Statistical modeling

Descriptive analysis and basic statistics in biomedical studies using R and Markdown

Juan R Gonzalez
juanr.gonzalez@isglobal.org

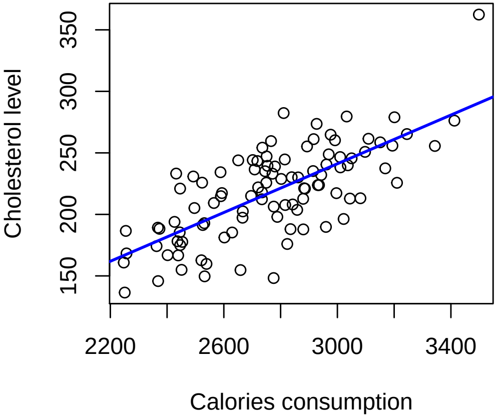
BRGE - Bioinformatics Research Group in Epidemiology
ISGlobal - Barcelona Institute for Global Health
http://brge.isglobal.org

IACS - Instituto Aragonés de Ciencias de la Salud
Zaragoza, February 26th

Statistical modelling

Outcome	Method	Example
Continuous	Linear regression	Factors that affects cholesterol levels
Binary	Logistic regression	Factors that affects developing cancer
Time to event	Survival	Factor that affect time until developing cancer

Linear regression



Cholesterol level
150
200
250
300
350
2200
2600
3000
3400
Calories consumption

Linear model

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n + \epsilon$$

α *correspond to the mean level of* Y *in the population* β_j *indicates the change in* Y *when* X_j *changes in 1 unit (after keeping the rest of* X_k *fixed)*

Example: Researchers are interested in knowing the factors that better explain air Ozone levels (variable `Ozone` in data frame `airquality`). They measure solar radiation (`Solar.R`), average wind (`Wind`) and temperature (`Temp`) in different months (`Months`) on 154 observations.

```r
data(airquality)
head(airquality)
```

```
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

Simple linear regression

```r
mod <- lm(Ozone ~ Temp, data=airquality)
mod
```

```
Call:
lm(formula = Ozone ~ Temp, data = airquality)

Coefficients:
(Intercept)         Temp
   -146.995        2.429
```

```r
summary(mod)
```

```
Call:
lm(formula = Ozone ~ Temp, data = airquality)

Residuals:
    Min      1Q  Median      3Q     Max 
-40.729 -17.409  -0.587  11.306 118.271 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -146.9955    18.2872  -8.038 9.37e-13 ***
Temp           2.4287     0.2331  10.418  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 23.71 on 114 degrees of freedom
  (37 observations deleted due to missingness)
Multiple R-squared:  0.4877,	Adjusted R-squared:  0.4832 
F-statistic: 108.5 on 1 and 114 DF,  p-value: < 2.2e-16
```

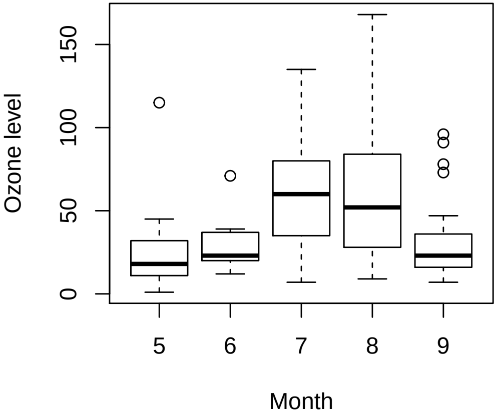
Interpretation of categorical factors

```r
mod.lin <- lm(Ozone ~ Month, data=airquality)
mod.lin
```

```
Call:
lm(formula = Ozone ~ Month, data = airquality)

Coefficients:
(Intercept)        Month
     15.657        3.678
```

```r
mod.fac <- lm(Ozone ~ as.factor(Month), data=airquality)
mod.fac
```

```
Call:
lm(formula = Ozone ~ as.factor(Month), data = airquality)

