BIO 754 - Lecture 14

25-05-2017

Contents

Multiple regression and model choice

The Galapagos dataset

Plant species numbers, endemic species numbers, and area, elevation, etc of Galapagos islands.

https://upload.wikimedia.org/wikipedia/commons/e/e7/Galapagos%2Bmap.jpg

```
library(faraway)
# str gives a useful summary, including dimensions and the first 6 entries
str(gala)
```

```
## 'data.frame':    30 obs. of  7 variables:
##  $ Species  : num  58 31 3 25 2 18 24 10 8 2 ...
##  $ Endemics : num  23 21 3 9 1 11 0 7 4 2 ...
##  $ Area     : num  25.09 1.24 0.21 0.1 0.05 ...
##  $ Elevation: num  346 109 114 46 77 119 93 168 71 112 ...
##  $ Nearest  : num  0.6 0.6 2.8 1.9 1.9 8 6 34.1 0.4 2.6 ...
##  $ Scruz    : num  0.6 26.3 58.7 47.4 1.9 ...
##  $ Adjacent : num  1.84 572.33 0.78 0.18 903.82 ...
```

```
dim(gala)
```

```
## [1] 30  7
```

```
head(gala)
```

```
##              Species Endemics  Area Elevation Nearest Scruz Adjacent
## Baltra            58       23 25.09       346     0.6   0.6     1.84
```

```
## Bartolome         31       21  1.24       109     0.6  26.3   572.33
## Caldwell           3        3  0.21       114     2.8  58.7     0.78
## Champion          25        9  0.10        46     1.9  47.4     0.18
## Coamano            2        1  0.05        77     1.9   1.9   903.82
## Daphne.Major      18       11  0.34       119     8.0   8.0     1.84
```

```
summary(gala)
```

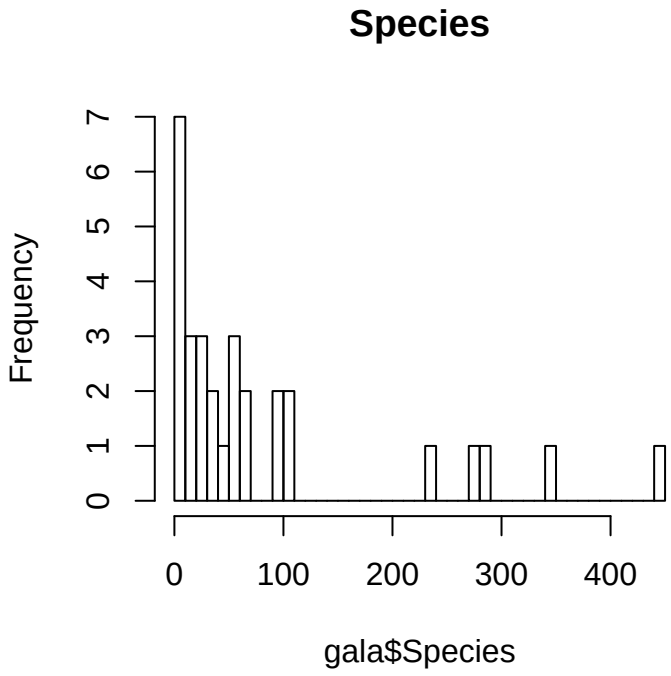
```
##     Species          Endemics          Area            Elevation
##  Min.   :  2.00   Min.   : 0.00   Min.   :   0.010   Min.   :  25.00
##  1st Qu.: 13.00   1st Qu.: 7.25   1st Qu.:   0.258   1st Qu.:  97.75
##  Median : 42.00   Median :18.00   Median :   2.590   Median : 192.00
##  Mean   : 85.23   Mean   :26.10   Mean   : 261.709   Mean   : 368.03
##  3rd Qu.: 96.00   3rd Qu.:32.25   3rd Qu.:  59.237   3rd Qu.: 435.25
##  Max.   :444.00   Max.   :95.00   Max.   :4669.320   Max.   :1707.00
##     Nearest           Scruz           Adjacent
##  Min.   : 0.20   Min.   :  0.00   Min.   :   0.03
##  1st Qu.: 0.80   1st Qu.: 11.03   1st Qu.:   0.52
##  Median : 3.05   Median : 46.65   Median :   2.59
##  Mean   :10.06   Mean   : 56.98   Mean   : 261.10
##  3rd Qu.:10.03   3rd Qu.: 81.08   3rd Qu.:  59.24
##  Max.   :47.40   Max.   :290.20   Max.   :4669.32
```

So all data seems numeric and continuous. Each row corresponds to an island.

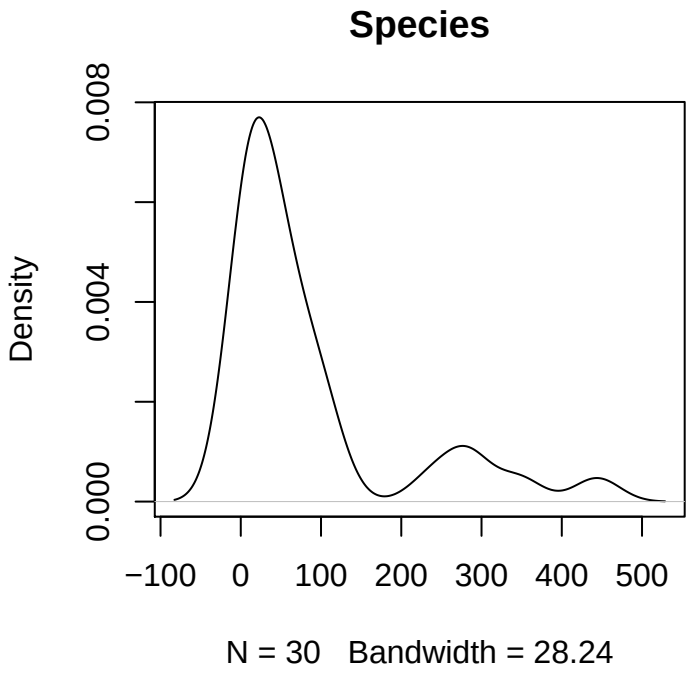
The `density` function

`Species` is the variable of highest interest and we hope to explain. It is the number of different plant species across islands. How is this distributed? We could study this using a histogram. But one could also use a density plot.

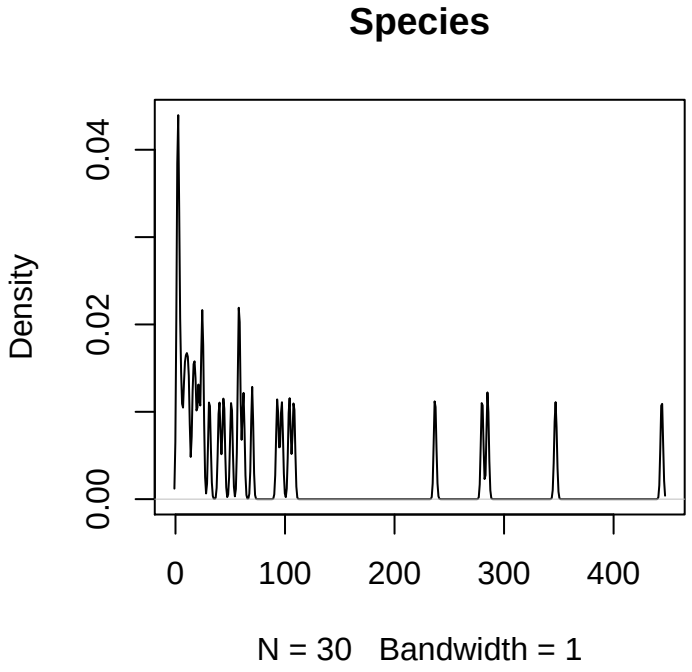
```
hist(gala$Species, br=50, main="Species")
```



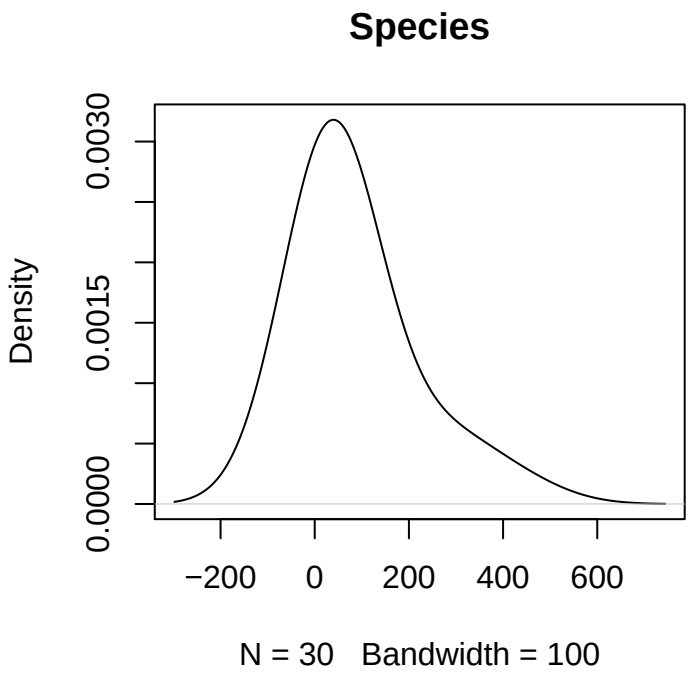
```r
plot(density(gala$Species), main="Species")
```




```r
plot(density(gala$Species, bw = 1), main="Species")
```

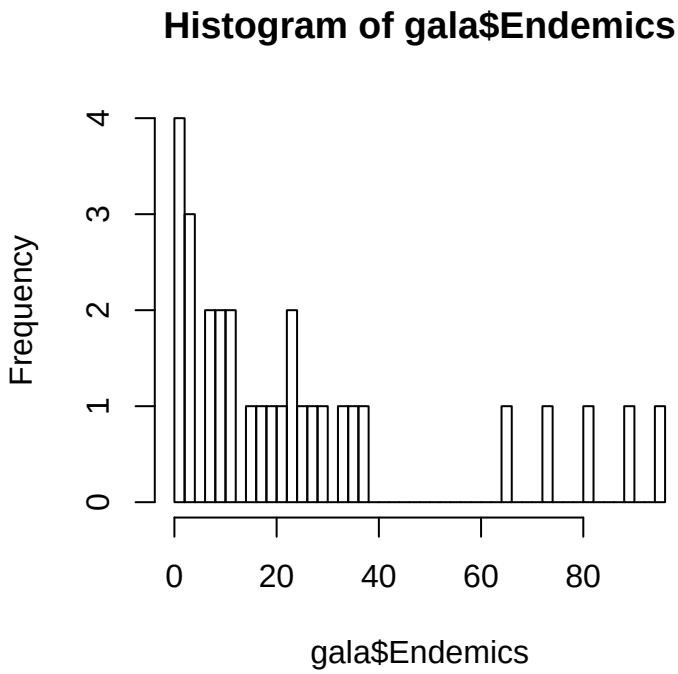



```r
plot(density(gala$Species, bw = 100), main="Species")
```

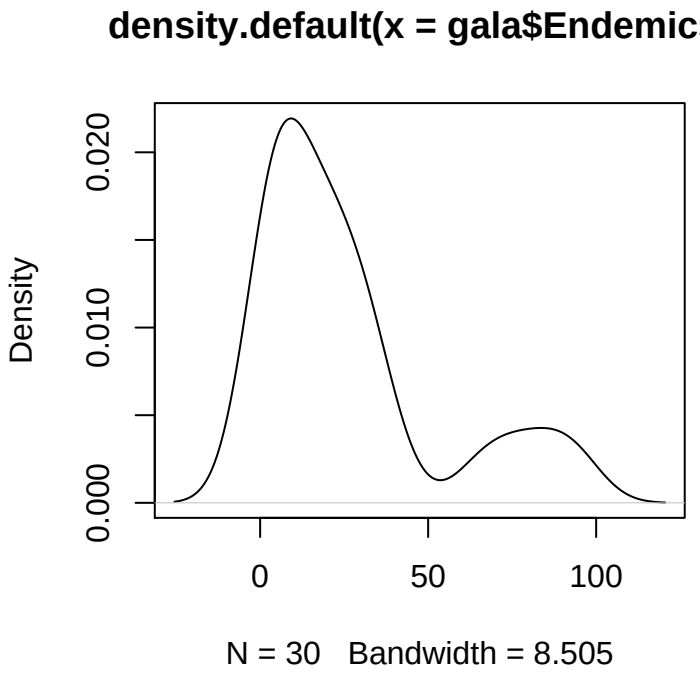


So this is a highly right-skewed distribution. How about the endemic species' numbers?

```
hist(gala$Endemics, br=50)
```

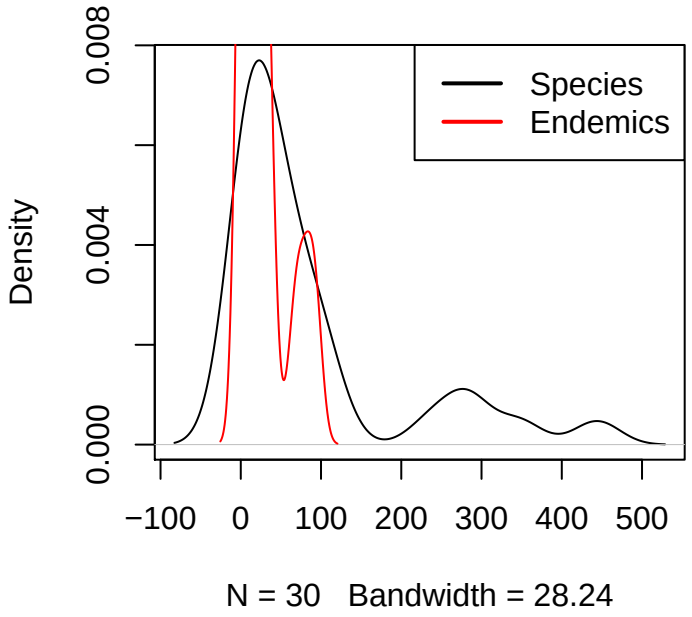



```
plot(density(gala$Endemics))
```



What if we want to compare this distribution of `Endemics` to that of `Species`? We could plot two histograms over each other. But this might be difficult to display nicely. Instead, drawing two density plots over each other is much easier:

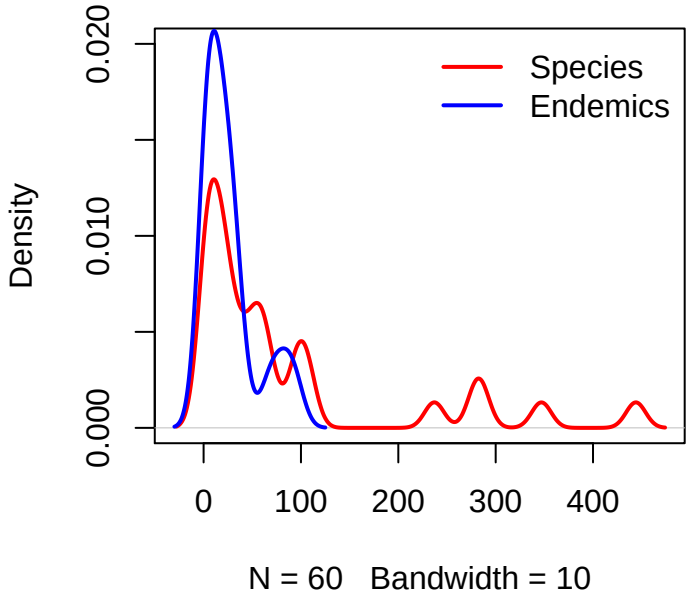
```
plot(density(gala$Species), main="")
lines(density(gala$Endemics), col=2)
legend("topright", legend = c("Species", "Endemics"), lwd = 2, col = 1:2)
```



However, this also looks imperfect. We need to fit both in. One trick is to join the data, suppress plotting, and then add lines:

```
# type="n" suppresses plotting inside the frame
plot(density(c(gala$Endemics, gala$Species), bw=10), main="", type="n", ylim=c(0, 0.02))
lines(density(gala$Species, bw=10), col=2, lwd=2)
```

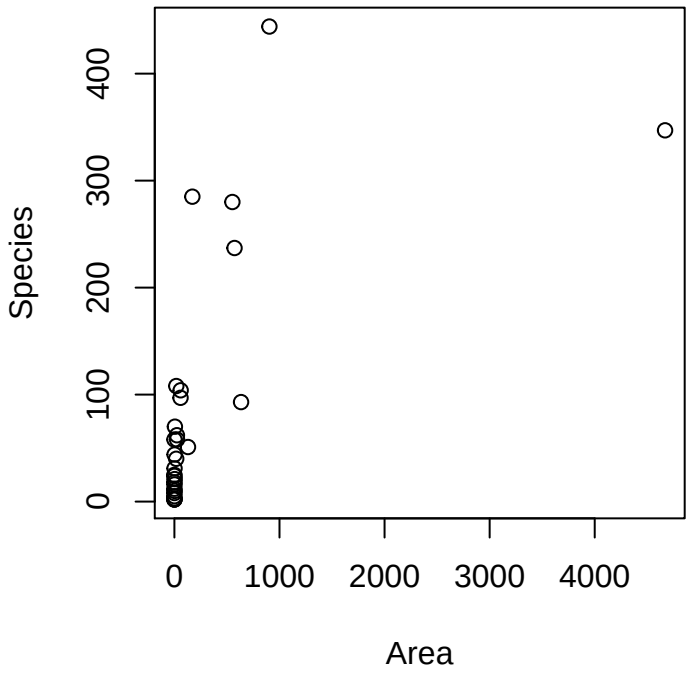
```
lines(density(gala$Endemics, bw=10), col="blue", lwd=2)
legend("topright", legend = c("Species", "Endemics"), lwd = 2, col = c("red", "blue"), bty="n")
```



