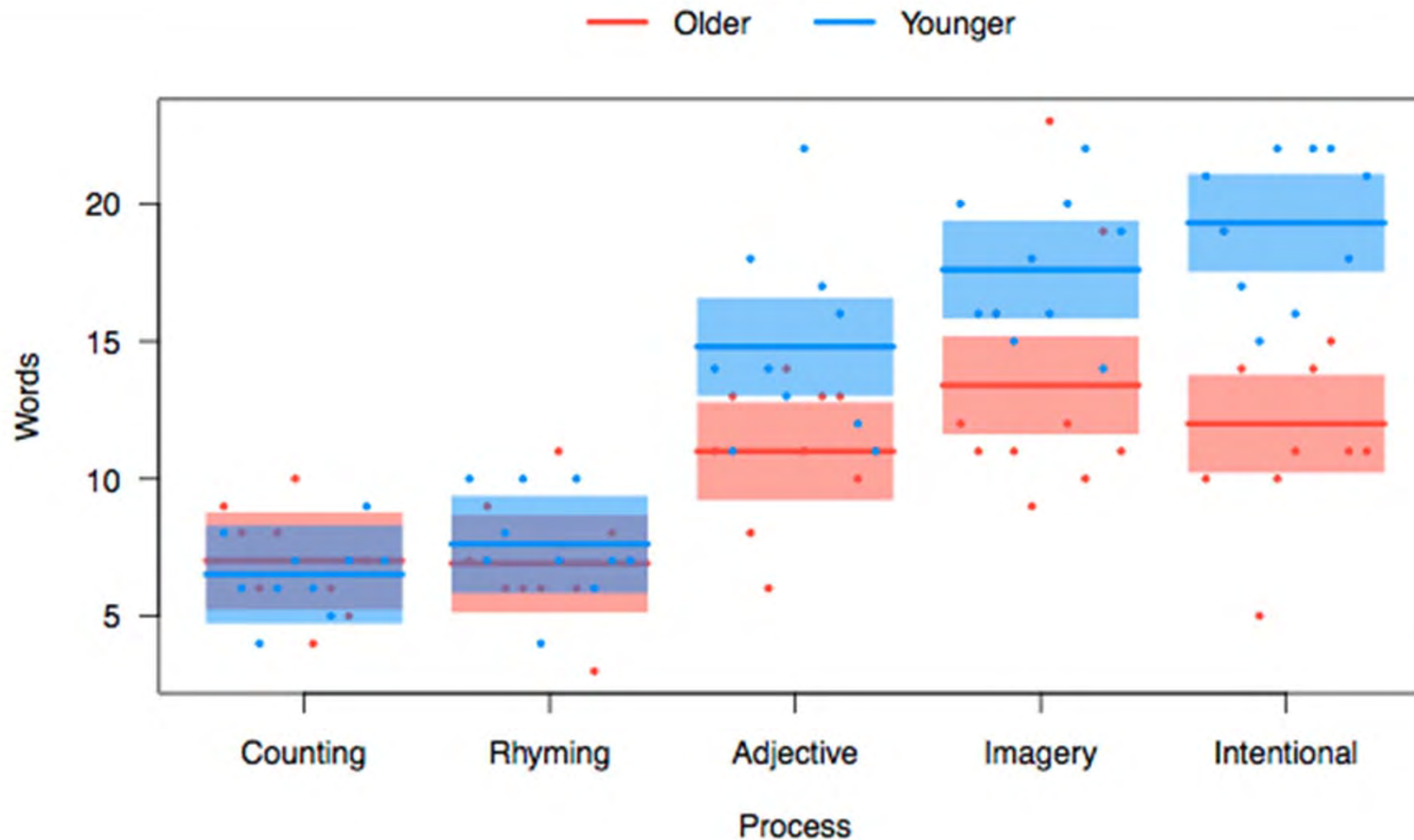
## 7.5/6 Generalised linear models

```
# Fit a generalized linear model
fit13 <- glm(Words~Age*Process, data=memory, family=poisson)