Coefficients:
      (Intercept)  as.factor(Month)6  as.factor(Month)7  as.factor(Month)8
           23.615              5.829             35.500             36.346
as.factor(Month)9
            7.833
```



Ozone level
0
50
100
150
5
6
7
8
9
Month

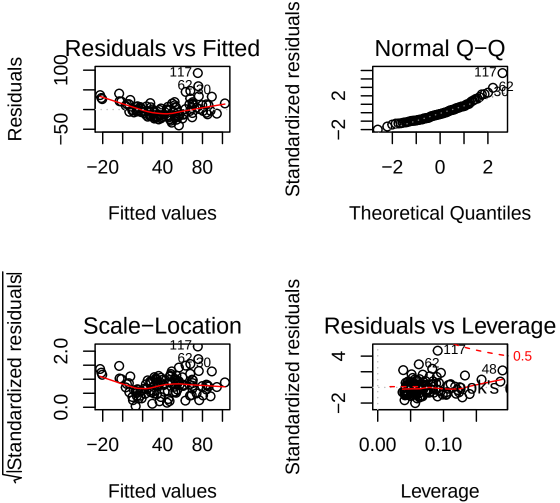
Multiple linear regression

```r
mod <- lm(Ozone ~ Solar.R + Wind +
            Temp + as.factor(Month), data=airquality)
mod
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp + as.factor(Month),
    data = airquality)

Coefficients:
      (Intercept)             Solar.R                Wind                Temp
        -74.23481             0.05222            -3.10872             1.87511
as.factor(Month)6  as.factor(Month)7  as.factor(Month)8  as.factor(Month)9
        -14.75895            -8.74861            -4.19654           -15.96728
```

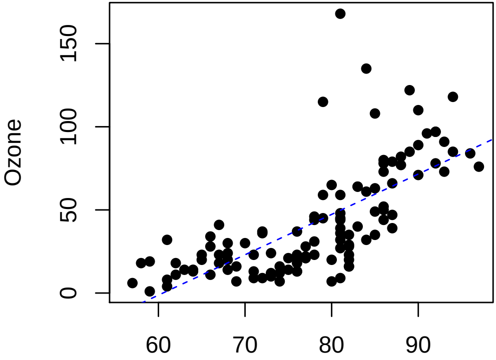
Model validation

```
par(mfrow=c(2,2))
plot(mod)
```



Visualization

```
mod <- lm(Ozone ~ Temp, data=airquality)
plot(airquality$Temp, airquality$Ozone,
     xlab="Temperature", ylab="Ozone", pch=16)
abline(mod, lty=2, col="blue")
```



Linear transformation

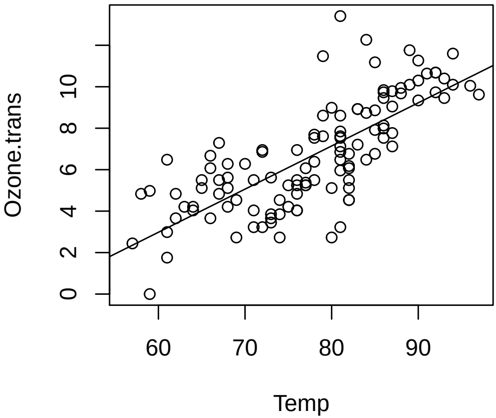
```r
library(car)
trans <- powerTransform(mod)
trans
```

```
Estimated transformation parameters
       Y1
0.2206725
```

```r
Ozone.trans <- bcPower(airquality$Ozone,
                       coef(trans, round=TRUE))

mod.trans <- lm(Ozone.trans ~ Temp, data=airquality)
mod.trans
```

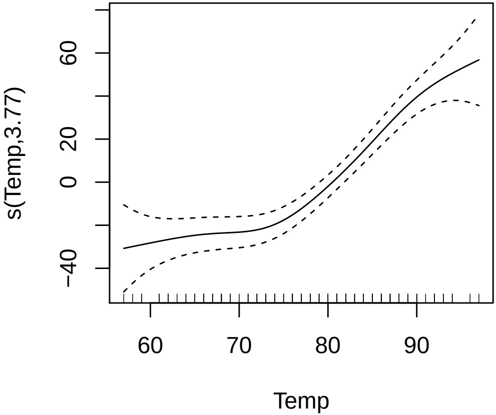
```
Call:
lm(formula = Ozone.trans ~ Temp, data = airquality)