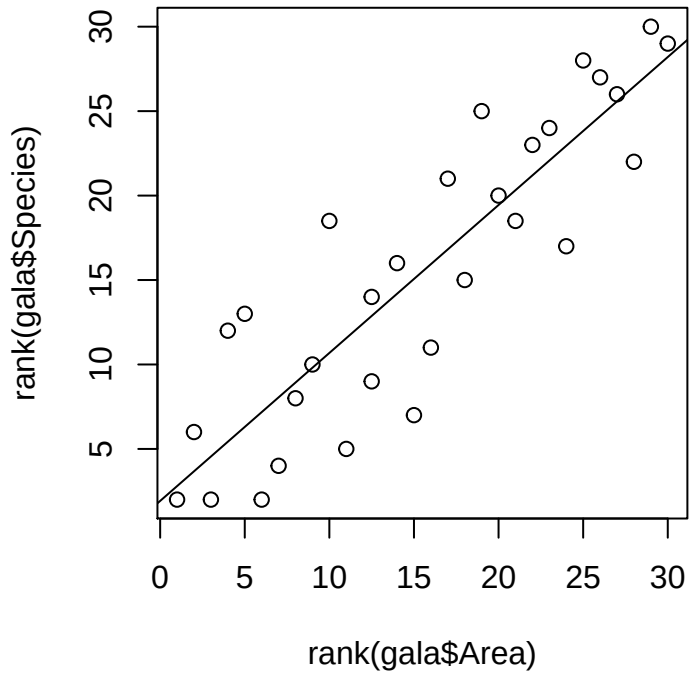
biplot and pairs functions

The variables interrelated? We can study one-by-one or in bulk:

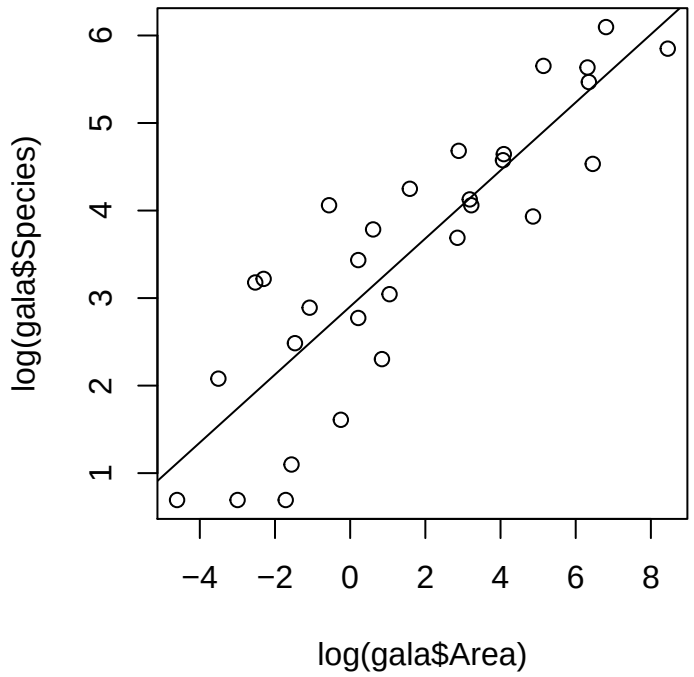
```
plot(Species ~ Area, gala)
```



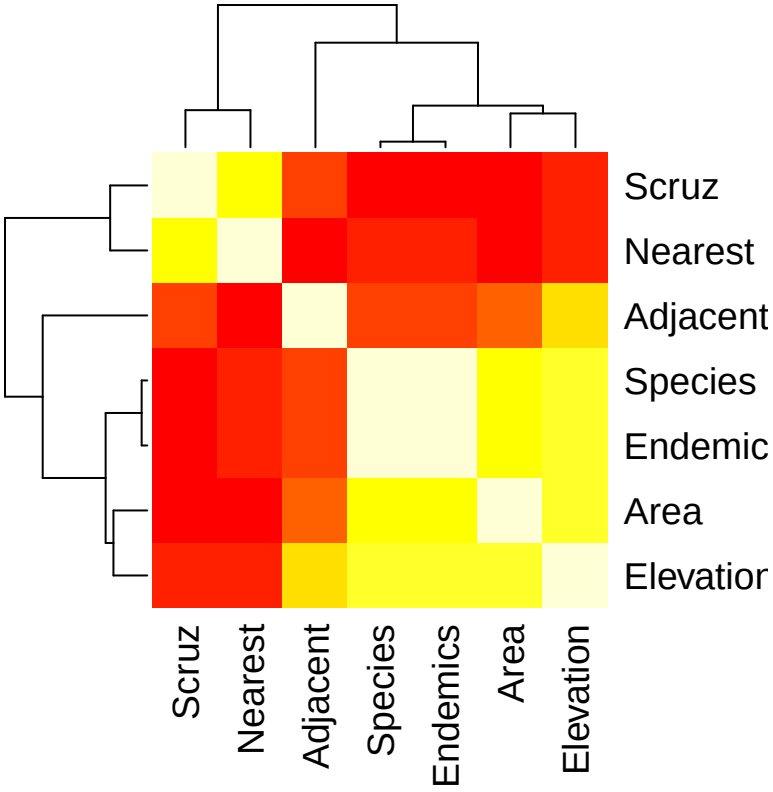

```
plot(rank(gala$Area), rank(gala$Species) )
abline( lm(rank(gala$Species) ~ rank(gala$Area)) )
```



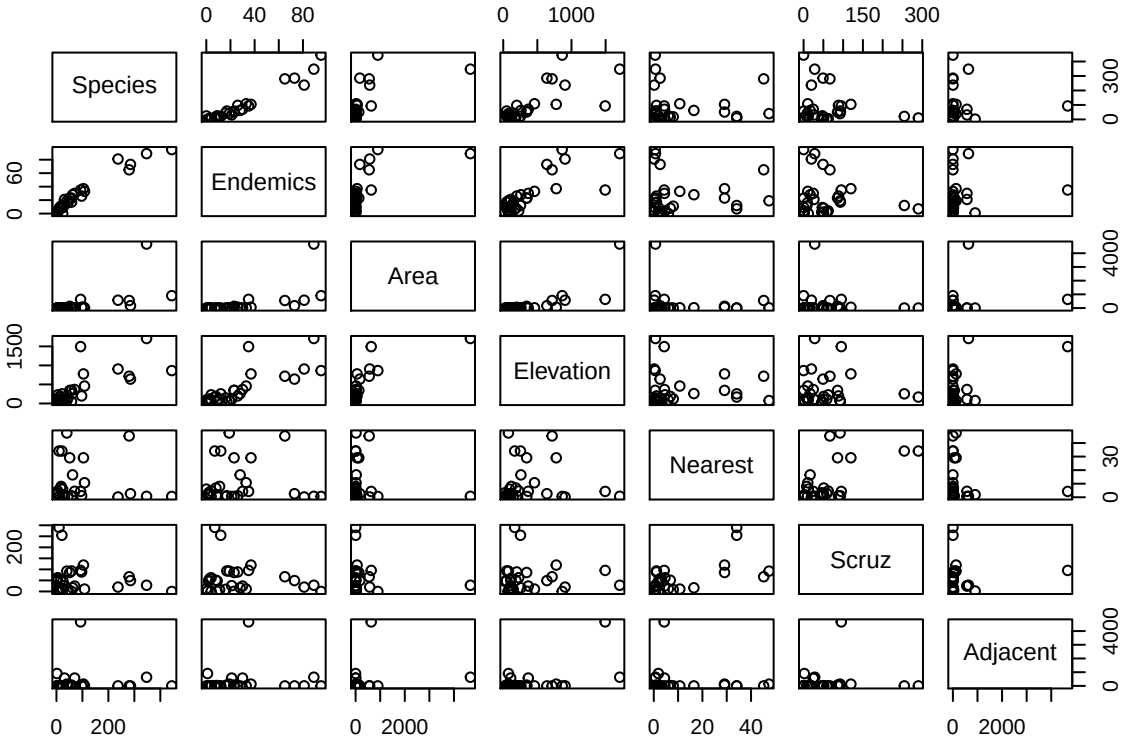

```
plot(log(gala$Area), log(gala$Species) )
abline( lm(log(gala$Species) ~ log(gala$Area)) )
```




```
heatmap(cor(gala), symm = T)
```

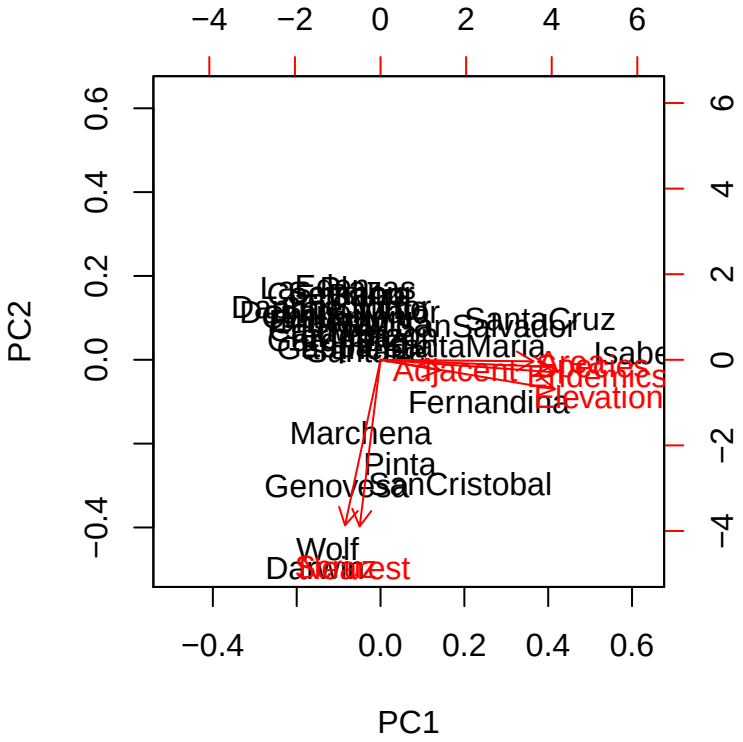



```
# correlation between pairs of variables
pairs(gala)
```




```
# this runs the same: plot(gala)
```

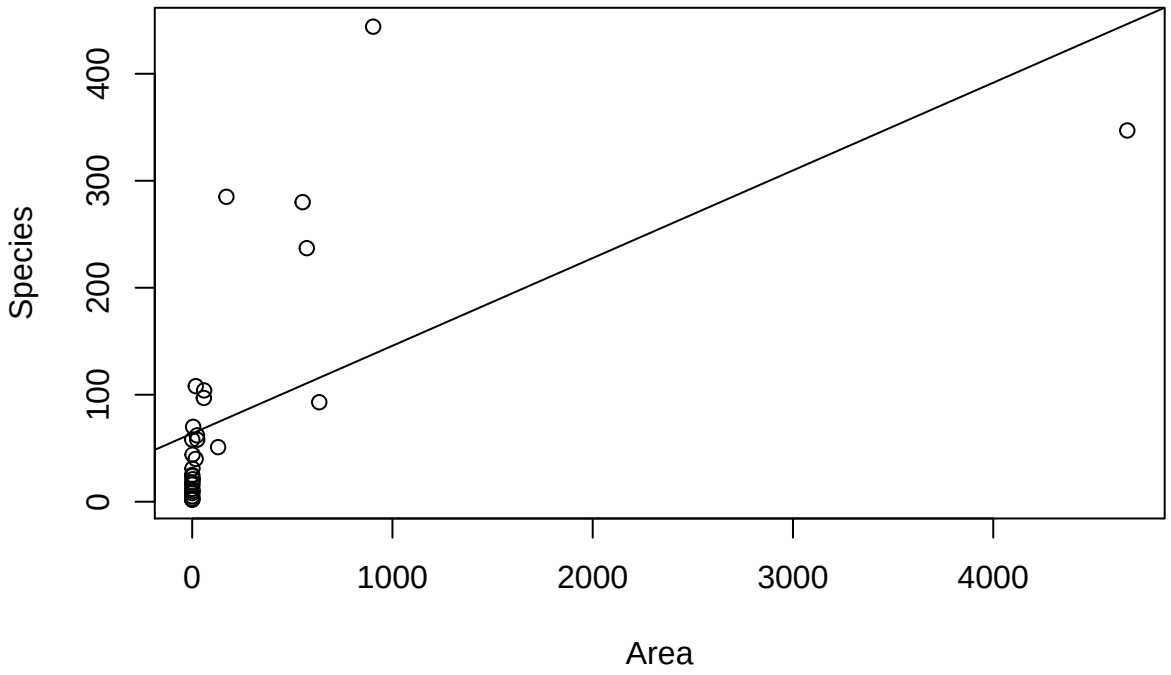
```
# biplot of PCA
pc = prcomp(gala, scale = T)
biplot(pc)
```



So we might guess that plant species numbers is best explained by island area and elevation. But

Single predictors

```
plot(Species ~ Area, gala)
g1a = lm(Species ~ Area, gala)
abline(g1a)
```




```
summary( g1a )
```

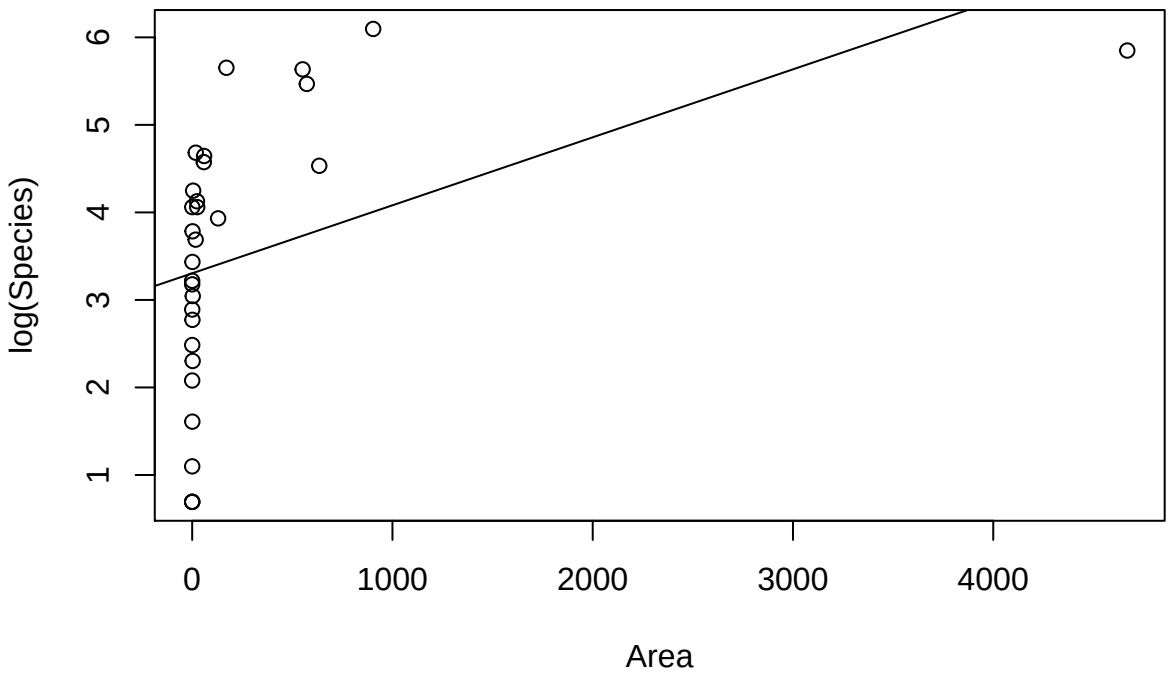
```
##
## Call:
## lm(formula = Species ~ Area, data = gala)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -99.495 -53.431 -29.045   3.423 306.137
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 63.78286   17.52442   3.640 0.001094 **
## Area         0.08196    0.01971   4.158 0.000275 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 91.73 on 28 degrees of freedom
## Multiple R-squared:  0.3817, Adjusted R-squared:  0.3596
## F-statistic: 17.29 on 1 and 28 DF,  p-value: 0.0002748
```

```
anova( g1a )
```

```
## Analysis of Variance Table
##
## Response: Species
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Area       1 145470  145470  17.288 0.0002748 ***
## Residuals 28 235611    8415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

There is obviously a correlation. But could we improve this fit?