# Look at an ANOVA of the fitted model, and provide likelihood-ratio tests.
anova(fit13, test="LRT")

## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Words
##
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                99      227.5
## Age                1      20.8      98      206.7 5.2e-06 ***
## Process            4     137.5      94       69.3 < 2e-16 ***
## Age:Process        4       8.3      90       61.0 0.082 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Model predictions - using 'visreg' library

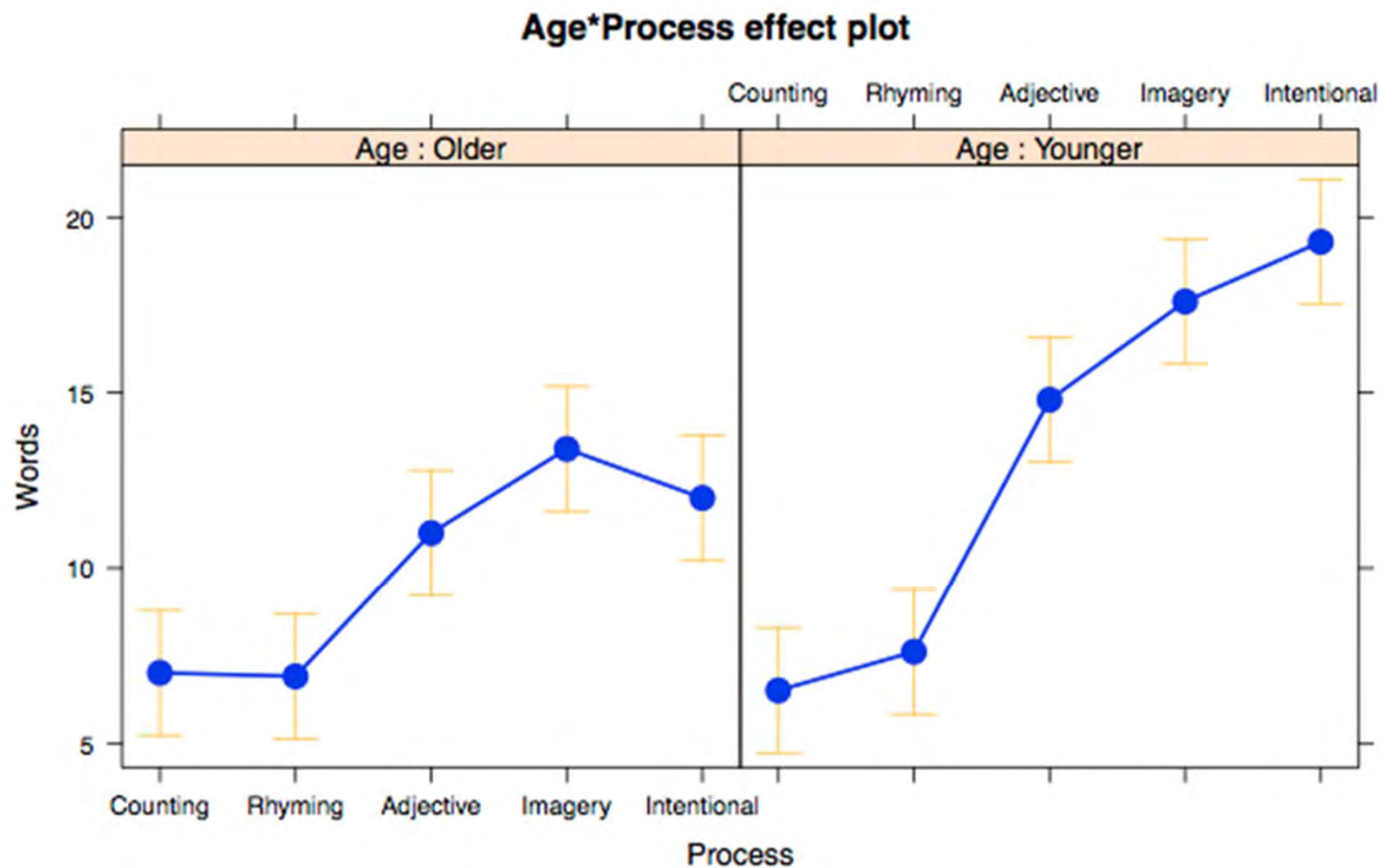
```
fit10 <- lm(Words~Age*Process, data=memory)
# Here we specify which variable should be on the X axis (Process),
# and which variable should be added with different colours (Age).
visreg(fit10, "Process", by="Age", overlay=TRUE)
```





# Model predictions - using 'effects' library

```
fit10 <- lm(Words~Age*Process, data=memory)  
plot(allEffects(fit10))
```



# Introduction to R – Thursday AM

## 1) Linear modelling

- One-way ANOVA
- Two-way ANOVA
- Multiple linear regression
- Combining continuous and categorical variables
- Generalised linear modelling
- **Examples and exercises**