```
plot(log(Species) ~ Area, gala)
g1a = lm(log(Species) ~ Area, gala)
abline(g1a)
```

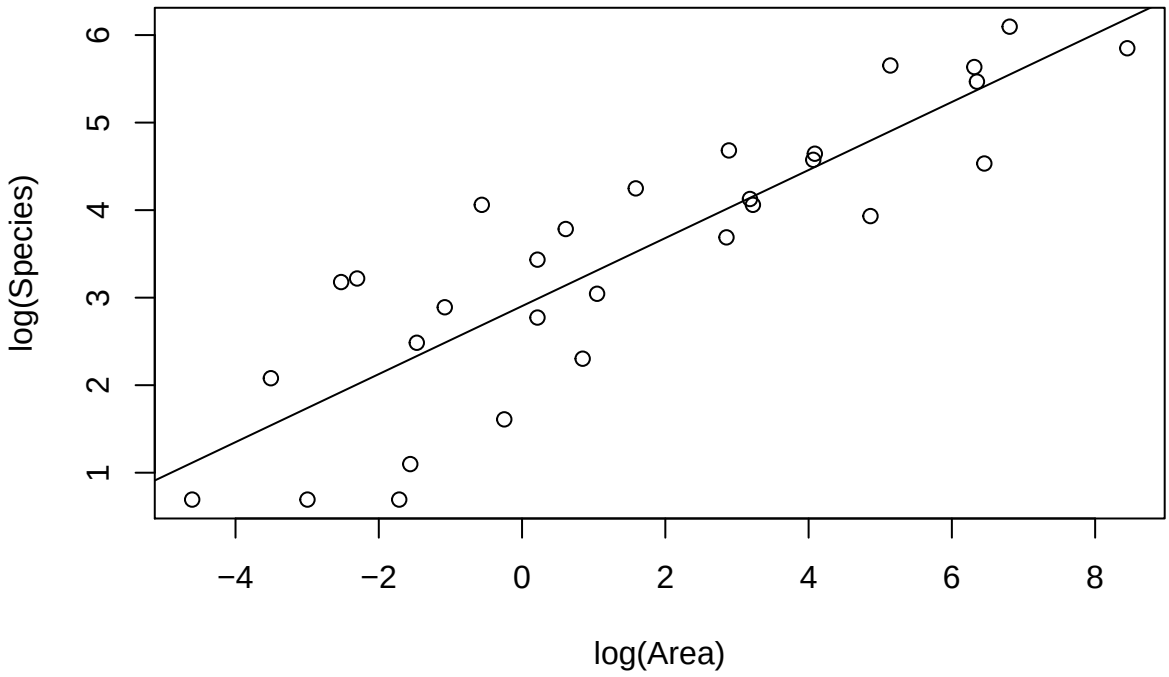



```
summary( g1a )
```

```
##
## Call:
## lm(formula = log(Species) ~ Area, data = gala)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6113 -0.9576  0.2499  0.9063  2.2154
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.3043356  0.2745303  12.036 1.39e-12 ***
## Area        0.0007768  0.0003088   2.515   0.0179 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.437 on 28 degrees of freedom
## Multiple R-squared:  0.1843, Adjusted R-squared:  0.1552
## F-statistic: 6.327 on 1 and 28 DF,  p-value: 0.01791
```

How about log transforming both?

```
plot(log(Species) ~ log(Area), gala)
g1a = lm(log(Species) ~ log(Area), gala)
abline(g1a)
```




```
summary( g1a )
```

```
##
## Call:
## lm(formula = log(Species) ~ log(Area), data = gala)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5442 -0.4001  0.0941  0.5449  1.3752
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.9037     0.1571  18.484  < 2e-16 ***
## log(Area)     0.3886     0.0416   9.342 4.23e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7842 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
```

Problems with residuals

We can see how log-transformation might help by studying residuals. For each x, we assume there exists y values that are normally distributed, whose mean are described by the regression function. [http://cnx.org/resources/ccc0cfb872dac9d2b3a3edb26247f9f1fa07d51e/lnregs_Facts_normal.png]

If so, the residuals (errors) are supposed to be **homogeneously* distributed around** `0` across the predictor range. There can be different types of deviations from this assumption.

- If the relationship between predictor and response are **non-linear**, this will create non-uniform residuals.
- You may have **autocorrelation**, where residuals are all positive and all negative for different ranges of

the predictor distribution.

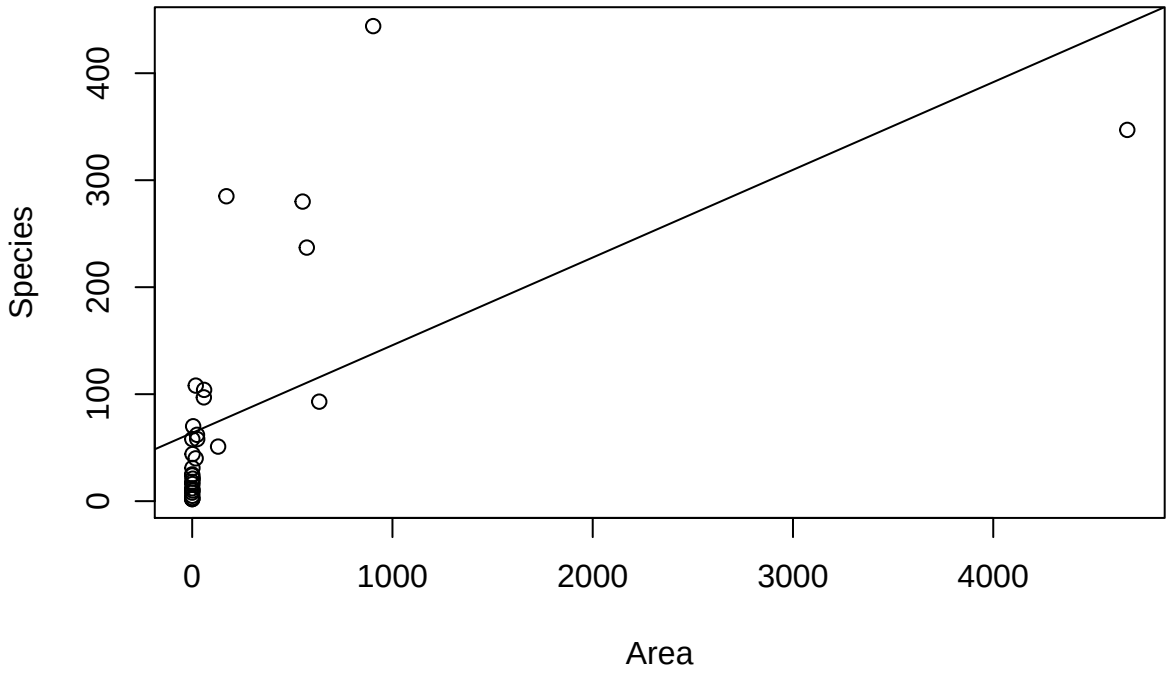
- You may have **heteroskedasticity**, which is when the residual variance is not constant.

[http://www.acastat.com/statbook/molsassump018.jpg]

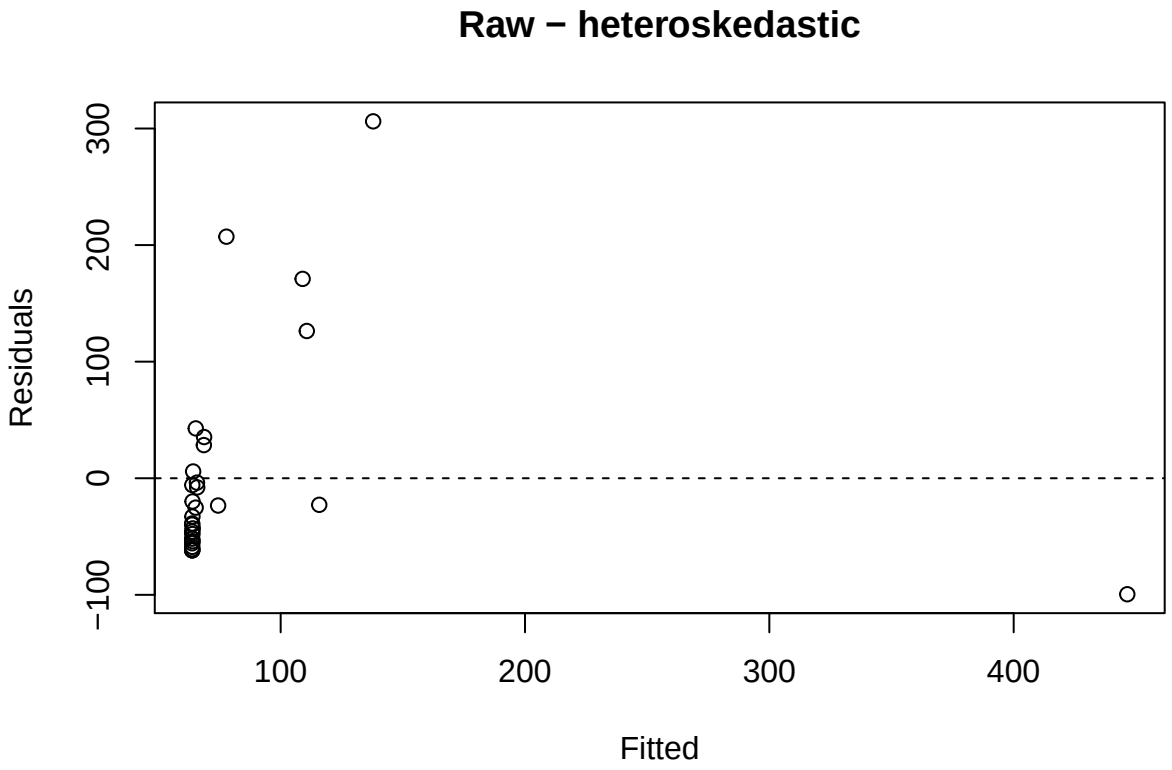
All these cases violate linear regression assumptions. See more here:

http://www.acastat.com/statbook/molsassumptions.htm

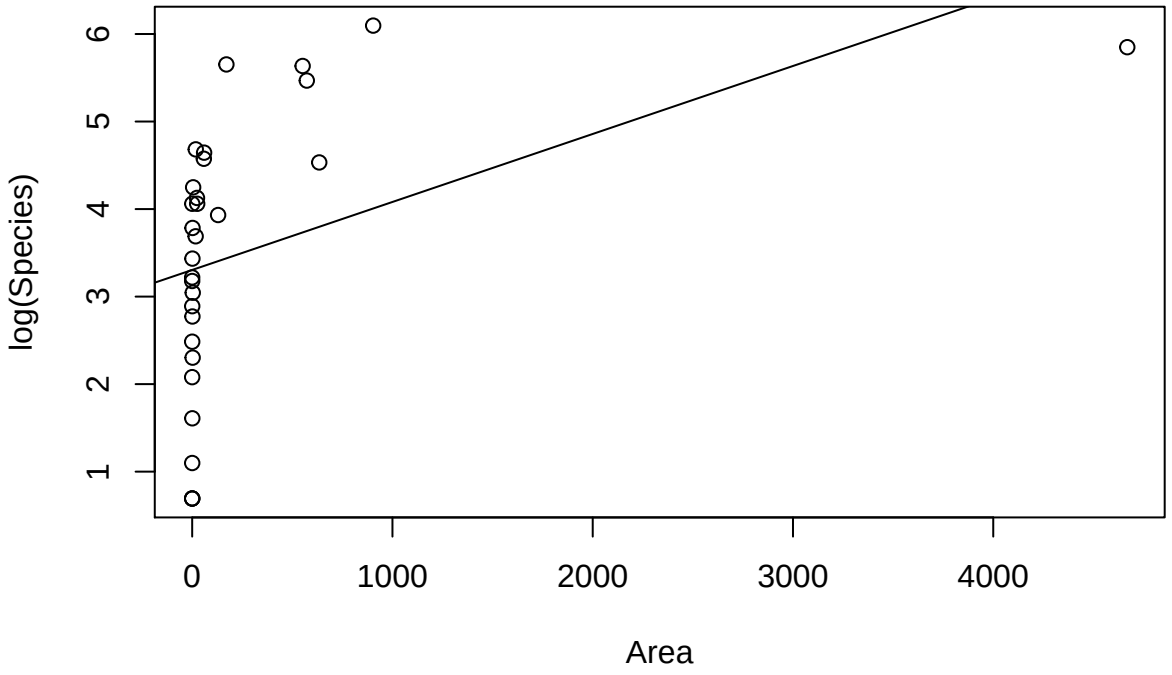
```
plot(Species ~ Area, gala)
g1a = lm(Species ~ Area, gala)
abline(g1a)
```



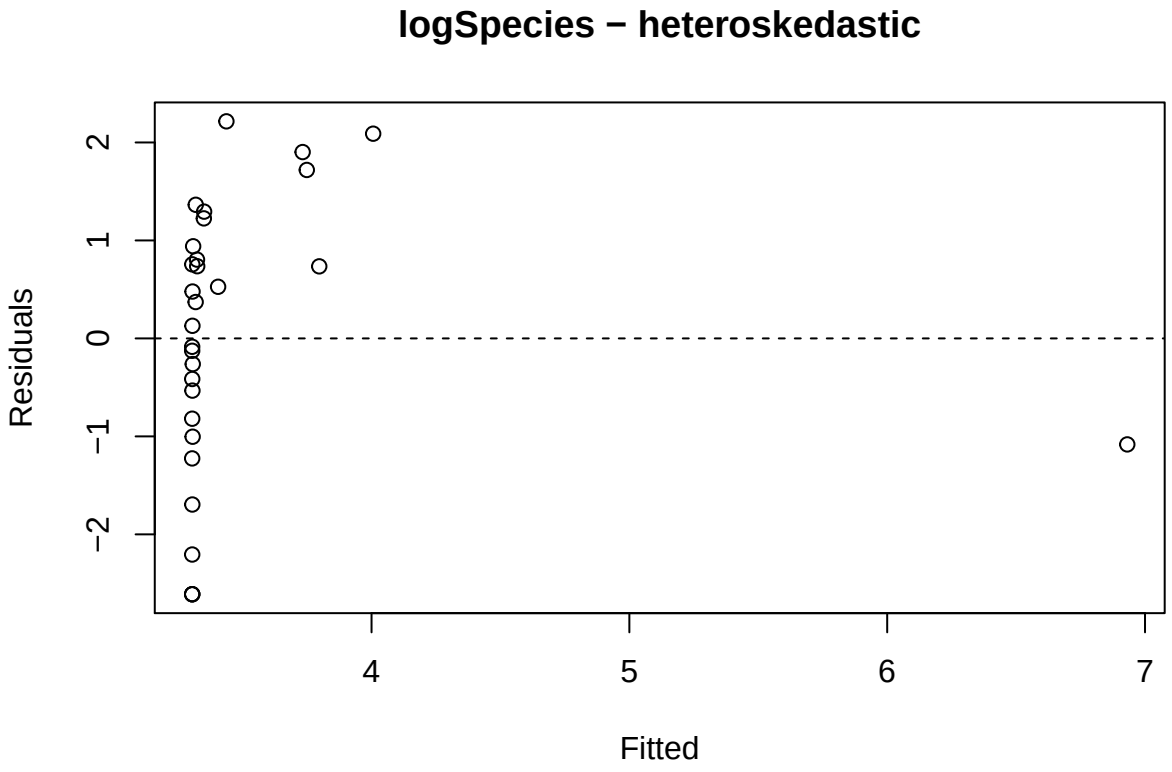

```
plot(g1a$fit, g1a$res, xlab="Fitted", ylab="Residuals", main="Raw - heteroskedastic")
abline(h=0, lty=2)
```



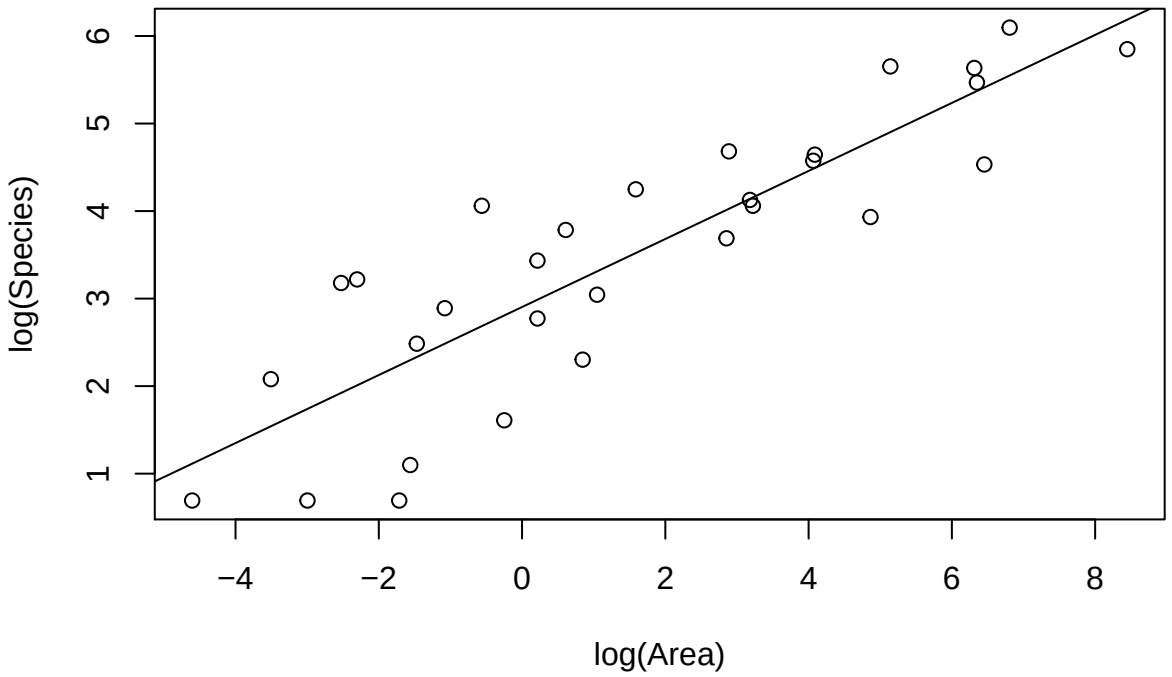

```
# log transform species
plot(log(Species) ~ Area, gala)
g1a = lm(log(Species) ~ Area, gala)
abline(g1a)
```



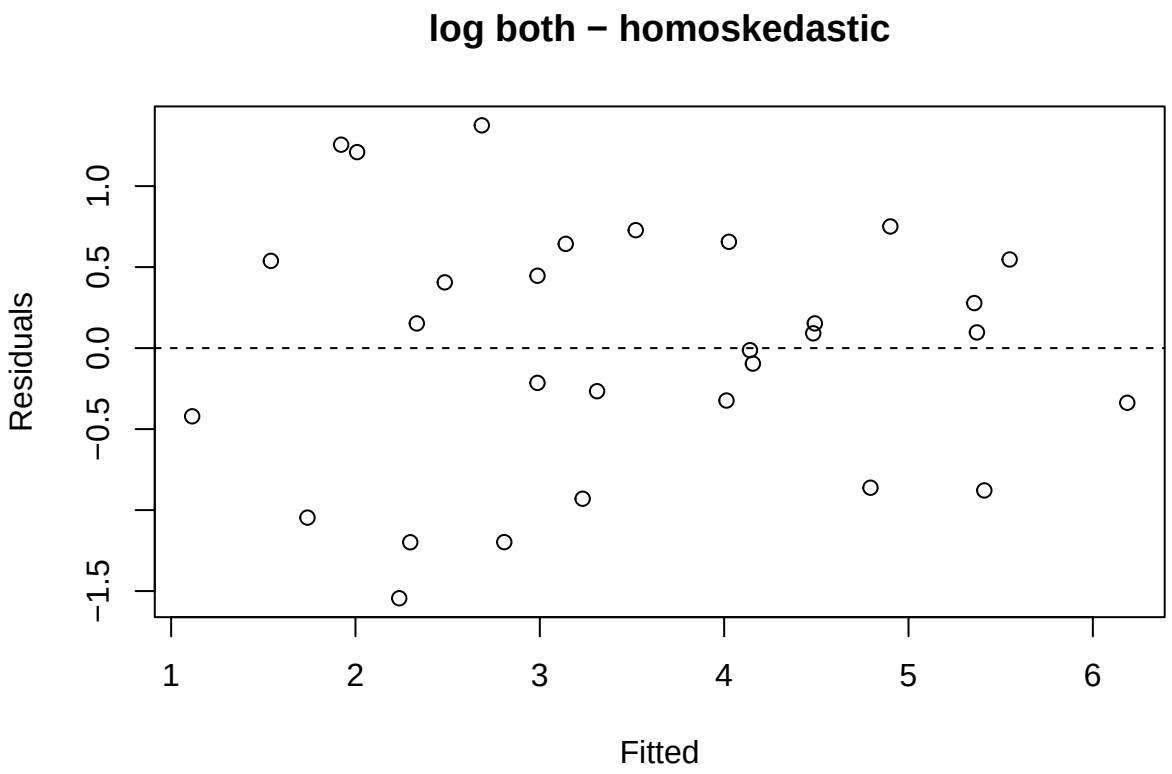

```
plot(g1a$fit, g1a$res, xlab="Fitted", ylab="Residuals", main="logSpecies - heteroskedastic")
abline(h=0, lty=2)
```




```
# log transform both variables
plot(log(Species) ~ log(Area), gala)
g1a = lm(log(Species) ~ log(Area), gala)
abline(g1a)
```




```
plot(g1a$fit, g1a$res, xlab="Fitted", ylab="Residuals", main="log both - homoskedastic")
abline(h=0, lty=2)
```



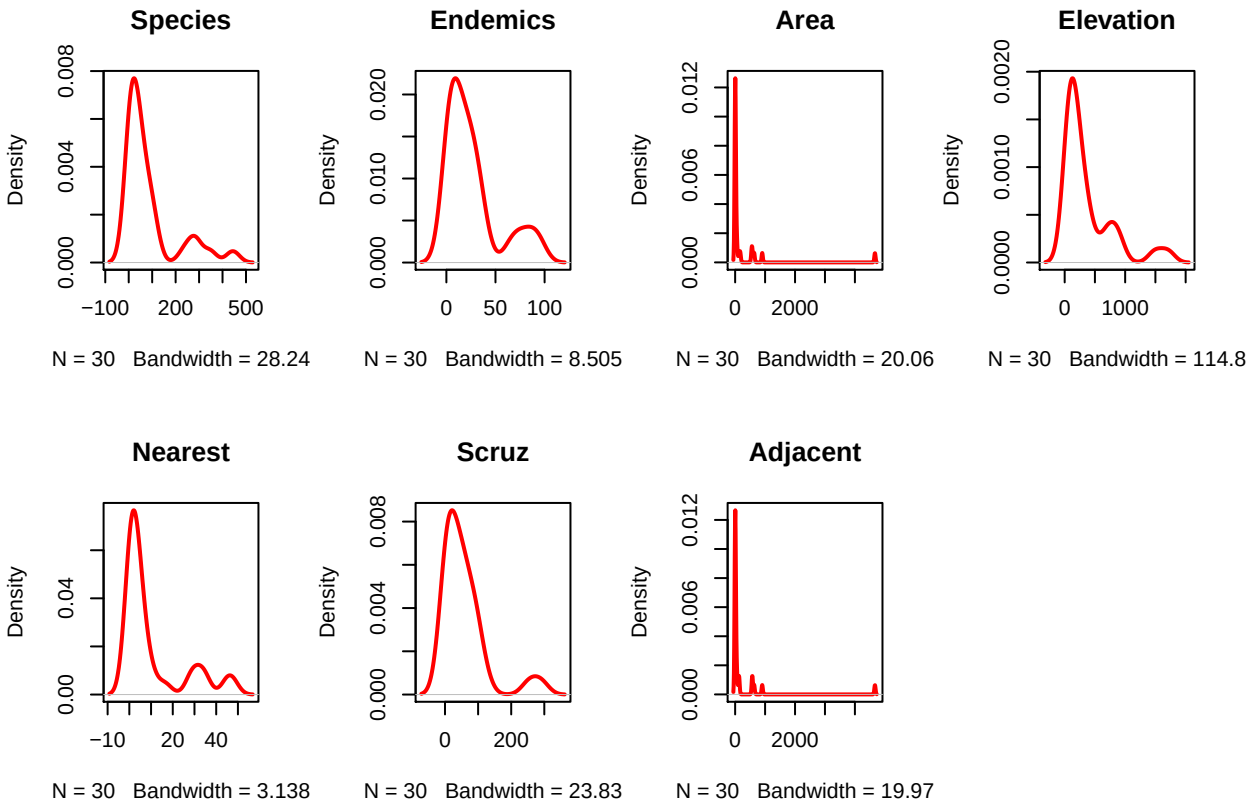

```
cor( abs(g1a$fit), g1a$res )
```

```
## [1] -5.75303e-17
```

Data transformation

Log transformation appears helpful for these variables. How about the others? The check, plot all variables in one frame:

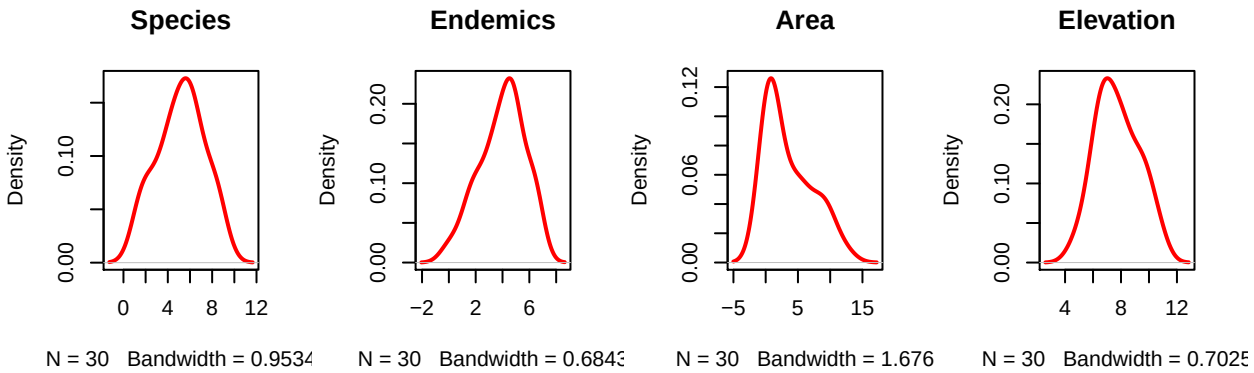
```
par(mfrow=c(2,4))
for (i in 1:ncol(gala)) {
  plot(density(gala[,i]), col=2, lwd=2, main=colnames(gala)[i] )
}
```

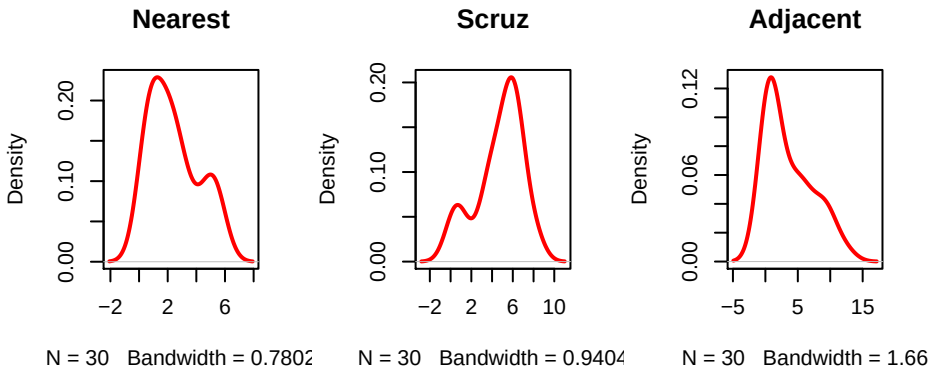


So all the data are right-skewed. What if we log2-transform all? It might be worth it, generally.

We should add `+1` to values as two columns contain 0s.

```
par(mfrow=c(2,4))
for (i in 1:ncol(gala)) {
  plot(density(log2(gala[,i]+1)), col=2, lwd=2, main=colnames(gala)[i] )
}
```



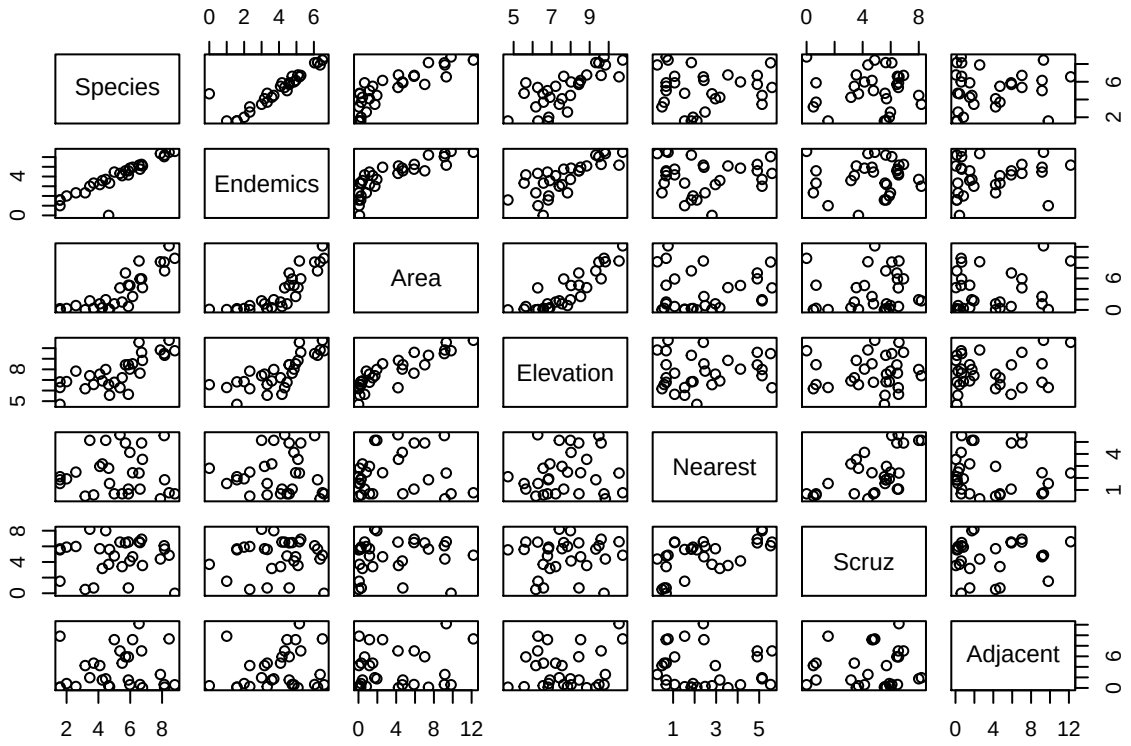


We might redefine a Galapagos dataset with all variables log2-transformed:

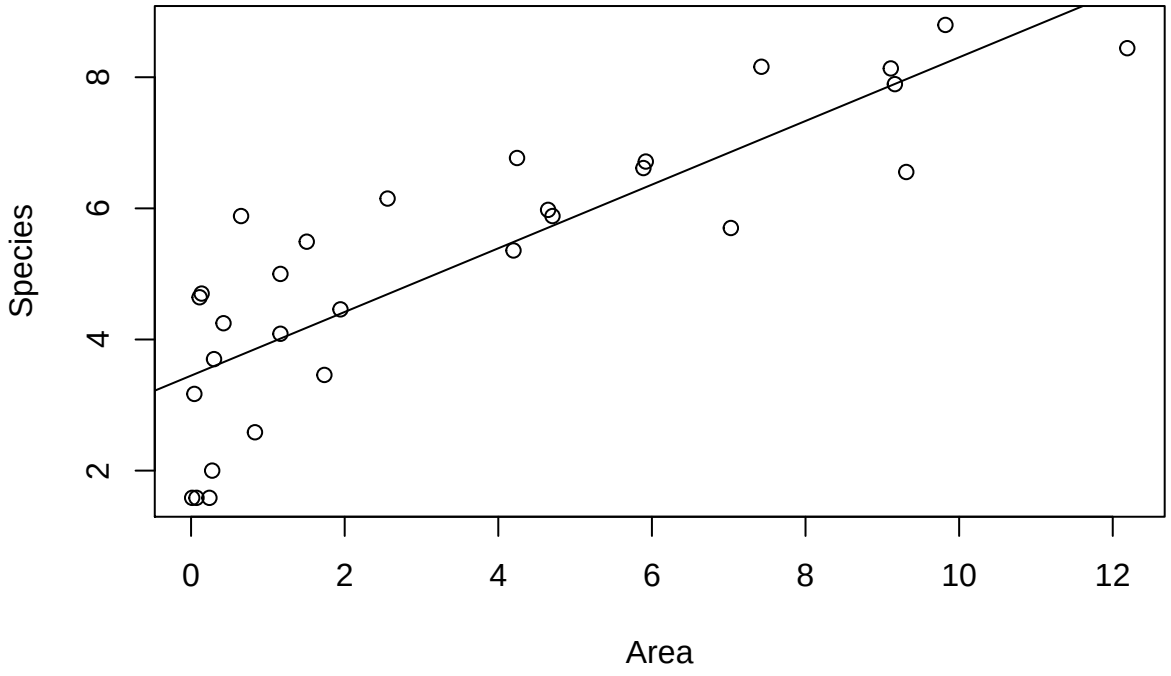
```
gala2 = log2(gala + 1)
str(gala2)
```

```
## 'data.frame':    30 obs. of  7 variables:
##  $ Species  : num  5.88 5 2 4.7 1.58 ...
##  $ Endemics : num  4.58 4.46 2 3.32 1 ...
##  $ Area     : num  4.7054 1.1635 0.275 0.1375 0.0704 ...
##  $ Elevation: num  8.44 6.78 6.85 5.55 6.29 ...
##  $ Nearest  : num  0.678 0.678 1.926 1.536 1.536 ...
##  $ Scruz    : num  0.678 4.771 5.9 5.597 1.536 ...
##  $ Adjacent : num  1.506 9.163 0.832 0.239 9.821 ...
```

```
plot(gala2)
```




```
plot(Species ~ Area, gala2)
g1a = lm(Species ~ Area, gala2)
abline(g1a)
```

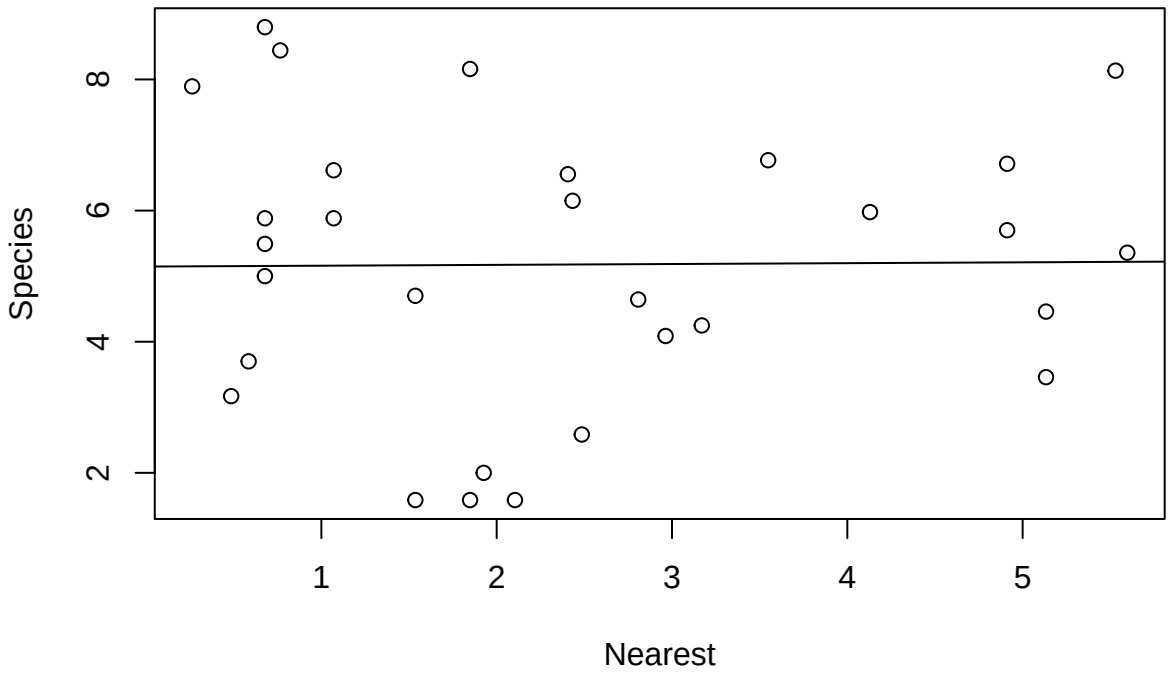



```
summary( g1a )
```

```
##
## Call:
## lm(formula = Species ~ Area, data = gala2)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9801 -0.9010  0.1277  0.8880  2.1176
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.44912    0.29093  11.856 1.98e-12 ***
## Area         0.48548    0.05734   8.466 3.31e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.135 on 28 degrees of freedom
## Multiple R-squared:  0.7191, Adjusted R-squared:  0.7091
## F-statistic: 71.68 on 1 and 28 DF,  p-value: 3.313e-09
```

```
plot(Species ~ Nearest, gala2)
g1n = lm(Species ~ Nearest, gala2)
abline(g1n)
```

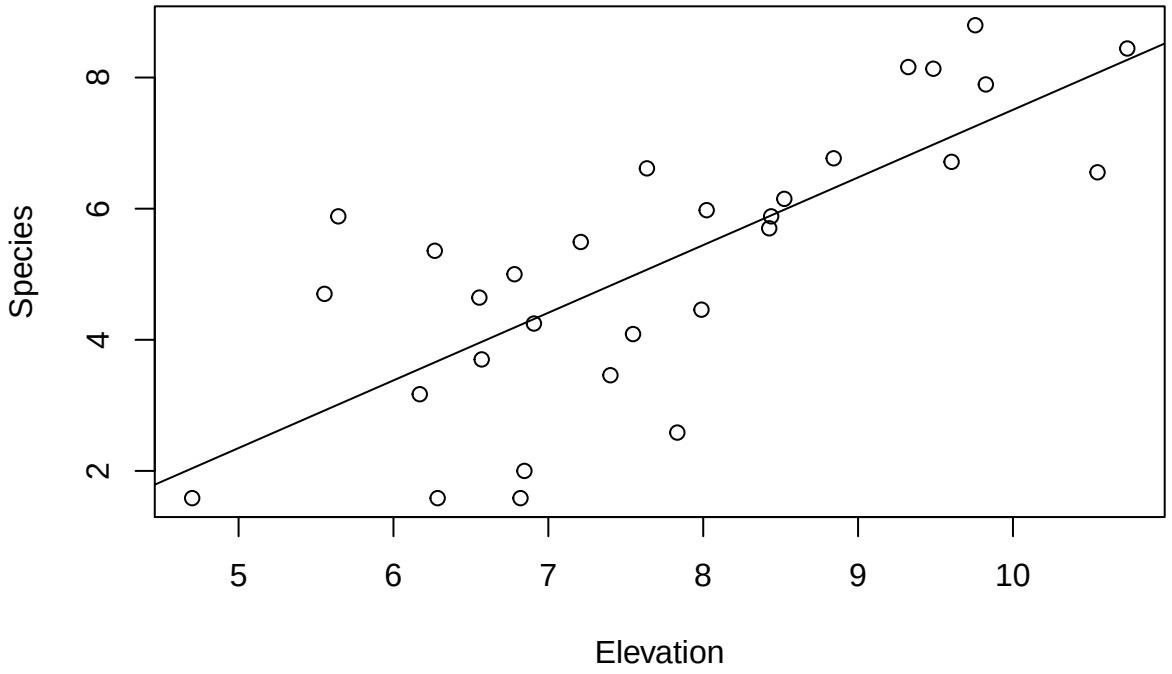



```
summary( g1n )
```

```
##
## Call:
## lm(formula = Species ~ Nearest, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5886 -1.3644  0.2378  1.4352  3.6425
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.14641    0.68687   7.493 3.68e-08 ***
## Nearest      0.01289    0.23236   0.055    0.956
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.142 on 28 degrees of freedom
## Multiple R-squared:  0.00011,    Adjusted R-squared:  -0.0356
## F-statistic: 0.003079 on 1 and 28 DF,  p-value: 0.9561
```

```
plot(Species ~ Elevation, gala2)
g1e = lm(Species ~ Elevation, gala2)
abline(g1e)
```




```
summary( g1e )
```

```
##
## Call:
## lm(formula = Species ~ Elevation, data = gala2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.68700 -0.78082  0.07536  0.85059  2.86925
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.810      1.333  -2.107   0.0442 *
## Elevation      1.032      0.169   6.104 1.39e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.403 on 28 degrees of freedom
## Multiple R-squared:  0.5709, Adjusted R-squared:  0.5556
## F-statistic: 37.25 on 1 and 28 DF,  p-value: 1.385e-06
```

Model comparison and predictor choice

So which variables to include in the model? We can start agnostically, by studying the full model, and removing unnecessary variables.

```
gfull = lm(Species ~ ., gala2)
summary(gfull)
```

```
##
## Call:
## lm(formula = Species ~ ., data = gala2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.11568 -0.41969 -0.09676  0.33919  3.15438
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.82181    1.54610   1.178 0.250714
## Endemics     0.82413    0.18131   4.545 0.000145 ***
## Area         0.18203    0.11296   1.611 0.120712
## Elevation   -0.02308    0.22881  -0.101 0.920516
## Nearest      0.01721    0.12167   0.141 0.888735
## Scruz       -0.06360    0.09262  -0.687 0.499157
## Adjacent    -0.03364    0.04698  -0.716 0.481220
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8971 on 23 degrees of freedom
## Multiple R-squared:  0.8559, Adjusted R-squared:  0.8183
## F-statistic: 22.76 on 6 and 23 DF,  p-value: 1.343e-08
```

This reminds us that `Endemics` is actually a variable dependent on, and correlated with `Species`. So, it is not really an independent predictor. We might consider removing it from the model altogether.

```
gfull1 = lm(Species ~ . - Endemics, gala2)
summary(gfull1)
```

```
##
## Call:
## lm(formula = Species ~ . - Endemics, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0530 -0.8083  0.1946  0.8819  2.3348
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.26110    2.04113   1.598  0.12320
## Area         0.46135    0.12783   3.609  0.00141 **
## Elevation    0.08707    0.30688   0.284  0.77904
## Nearest     -0.03309    0.16343  -0.203  0.84123
## Scruz       -0.04617    0.12482  -0.370  0.71472
## Adjacent    -0.02832    0.06335  -0.447  0.65881
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.21 on 24 degrees of freedom
## Multiple R-squared:  0.7264, Adjusted R-squared:  0.6694
## F-statistic: 12.74 on 5 and 24 DF,  p-value: 4.122e-06
```

How about removing `Elevation`?

```
gfull2 = lm(Species ~ . - Endemics - Elevation, gala2)
summary(gfull2)
```

```
##
## Call:
## lm(formula = Species ~ . - Endemics - Elevation, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0163 -0.8103  0.1729  0.9203  2.2543
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.81544    0.57995   6.579 6.83e-07 ***
## Area         0.49302    0.06115   8.062 2.04e-08 ***
## Nearest     -0.02697    0.15899  -0.170    0.867
## Scruz       -0.04902    0.12211  -0.401    0.691
## Adjacent    -0.02676    0.06194  -0.432    0.669
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.188 on 25 degrees of freedom
## Multiple R-squared:  0.7255, Adjusted R-squared:  0.6816
## F-statistic: 16.52 on 4 and 25 DF,  p-value: 9.657e-07
```

```
anova(gfull1, gfull2)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics
## Model 2: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     24 35.140
## 2     25 35.258 -1  -0.11787 0.0805  0.779
```

So adding `Elevation` does not explain any additional variation (further than what `Area` does). We could discard the former.

```
gfull3 = lm(Species ~ . - Endemics - Elevation - Nearest, gala2)
summary(gfull3)
```

```
##
## Call:
## lm(formula = Species ~ . - Endemics - Elevation - Nearest, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9869 -0.7847  0.1588  0.9151  2.3083
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.80415    0.56526   6.730 3.85e-07 ***
## Area         0.49245    0.05991   8.220 1.06e-08 ***
## Scruz       -0.06101    0.09771  -0.624    0.538
## Adjacent    -0.02553    0.06035  -0.423    0.676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.165 on 26 degrees of freedom
## Multiple R-squared:  0.7252, Adjusted R-squared:  0.6935
## F-statistic: 22.87 on 3 and 26 DF,  p-value: 1.842e-07
```

```
anova(gfull3, gfull2)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation - Nearest
## Model 2: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     26 35.298
## 2     25 35.258  1  0.040578 0.0288 0.8667
```

```
# and this comparison is also not significant, as would be expected
anova(gfull3, gfull1)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation - Nearest
## Model 2: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     26 35.298
## 2     24 35.140  2   0.15845 0.0541 0.9474
```

```
gfull4 = lm(Species ~ . - Endemics - Elevation - Nearest - Adjacent, gala2)
summary(gfull4)
```

```
##
## Call:
## lm(formula = Species ~ . - Endemics - Elevation - Nearest - Adjacent,
##     data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0856 -0.8571  0.1855  0.8759  2.2375
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.73068    0.52966   7.043 1.43e-07 ***
## Area         0.48806    0.05810   8.401 5.18e-09 ***
## Scruz       -0.06149    0.09621  -0.639    0.528
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.147 on 27 degrees of freedom
## Multiple R-squared:  0.7233, Adjusted R-squared:  0.7028
## F-statistic: 35.29 on 2 and 27 DF,  p-value: 2.933e-08
```

```
anova(gfull4, gfull3)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation - Nearest - Adjacent
## Model 2: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation - Nearest
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     27 35.541
## 2     26 35.298  1   0.24288 0.1789 0.6758
```

```
# neither is Scruz significant
# how about scruz+area vs area only?
anova(gfull4, g1a)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics - Elevation - Nearest - Adjacent
## Model 2: Species ~ Area
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     27 35.541
## 2     28 36.079 -1   -0.5377 0.4085 0.5281
```

```
# is the area only model as good as the full model?
anova(gfull1, g1a)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ (Endemics + Area + Elevation + Nearest + Scruz + Adjacent) -
##     Endemics
## Model 2: Species ~ Area
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     24 35.140
## 2     28 36.079 -4  -0.93903 0.1603 0.9563
```

So our best model is that explained by `Area`.

Correlated variables

Regression also assumes no (at least limited) correlation between predictors. What if we had a perfectly correlated additional variable?

```
# a toy predictor correlated with Area
NewPred = 100 - gala2$Area
gfull1x = lm(Species ~ NewPred + . - Endemics, gala2)
summary( gfull1x )
```

```
##
```

```
## Call:
## lm(formula = Species ~ NewPred + . - Endemics, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0530 -0.8083  0.1946  0.8819  2.3348
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 49.39604   14.46958   3.414  0.00228 **
## NewPred     -0.46135    0.12783  -3.609  0.00141 **
## Area             NA         NA      NA       NA
## Elevation    0.08707    0.30688   0.284  0.77904
## Nearest     -0.03309    0.16343  -0.203  0.84123
## Scruz       -0.04617    0.12482  -0.370  0.71472
## Adjacent    -0.02832    0.06335  -0.447  0.65881
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.21 on 24 degrees of freedom
## Multiple R-squared:  0.7264, Adjusted R-squared:  0.6694
## F-statistic: 12.74 on 5 and 24 DF,  p-value: 4.122e-06
```

Note the mention: "not defined because of singularities".

What if we added a very small bit of noise, but the correlation remains? This also disrupts the analysis:

```
# a toy variable correlated with Area, with some noise
set.seed(1)
NewPred = 100 - gala2$Area + rnorm(nrow(gala2), mean=0, sd=0.00001)
gfull1x2 = lm(Species ~ NewPred + . - Endemics, gala2)
summary( gfull1x2 )
```

```
##
## Call:
## lm(formula = Species ~ NewPred + . - Endemics, data = gala2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0661 -0.7581  0.1223  0.8384  2.5440
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.161e+06  2.812e+06  -0.768    0.450
## NewPred      2.161e+04  2.812e+04   0.768    0.450
## Area         2.161e+04  2.812e+04   0.768    0.450
## Elevation    1.964e-01  3.407e-01   0.577    0.570
## Nearest     -7.685e-02  1.744e-01  -0.441    0.664
## Scruz       -3.059e-02  1.275e-01  -0.240    0.813
## Adjacent    -3.839e-02  6.523e-02  -0.589    0.562
##
## Residual standard error: 1.22 on 23 degrees of freedom
## Multiple R-squared:  0.7333, Adjusted R-squared:  0.6637
## F-statistic: 10.54 on 6 and 23 DF,  p-value: 1.208e-05
```

The savings dataset

Economic and demographic variables for different countries between 1960-1970, collected to study savings rate (`sr` - personal saving divided by disposable income). `pop15`: percent population under age of 15. `pop75`: percent population over age of 75. `dpi`: per-capita disposable income. `ddpi`: percent growth rate of dpi.

```
str(savings)
```

```
## 'data.frame':    50 obs. of  5 variables:
##  $ sr   : num  11.43 12.07 13.17 5.75 12.88 ...
##  $ pop15: num  29.4 23.3 23.8 41.9 42.2 ...
##  $ pop75: num  2.87 4.41 4.43 1.67 0.83 2.85 1.34 0.67 1.06 1.14 ...
##  $ dpi  : num  2330 1508 2108 189 728 ...
##  $ ddpi : num  2.87 3.93 3.82 0.22 4.56 2.43 2.67 6.51 3.08 2.8 ...
```

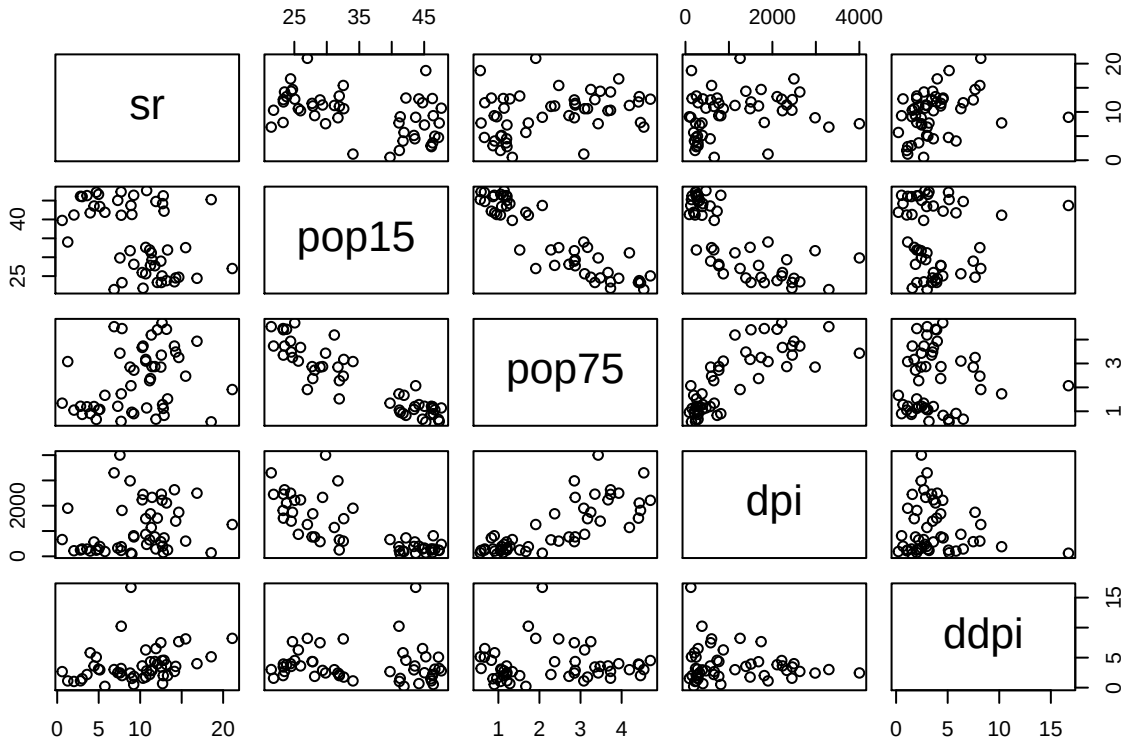
```
head(savings)
```

```
##              sr pop15 pop75     dpi ddpi
## Australia 11.43 29.35  2.87 2329.68 2.87
## Austria   12.07 23.32  4.41 1507.99 3.93
## Belgium   13.17 23.80  4.43 2108.47 3.82
## Bolivia    5.75 41.89  1.67  189.13 0.22
## Brazil    12.88 42.19  0.83  728.47 4.56
## Canada     8.79 31.72  2.85 2982.88 2.43
```

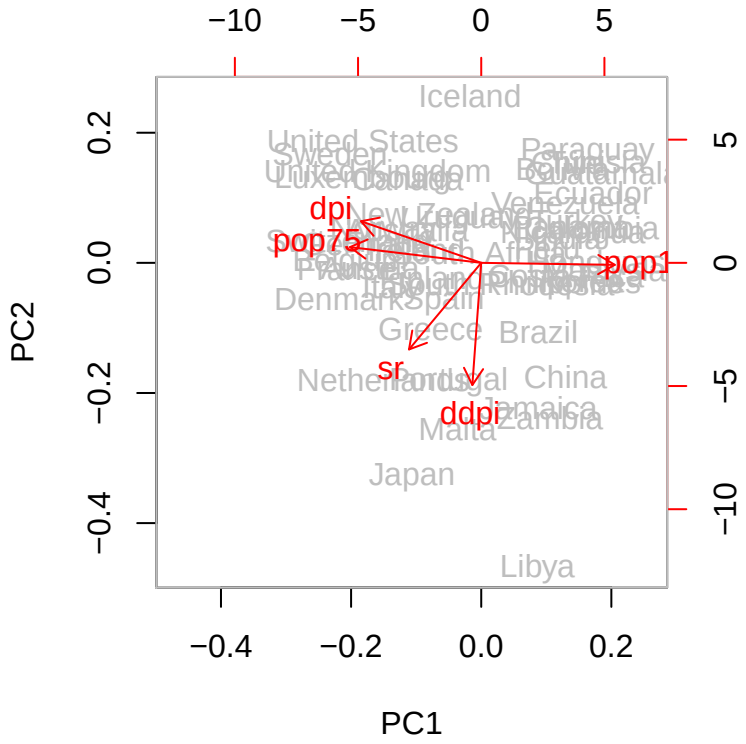
```
summary(savings)
```

```
##        sr             pop15           pop75            dpi
##  Min.   : 0.600   Min.   :21.44   Min.   :0.560   Min.   :  88.94
##  1st Qu.: 6.970   1st Qu.:26.21   1st Qu.:1.125   1st Qu.: 288.21
##  Median :10.510   Median :32.58   Median :2.175   Median : 695.66
##  Mean   : 9.671   Mean   :35.09   Mean   :2.293   Mean   :1106.76
##  3rd Qu.:12.617   3rd Qu.:44.06   3rd Qu.:3.325   3rd Qu.:1795.62
##  Max.   :21.100   Max.   :47.64   Max.   :4.700   Max.   :4001.89
##       ddpi
##  Min.   : 0.220
##  1st Qu.: 2.002
##  Median : 3.000
##  Mean   : 3.758
##  3rd Qu.: 4.478
##  Max.   :16.710
```

```
pairs(savings)
```




```
biplot( prcomp( savings, scale = T), col=c('grey', 'red') )
```



What does the PCA plot and the variable vectors suggest?

The full model

Now check the full model:

```
g = lm(sr ~ ., data=savings)
g
```

```
##
## Call:
## lm(formula = sr ~ ., data = savings)
##
## Coefficients:
## (Intercept)        pop15        pop75          dpi         ddpi
##  28.5660865   -0.4611931   -1.6914977   -0.0003369    0.4096949
```

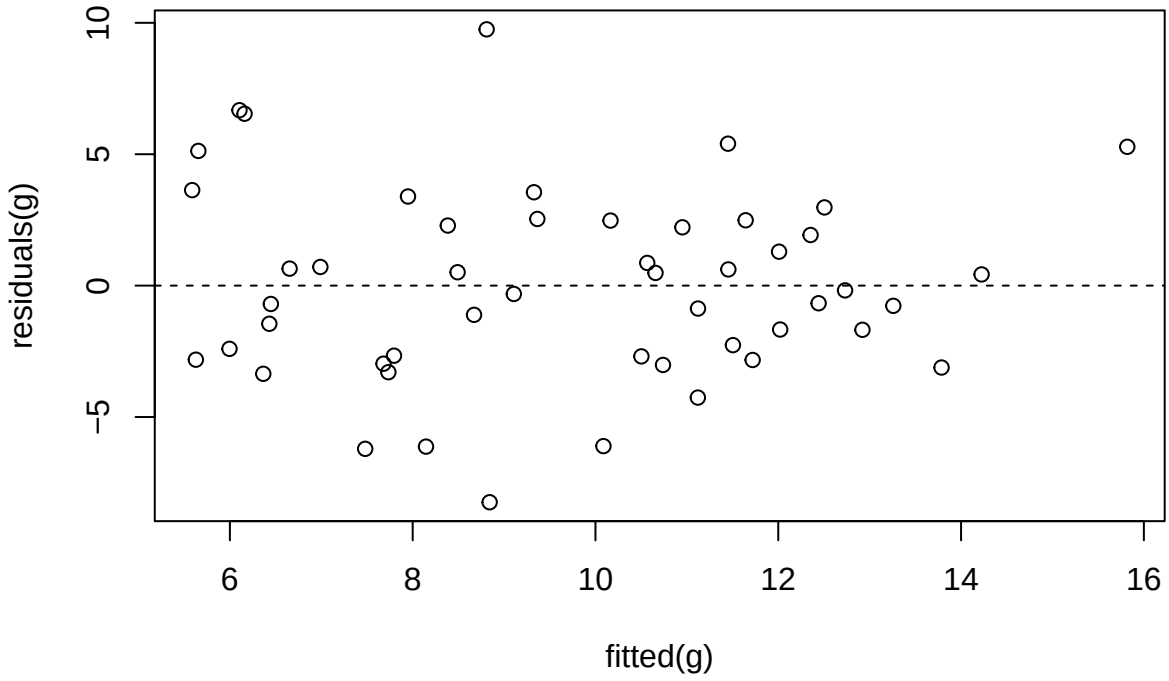
```
summary( g )
```

```
##
## Call:
## lm(formula = sr ~ ., data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2422 -2.6857 -0.2488  2.4280  9.7509
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 28.5660865  7.3545161   3.884 0.000334 ***
## pop15       -0.4611931  0.1446422  -3.189 0.002603 **
## pop75       -1.6914977  1.0835989  -1.561 0.125530
## dpi         -0.0003369  0.0009311  -0.362 0.719173
## ddpi         0.4096949  0.1961971   2.088 0.042471 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.803 on 45 degrees of freedom
## Multiple R-squared:  0.3385, Adjusted R-squared:  0.2797
## F-statistic: 5.756 on 4 and 45 DF,  p-value: 0.0007904
```

What can you tell about the factors? How does the young population frequency affect savings? Or the growth in income?

Study residuals

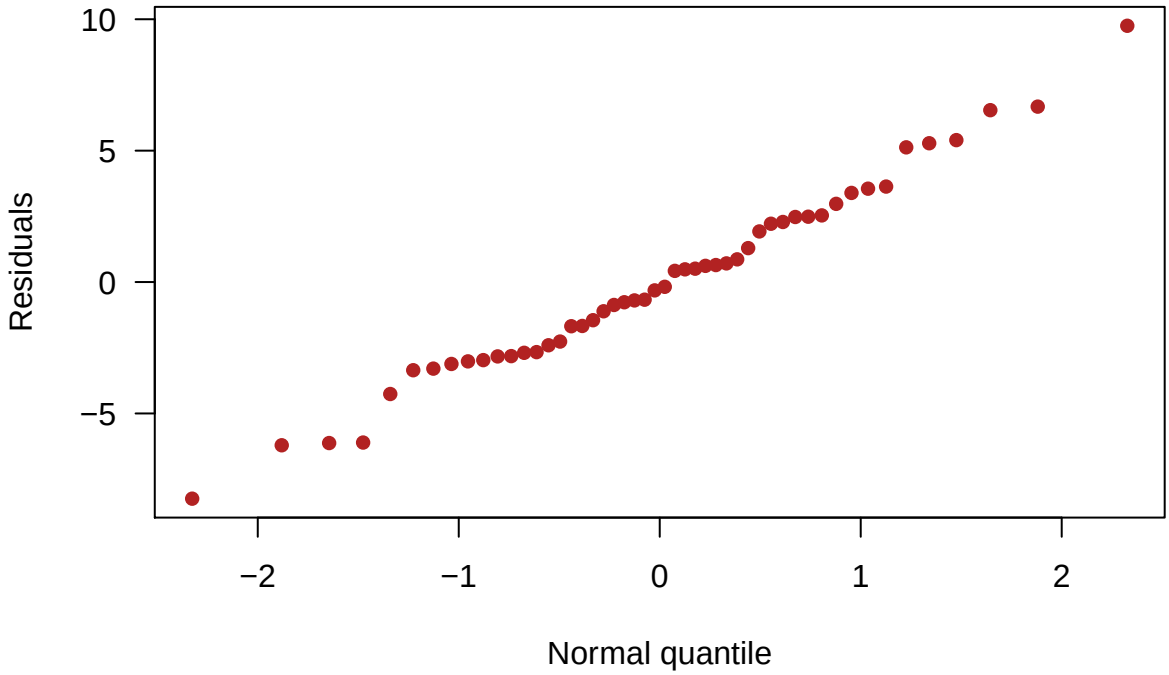
```
plot( residuals(g) ~ fitted(g) )
abline(h=0, lty=2)
```




```
cor( abs(residuals(g)), fitted(g) )
```

```
## [1] -0.2404972
```

```
qqnorm(residuals(g), pch = 16, col = "firebrick",
    las = 1, ylab = "Residuals", xlab = "Normal quantile", main = "")
```



ANOVA

Looks OK. Now study the ANOVA table:

```
anova( g )
```

```
## Analysis of Variance Table
##
## Response: sr
##           Df Sum Sq Mean Sq F value    Pr(>F)
## pop15      1 204.12 204.118 14.1157 0.0004922 ***
## pop75      1  53.34  53.343  3.6889 0.0611255 .
## dpi        1  12.40  12.401  0.8576 0.3593551
## ddpi       1  63.05  63.054  4.3605 0.0424711 *
## Residuals 45 650.71  14.460
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Let's remember how the ANOVA table is calculated:

```
ss_res = sum( g$res ^ 2 ) # residual sum of squares
ss_res
```

```
## [1] 650.713
```

```
ss_tot = sum( ( savings$sr - mean(savings$sr) ) ^ 2 ) # total sum of squares
ss_tot
```

```
## [1] 983.6283
```

```
ss_reg = ss_tot - ss_res
df_reg = 4  # number of predictor parameters
df_res = length(savings$sr) - ( df_reg + 1 )
df_res
```

```
## [1] 45
```

```
fval = (ss_reg/df_reg)/(ss_res/df_res)
fval
```

```
## [1] 5.755681
```

```
pf( fval,
        df1 = df_reg,
        df2 = df_res, lower.tail = F)
```

```
## [1] 0.0007903779
```

Compare models

```
summary(g)
```

```
##
## Call:
## lm(formula = sr ~ ., data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2422 -2.6857 -0.2488  2.4280  9.7509
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 28.5660865  7.3545161   3.884 0.000334 ***
## pop15       -0.4611931  0.1446422  -3.189 0.002603 **
## pop75       -1.6914977  1.0835989  -1.561 0.125530
## dpi         -0.0003369  0.0009311  -0.362 0.719173
## ddpi         0.4096949  0.1961971   2.088 0.042471 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.803 on 45 degrees of freedom
## Multiple R-squared:  0.3385, Adjusted R-squared:  0.2797
## F-statistic: 5.756 on 4 and 45 DF,  p-value: 0.0007904
```

Let's remove `dpi`, the least significant:

```
g2 = lm(sr ~ . - dpi, data=savings)
summary( g2 )
```

```
##
## Call:
## lm(formula = sr ~ . - dpi, data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2539 -2.6159 -0.3913  2.3344  9.7070
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  28.1247     7.1838   3.915 0.000297 ***
## pop15        -0.4518     0.1409  -3.206 0.002452 **
## pop75        -1.8354     0.9984  -1.838 0.072473 .
## ddpi          0.4278     0.1879   2.277 0.027478 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.767 on 46 degrees of freedom
## Multiple R-squared:  0.3365, Adjusted R-squared:  0.2933
## F-statistic: 7.778 on 3 and 46 DF,  p-value: 0.0002646
```

```
anova(g, g2)
```

```
## Analysis of Variance Table
##
## Model 1: sr ~ pop15 + pop75 + dpi + ddpi
## Model 2: sr ~ (pop15 + pop75 + dpi + ddpi) - dpi
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     45 650.71
## 2     46 652.61 -1   -1.8932 0.1309 0.7192
```

The model now becomes more significant than the full model. Removing an ineffective predictor improves the fit (you don't pay the penalty of having an extra factor). The "Adjusted R-squared" is also higher.

So we can remove `dpi`, as we predicted from `biplot`:

How about `pop75`?

```
g3 = lm(sr ~ . - dpi - pop75, data=savings)
summary( g3 )
```

```
##
## Call:
## lm(formula = sr ~ . - dpi - pop75, data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5831 -2.8632  0.0453  2.2273 10.4753
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 15.59958    2.33439   6.682 2.48e-08 ***
## pop15       -0.21638    0.06033  -3.586 0.000796 ***
## ddpi         0.44283    0.19240   2.302 0.025837 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.861 on 47 degrees of freedom
## Multiple R-squared:  0.2878, Adjusted R-squared:  0.2575
## F-statistic: 9.496 on 2 and 47 DF,  p-value: 0.0003438
```

```
anova(g2, g3)
```

```
## Analysis of Variance Table
##
## Model 1: sr ~ (pop15 + pop75 + dpi + ddpi) - dpi
## Model 2: sr ~ (pop15 + pop75 + dpi + ddpi) - dpi - pop75
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     46 652.61
## 2     47 700.55 -1   -47.946 3.3795 0.07247 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Perhaps not worth removing `pop75`. It is marginally better to keep it than remove.

Permutation test

Can we test the significance of a predictor without using the F distribution? Yes, we could randomize the predictor values and calculate the R-squared (or any related statistic). E.g. let's do this for `pop15`:

```
g2 = lm(sr ~ . - dpi, data=savings)
summary(g2)
```

```
##
## Call:
## lm(formula = sr ~ . - dpi, data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2539 -2.6159 -0.3913  2.3344  9.7070
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  28.1247     7.1838   3.915 0.000297 ***
## pop15        -0.4518     0.1409  -3.206 0.002452 **
## pop75        -1.8354     0.9984  -1.838 0.072473 .
```

```
## ddpi          0.4278     0.1879   2.277 0.027478 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.767 on 46 degrees of freedom
## Multiple R-squared:  0.3365, Adjusted R-squared:  0.2933
## F-statistic: 7.778 on 3 and 46 DF,  p-value: 0.0002646
```

```
summary(g2)$r.squared
```

```
## [1] 0.3365317
```

```
set.seed(1)
random_pop15 = sample(savings$pop15)
rg2 = lm(sr ~ random_pop15 + pop75 + ddpi, data=savings)
summary( rg2 )
```

```
##
## Call:
## lm(formula = sr ~ random_pop15 + pop75 + ddpi, data = savings)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.326 -3.473  0.480  2.276 10.082
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.59653    2.61974   2.900  0.00571 **
## random_pop15 -0.06432    0.06675  -0.964  0.34024
## pop75         1.04285    0.45769   2.279  0.02739 *
## ddpi          0.51638    0.21256   2.429  0.01909 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.125 on 46 degrees of freedom
## Multiple R-squared:  0.2044, Adjusted R-squared:  0.1525
## F-statistic: 3.939 on 3 and 46 DF,  p-value: 0.01392
```

```
g0 = lm(sr ~ pop75 + ddpi, data=savings)
summary( g0 )
```

```
##
## Call:
## lm(formula = sr ~ pop75 + ddpi, data = savings)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.0223 -3.2949  0.0889  2.4570 10.1069
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.4695     1.4101   3.879 0.000325 ***
## pop75         1.0726     0.4563   2.351 0.022992 *
## ddpi          0.4636     0.2052   2.259 0.028562 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 4.122 on 47 degrees of freedom
## Multiple R-squared:  0.1883, Adjusted R-squared:  0.1538
## F-statistic: 5.452 on 2 and 47 DF,  p-value: 0.007423
```

Note that `rg2` here explains *more* variance than `g0`, i.e. NOT having the factor. But `rg2` is *less* significant compared to `g0`.

Compare the two models:

```
anova( rg2, g0 )
```

```
## Analysis of Variance Table
##
## Model 1: sr ~ random_pop15 + pop75 + ddpi
## Model 2: sr ~ pop75 + ddpi
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     46 782.6
## 2     47 798.4 -1     -15.8 0.9287 0.3402
```

Now prepare for permutation:

```
random_pop15 = sample(savings$pop15)
rg2 = lm(sr ~ random_pop15 + pop75 + ddpi, data=savings)
summary( rg2 )$r.squared
```

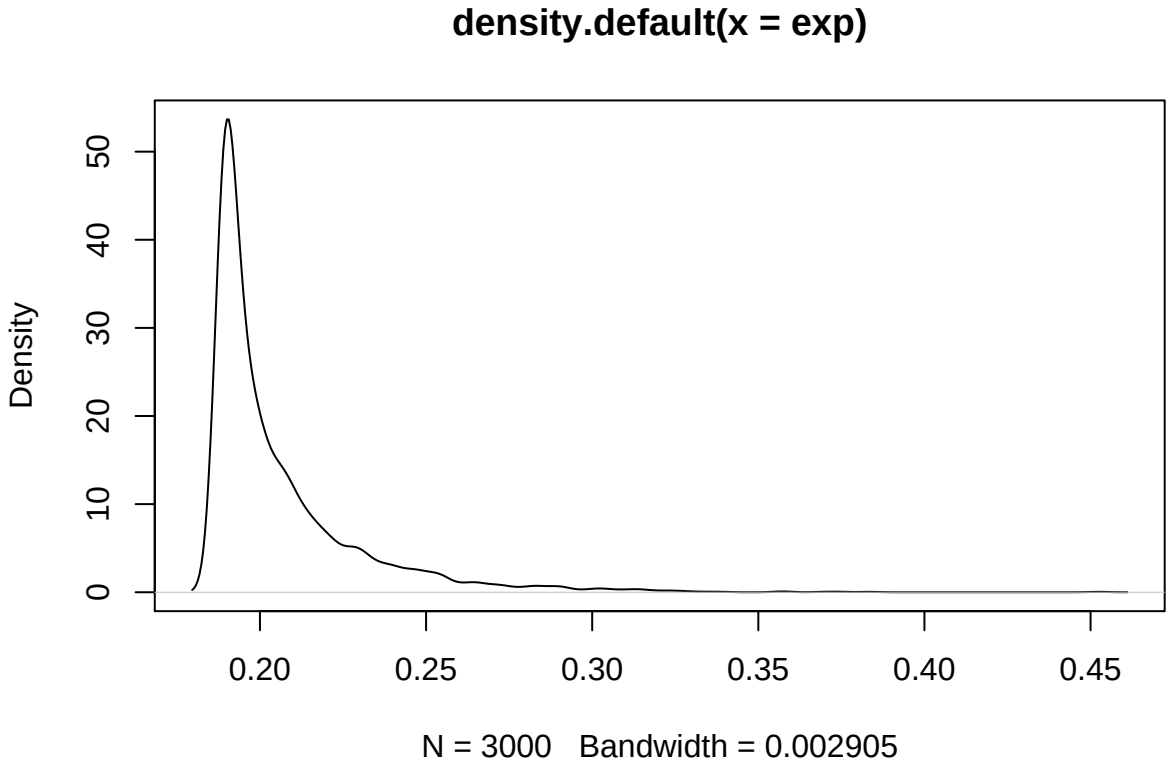
```
## [1] 0.1899419
```

```
random_pop15 = sample(savings$pop15)
rg2 = lm(sr ~ random_pop15 + pop75 + ddpi, data=savings)
summary( rg2 )$r.squared
```

```
## [1] 0.2118996
```

```
# seems to work. so we can loop

exp = sapply(1:3000, function(i) {
    random_pop15 = sample(savings$pop15)
    rg2 = lm(sr ~ random_pop15 + pop75 + ddpi, data=savings)
    summary( rg2 )$r.squared
})
plot(density(exp))
```




```
obs = summary( g2 )$r.squared
obs
```

```
## [1] 0.3365317
```

```
#p-value
mean( exp >= obs  )
```

```
## [1] 0.002333333
```

Generalised linear models

Time-dependent change may be subject to autocorrelation, where residuals tend to be positive or negative during subsequent years:

```
head(longley)
```

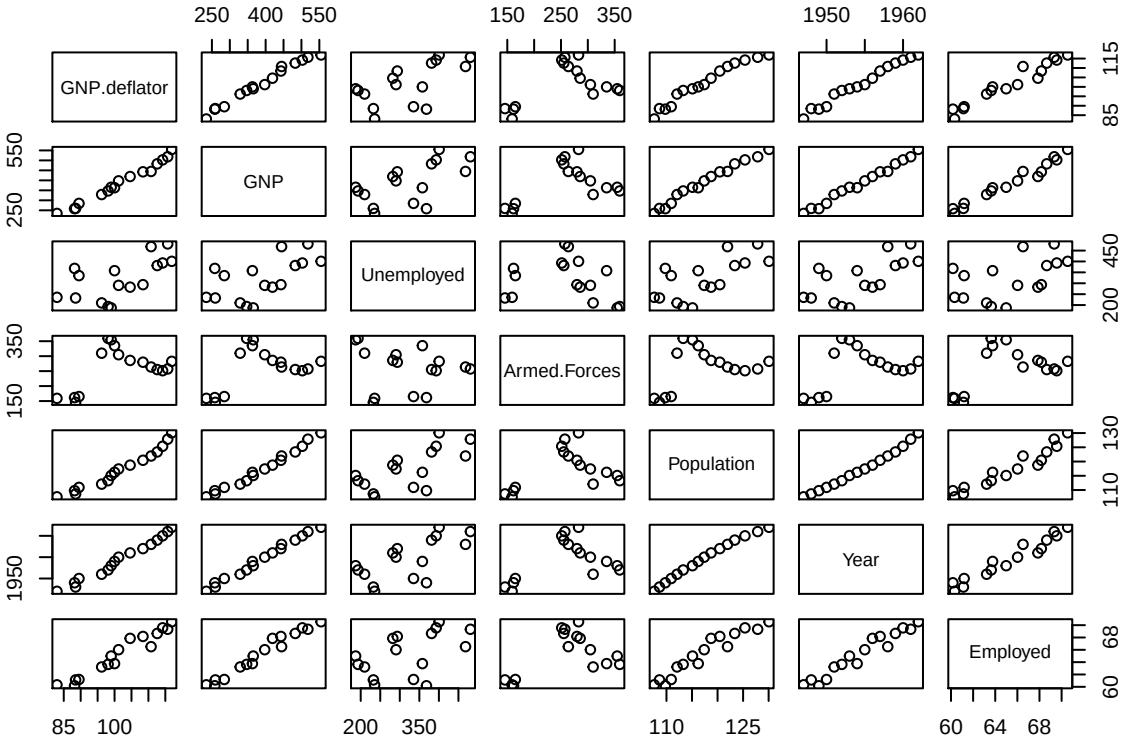
```
##      GNP.deflator     GNP Unemployed Armed.Forces Population Year Employed
## 1947         83.0 234.289      235.6        159.0    107.608 1947   60.323
## 1948         88.5 259.426      232.5        145.6    108.632 1948   61.122
## 1949         88.2 258.054      368.2        161.6    109.773 1949   60.171
## 1950         89.5 284.599      335.1        165.0    110.929 1950   61.187
## 1951         96.2 328.975      209.9        309.9    112.075 1951   63.221
## 1952         98.1 346.999      193.2        359.4    113.270 1952   63.639
```

```
str(longley)
```

```
## 'data.frame':    16 obs. of  7 variables:
##  $ GNP.deflator: num  83 88.5 88.2 89.5 96.2 ...
##  $ GNP         : num  234 259 258 285 329 ...
##  $ Unemployed  : num  236 232 368 335 210 ...
```

```
##  $ Armed.Forces: num  159 146 162 165 310 ...
##  $ Population  : num  108 109 110 111 112 ...
##  $ Year        : int  1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 ...
##  $ Employed    : num  60.3 61.1 60.2 61.2 63.2 ...
```

```
plot(longley)
```

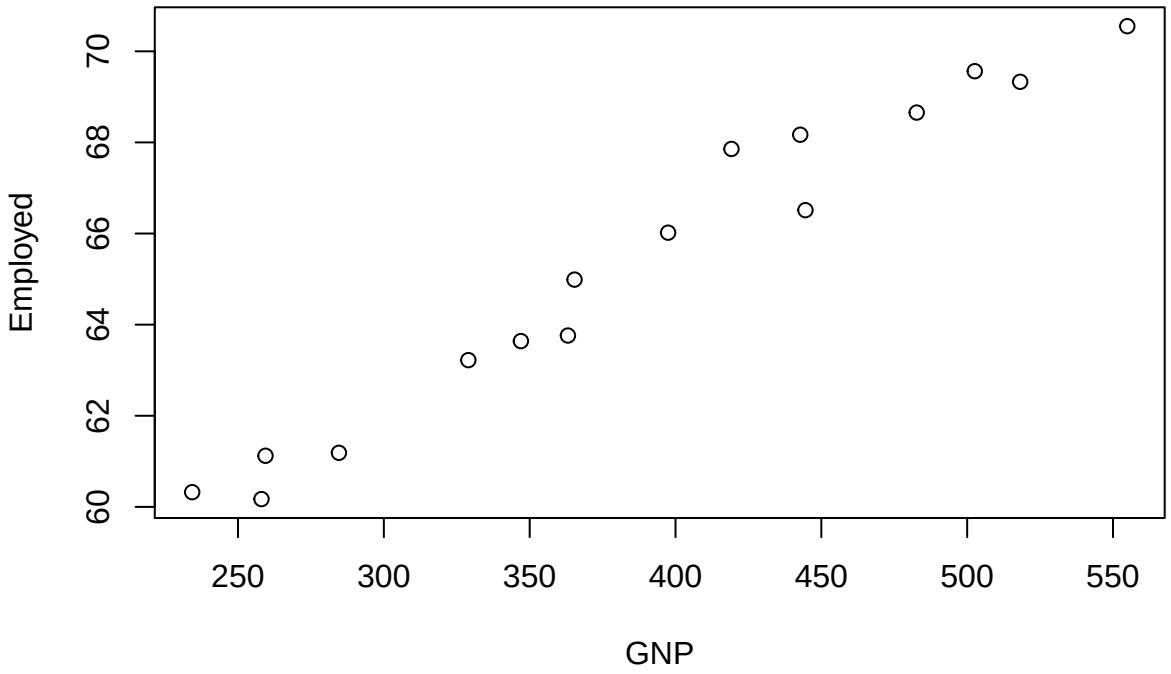


Now let's study `Employed` vs `GNP` and `Population`:

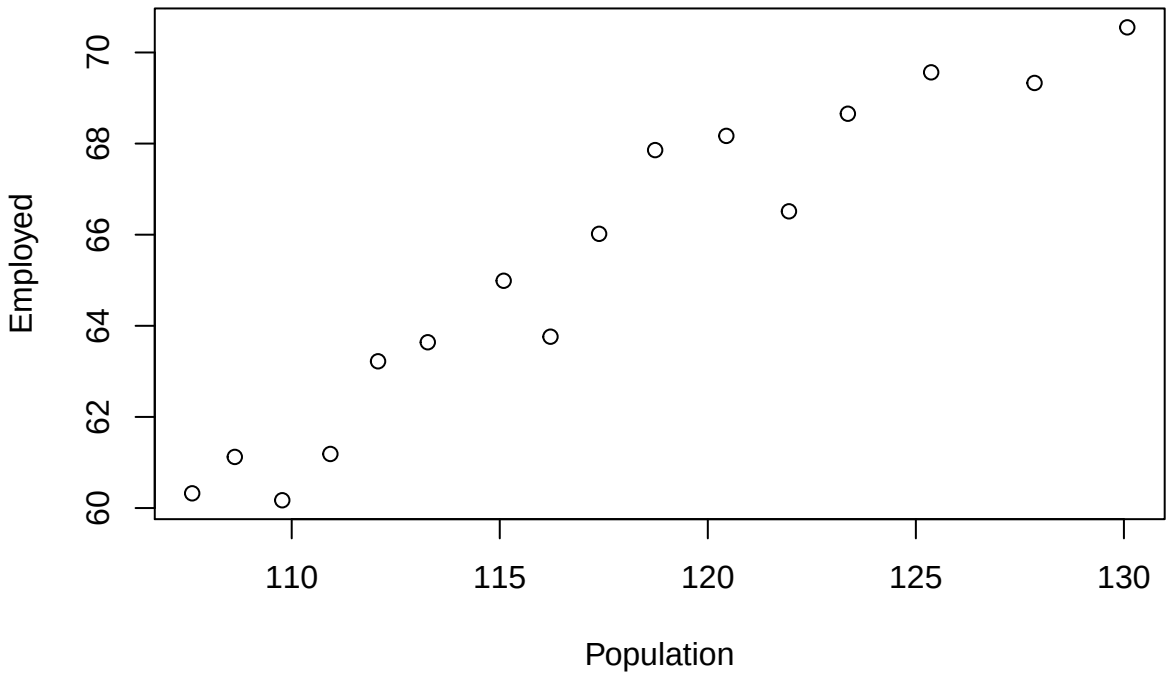
```
g = lm(Employed ~ GNP + Population, data=longley)
summary(g)
```

```
##
## Call:
## lm(formula = Employed ~ GNP + Population, data = longley)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.80899 -0.33282 -0.02329  0.25895  1.08800
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 88.93880   13.78503   6.452 2.16e-05 ***
## GNP          0.06317    0.01065   5.933 4.96e-05 ***
## Population  -0.40974    0.15214  -2.693   0.0184 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5459 on 13 degrees of freedom
## Multiple R-squared:  0.9791, Adjusted R-squared:  0.9758
## F-statistic: 303.9 on 2 and 13 DF,  p-value: 1.221e-11
```

```
plot( Employed ~ GNP, data=longley)
```




```
plot( Employed ~ Population, data=longley)
```

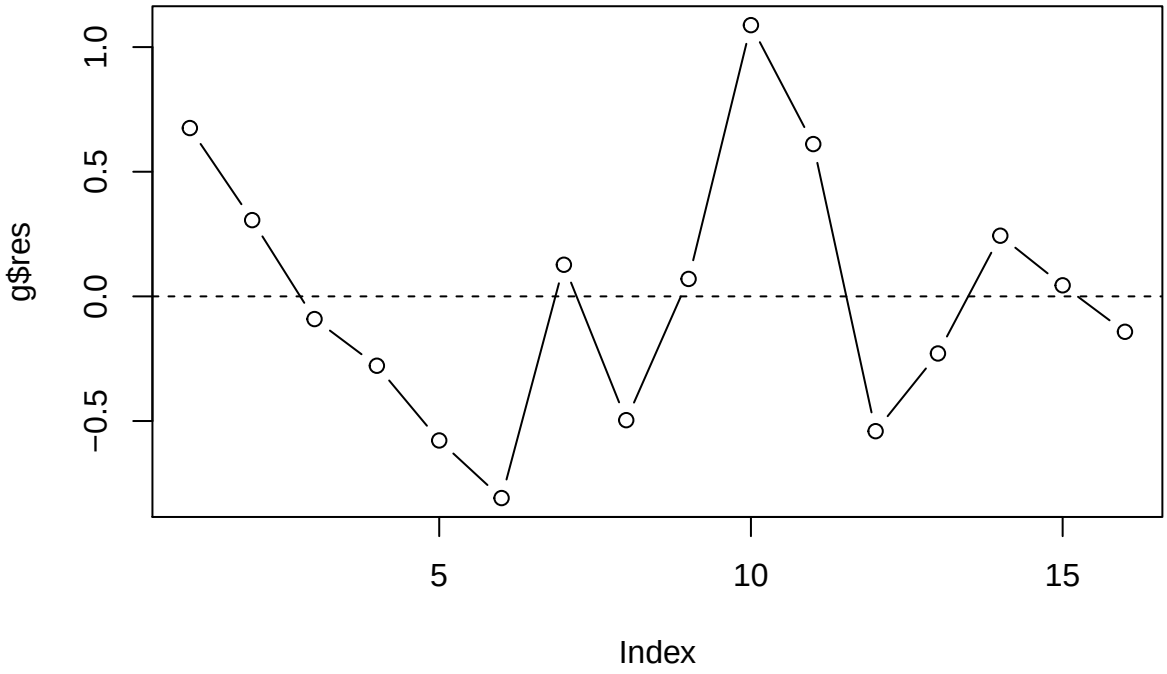



```
summary( g, correlation = T)
```

```
##
## Call:
## lm(formula = Employed ~ GNP + Population, data = longley)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.80899 -0.33282 -0.02329  0.25895  1.08800
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 88.93880   13.78503   6.452 2.16e-05 ***
## GNP          0.06317    0.01065   5.933 4.96e-05 ***
## Population  -0.40974    0.15214  -2.693   0.0184 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5459 on 13 degrees of freedom
## Multiple R-squared:  0.9791, Adjusted R-squared:  0.9758
## F-statistic: 303.9 on 2 and 13 DF,  p-value: 1.221e-11
##
## Correlation of Coefficients:
##            (Intercept) GNP
## GNP         0.98
## Population -1.00       -0.99
```

```
# residual plots
plot(g$res, type="b")
abline(h=0, lty=2)
```



Residuals appear autocorrelated (correlation of a signal with a delayed copy of itself as a function of delay):

```
cor(g$res[-1], g$res[-length(g$res)])
```

```
## [1] 0.3104092
```

We can then use generalised least squares. For this we estimate the correlation among errors first. And use these estimates in the model (there is also iteration involved):

```
library(nlme)
```

```
## Warning: package 'nlme' was built under R version 3.3.2
```

```
g = gls(Employed ~ GNP + Population,
        correlation=corAR1(form= ~ Year), # the correlation structure, predicted by Year
        data=longley)
summary(g)
```

```
## Generalized least squares fit by REML
##   Model: Employed ~ GNP + Population
##   Data: longley
##        AIC      BIC    logLik
##   44.66377 47.48852 -17.33188
##
## Correlation Structure: AR(1)
##  Formula: ~Year
##  Parameter estimate(s):
##       Phi
## 0.6441692
##
## Coefficients:
##                 Value Std.Error   t-value p-value
## (Intercept) 101.85813 14.198932  7.173647  0.0000
## GNP           0.07207  0.010606  6.795485  0.0000
## Population   -0.54851  0.154130 -3.558778  0.0035
##
##  Correlation:
##            (Intr) GNP
## GNP         0.943
## Population -0.997 -0.966
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -1.5924564 -0.5447822 -0.1055401  0.3639202  1.3281898
##
## Residual standard error: 0.689207
## Degrees of freedom: 16 total; 13 residual
```

Analysis of covariance

```
library(faraway)
head(twins)
```

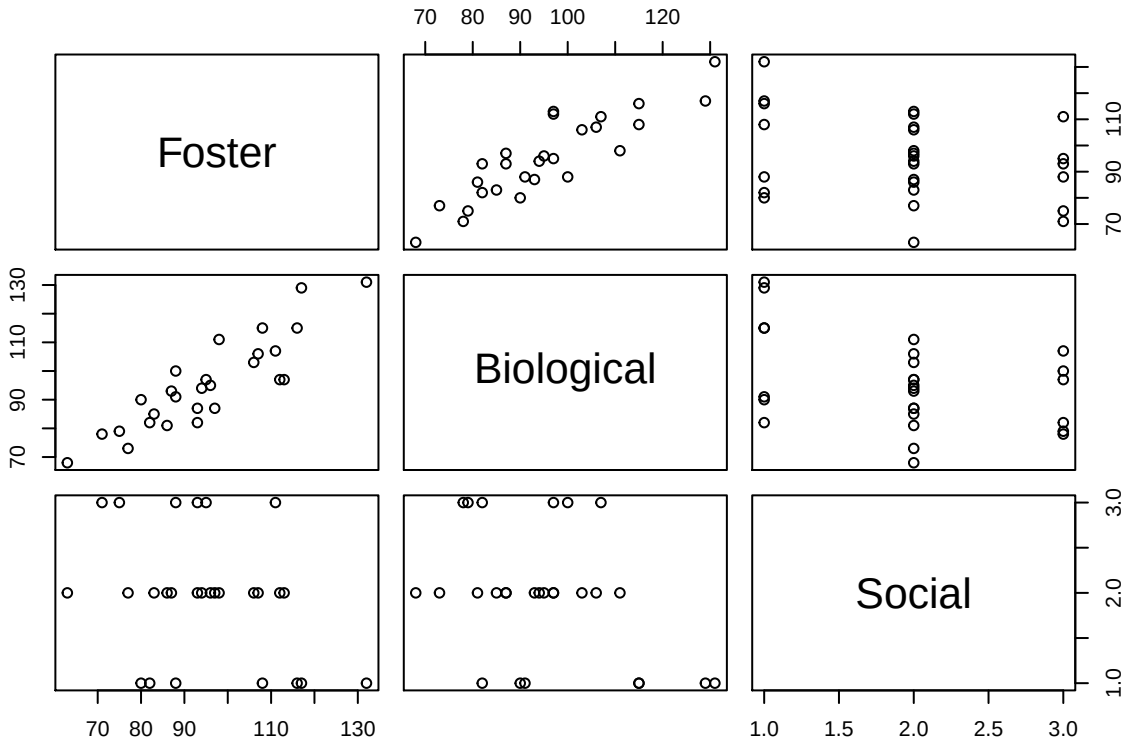
```
##   Foster Biological Social
## 1     82         82   high
## 2     80         90   high
## 3     88         91   high
## 4    108        115   high
## 5    116        115   high
## 6    117        129   high
```

```
str(twins)
```

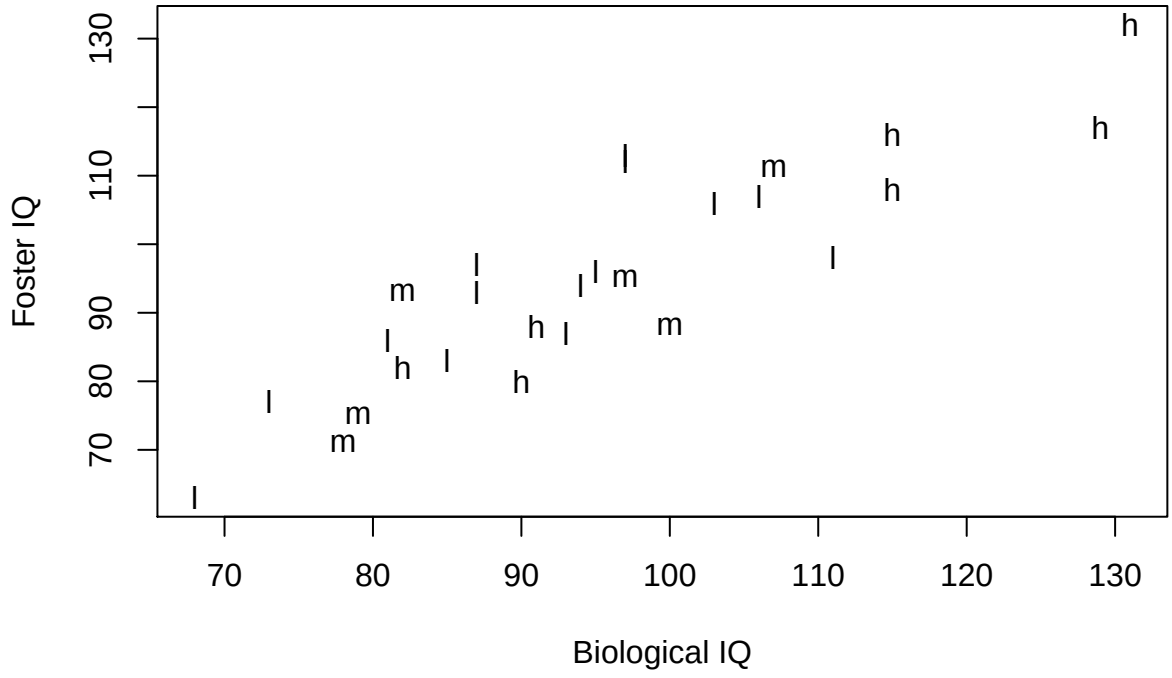
```
## 'data.frame':    27 obs. of  3 variables:
##  $ Foster    : num  82 80 88 108 116 117 132 71 75 93 ...
```

```
##  $ Biological: num  82 90 91 115 115 129 131 78 79 82 ...
##  $ Social    : Factor w/ 3 levels "high","low","middle": 1 1 1 1 1 1 1 3 3 3 ...
```

```
plot(twins)
```




```
plot(twins$B, twins$F, type="n",
     xlab="Biological IQ", ylab="Foster IQ")
text(twins$B, twins$F, substring(twins$S,1,1))
```



```r
g = lm(Biological ~ Foster*Social, twins)
summary(g)
```

```
##
## Call:
## lm(formula = Biological ~ Foster * Social, data = twins)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.4353  -4.6746   0.0629   2.9829  16.7255
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)           9.5054    14.8550   0.640    0.529
## Foster                0.9495     0.1415   6.708 1.23e-06 ***
## Sociallow            14.7930    19.8521   0.745    0.464
## Socialmiddle         18.4148    24.3230   0.757    0.457
## Foster:Sociallow     -0.2354     0.1985  -1.186    0.249
## Foster:Socialmiddle  -0.2450     0.2569  -0.954    0.351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.971 on 21 degrees of freedom
## Multiple R-squared:  0.8415, Adjusted R-squared:  0.8037
## F-statistic:  22.3 on 5 and 21 DF,  p-value: 9.551e-08
```

```r
anova(g)
```

```
## Analysis of Variance Table
##
## Response: Biological
##               Df Sum Sq Mean Sq  F value    Pr(>F)
## Foster         1 5007.8  5007.8 103.0579 1.487e-09 ***
## Social         2  327.3   163.7   3.3679   0.05387 .
## Foster:Social  2   82.0    41.0   0.8441   0.44403
## Residuals     21 1020.4    48.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# so no evidence for an interaction

g2 = lm(Biological ~ Foster + Social, twins)
summary(g2)
```

```
##
## Call:
## lm(formula = Biological ~ Foster + Social, data = twins)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.865  -4.105   1.235   2.497  16.324
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  24.15493    9.60058   2.516   0.0193 *
```

```
## Foster        0.80763    0.08943   9.031 5.05e-09 ***
## Sociallow    -8.62698    3.31726  -2.601   0.0160 *
## Socialmiddle -5.39927    4.06291  -1.329   0.1969
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.923 on 23 degrees of freedom
## Multiple R-squared:  0.8287, Adjusted R-squared:  0.8064
## F-statistic:  37.1 on 3 and 23 DF,  p-value: 5.575e-09
```

```
anova(g2)
```

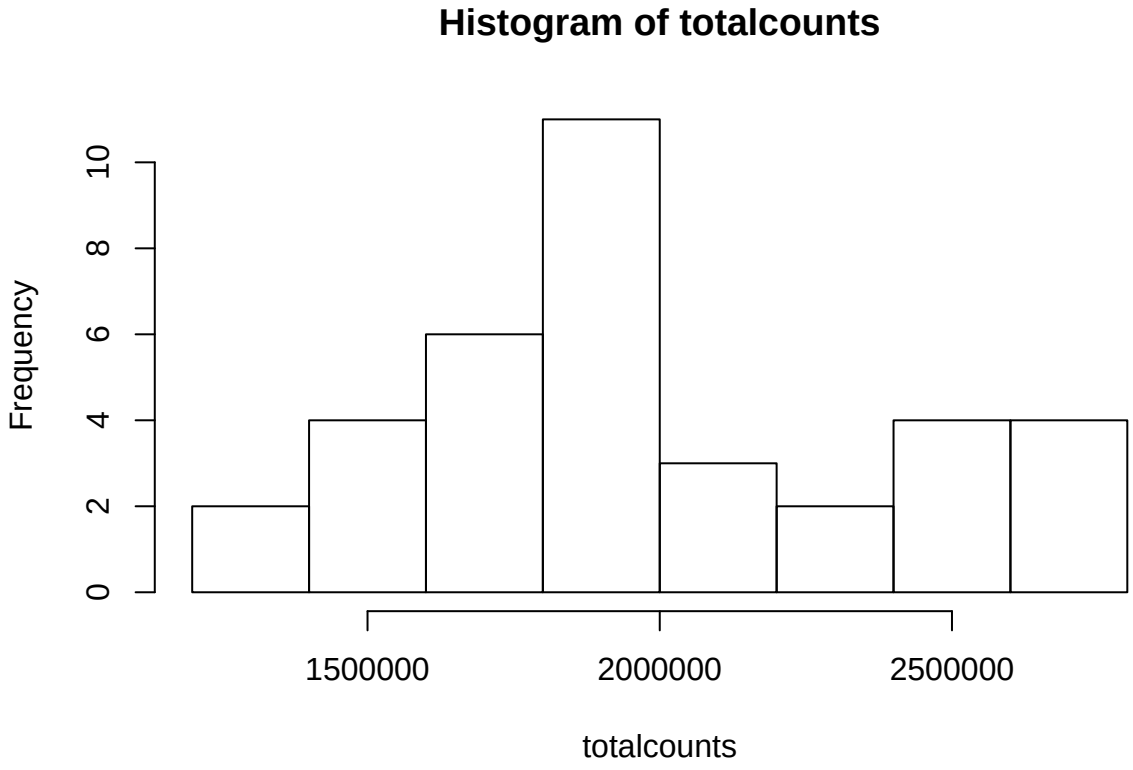
```
## Analysis of Variance Table
##
## Response: Biological
##           Df Sum Sq Mean Sq  F value   Pr(>F)
## Foster     1 5007.8  5007.8 104.4744 5.05e-10 ***
## Social     2  327.3   163.7   3.4142  0.05031 .
## Residuals 23 1102.5    47.9
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# so we might keep social
```

Mixed and nested models

Mixed models include both **fixed factors** and **random factors**, such as individual, family, region: where the grouping is of interest, but not the exact sampled elements. This can be tested when multiple measurements are made from the same subject.

```
load("liver_transcriptome_v1.Rdata")
hist(totalcounts)
```




```
# the linear model
anova( lm( totalcounts ~ indv + species + sex  ) )
```

```
## Analysis of Variance Table
##
## Response: totalcounts
##           Df     Sum Sq    Mean Sq F value    Pr(>F)
## indv      17 4.9880e+12 2.9341e+11  9.4778 8.652e-06 ***
## Residuals 18 5.5725e+11 3.0958e+10
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# note the difference
anova( lm( totalcounts ~ sex + species + indv ) )
```

```
## Analysis of Variance Table
##
## Response: totalcounts
##           Df     Sum Sq    Mean Sq F value    Pr(>F)
## sex        1 3.5093e+10 3.5093e+10  1.1335    0.3011
## species    2 1.8226e+12 9.1128e+11 29.4359 2.116e-06 ***
## indv      14 3.1304e+12 2.2360e+11  7.2226 8.587e-05 ***
## Residuals 18 5.5725e+11 3.0958e+10
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Because `indv` is **nested** inside `species` and `sex`, adding it first in the formula does not leave any variance to be explained by either.

Random effects using `lme`

But actually we should treat individual as a random effect:

```
null = lme( totalcounts ~ 1, random = ~ 1 | indv )
anova(null)
```

```
##             numDF denDF  F-value p-value
## (Intercept)     1    18 476.7498  <.0001
```

Now we can use the individual effect to estimate significance of the fixed effects:

```
full = lme( totalcounts ~ species + sex, random = ~ 1 | indv )
anova(full)
```

```
##             numDF denDF  F-value p-value
## (Intercept)     1    18 625.6047  <.0001
## species         2    14   4.0755  0.0403
## sex             1    14   0.1569  0.6980
```

As you notice, this appears much less significant.