Coefficients:
(Intercept)         Temp
    -9.5085       0.2081
```

```r
plot(Ozone.trans ~ Temp, data=airquality)
abline(mod.trans)
```



Splines

```
library(mgcv)
mod.gam <- gam(Ozone ~ s(Temp), data=airquality)
plot(mod.gam, se=TRUE)
```



Non-parametric test of linearity

```r
mod.gam <- gam(Ozone ~ s(Temp), data=airquality)
summary(mod.gam)
```

```
Family: gaussian
Link function: identity

Formula:
Ozone ~ s(Temp)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   42.129      2.044   20.61   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
          edf Ref.df     F p-value
s(Temp) 3.771  4.689 30.75  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.554   Deviance explained = 56.9%
GCV = 505.64  Scale est. = 484.84    n = 116
```

Logistic regression

Logistic Model

Y variable is binary (case/control, relapse/non-relapse, mortality, ...). In that case, the logit transformation guarantees linearity.

$$\log(p(Y = 1)/(1 - p(Y = 1))) = \alpha + \beta_1 X_1 + \ldots + \beta_k X_k$$

$\exp(\beta_k)$ can be interpreted as the odds ratio (OR) of having/developing/being $Y = 1$

Example: Reserchers are interested in determining whether a new treatment (varible `rx`) reduces mortality (variable `fustat`) in patients diagnosed with ovarian cancer. Data are available by typing:

```r
data(ovarian, package="survival")
head(ovarian)
```

```
  futime fustat     age resid.ds rx ecog.ps
1     59      1 72.3315        2  1       1
2    115      1 74.4932        2  1       1
3    156      1 66.4658        2  1       2
4    421      0 53.3644        2  2       1
5    431      1 50.3397        2  1       1
6    448      0 56.4301        1  1       2
```

```r
mod2 <- glm(fustat ~ rx, data=ovarian, family="binomial")
mod2
```

```
Call:  glm(formula = fustat ~ rx, family = "binomial", data = ovarian)

Coefficients:
(Intercept)           rx
     0.7783      -0.6242

Degrees of Freedom: 25 Total (i.e. Null);  24 Residual
Null Deviance:      35.89
Residual Deviance: 35.27    AIC: 39.27
```

```r
mod2 <- glm(fustat ~ rx, data=ovarian, family="binomial")
summary(mod2)
```

```
Call:
glm(formula = fustat ~ rx, family = "binomial", data = ovarian)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.2435  -0.9854  -0.9854   1.1127   1.3824

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.7783     1.2502   0.623    0.534
rx           -0.6242     0.7966  -0.784    0.433

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.890  on 25  degrees of freedom
Residual deviance: 35.268  on 24  degrees of freedom
AIC: 39.268

Number of Fisher Scoring iterations: 4
```

Survival analysis

However, in this study the probability of observing the outcome of interest depends on the time of follow-up. Therefore, survival analysis should be used instead. The most commong model is Cox proportional hazard risks model.

$$\lambda(t|X) = \lambda_0(t)exp(\beta X)$$

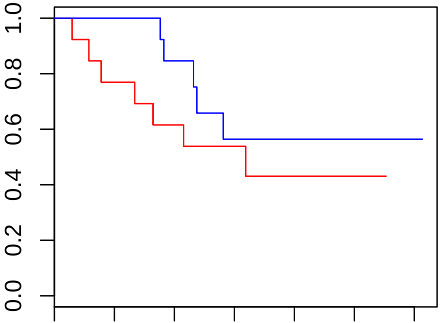
```r
library(survival)
mod3 <- coxph(Surv(futime, fustat) ~ rx, data=ovarian)
summary(mod3)
```

```
Call:
coxph(formula = Surv(futime, fustat) ~ rx, data = ovarian)

  n= 26, number of events= 12

      coef exp(coef) se(coef)      z Pr(>|z|)
rx -0.5964    0.5508   0.5870 -1.016     0.31

   exp(coef) exp(-coef) lower .95 upper .95
rx    0.5508      1.816    0.1743      1.74

Concordance= 0.608  (se = 0.078 )
Rsquare= 0.04   (max possible= 0.932 )
Likelihood ratio test= 1.05  on 1 df,   p=0.3052
Wald test            = 1.03  on 1 df,   p=0.3096
Score (logrank) test = 1.06  on 1 df,   p=0.3026
```

Kaplan-Meier

Cox regression is a semi-parametric model. A non-parametric estimation of survival curve can also be computed using Kaplan-Meier estimator:

```
mod4 <- survfit(Surv(futime, fustat) ~ rx, data=ovarian)
plot(mod4, col=c("red", "blue"))
```



Curves can be compared using log-rank rest

```r
mod5 <- survdiff(Surv(futime, fustat) ~ rx, data=ovarian)
mod5
```

```
Call:
survdiff(formula = Surv(futime, fustat) ~ rx, data = ovarian)

      N Observed Expected (O-E)^2/E (O-E)^2/V
rx=1 13        7     5.23     0.596      1.06
rx=2 13        5     6.77     0.461      1.06

 Chisq= 1.1  on 1 degrees of freedom, p= 0.303
```

Or Gehan-Wilcoxon test that is designed to detect differences at the begining of the study follow-up.

```
mod6 <- survdiff(Surv(futime, fustat) ~ rx, data=ovarian,
                 rho=1)
mod6
```

```
Call:
survdiff(formula = Surv(futime, fustat) ~ rx, data = ovarian,
    rho = 1)

      N Observed Expected (O-E)^2/E (O-E)^2/V
rx=1 13     5.89     4.12     0.761      1.68
rx=2 13     3.50     5.27     0.595      1.68

 Chisq= 1.7  on 1 degrees of freedom, p= 0.194
```

Model selection (general setting)

Models can be compared using Likelihoo Ratio Test (LRT)

```
air.ok <- airquality[complete.cases(airquality),]
mod0 <- lm(Ozone ~ Wind, data=air.ok)
mod1 <- lm(Ozone ~ Wind + Solar.R, data=air.ok)
anova(mod0, mod1, test="F")
```

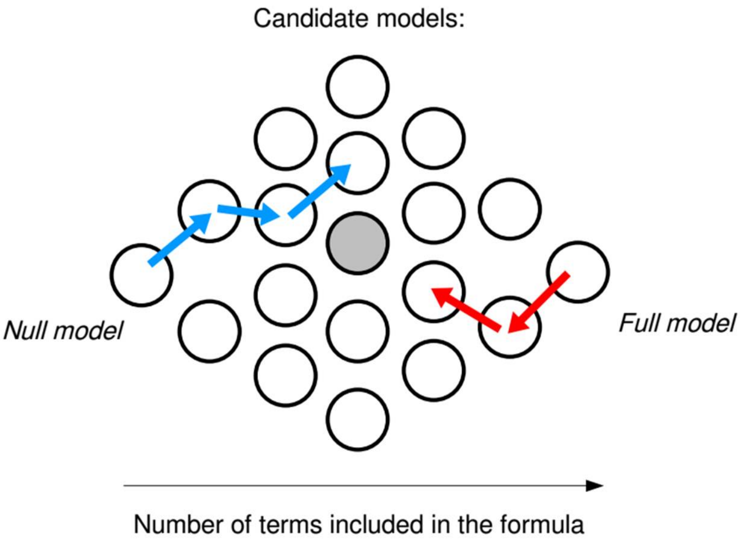
```
Analysis of Variance Table

Model 1: Ozone ~ Wind
Model 2: Ozone ~ Wind + Solar.R
  Res.Df   RSS Df Sum of Sq      F   Pr(>F)
1    109 76108
2    108 67053  1    9054.9 14.585 0.000224 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod0 <- glm(fustat ~ ecog.ps, data=ovarian, family="binomial")
mod1 <- glm(fustat ~ ecog.ps + rx, data=ovarian, family="binomia
anova(mod0, mod1, test="Chi")
```

```
Analysis of Deviance Table

Model 1: fustat ~ ecog.ps
Model 2: fustat ~ ecog.ps + rx
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1        24     34.550
2        23     33.894  1  0.65586    0.418
```

Stepwise selection

Real data problems normally consider several variables. Automatic LRTs should be used to select the best model:



```r
library(MASS)
modAll <- lm(Ozone ~ ., data=airquality)
modForw <- stepAIC(modAll, direction="forw", trace=0)
modForw
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp + Month + Day, data = airquality)

Coefficients:
(Intercept)      Solar.R         Wind         Temp        Month          Day
  -64.11632      0.05027     -3.31844      1.89579     -3.03996      0.27388
```

```r
modAll <- lm(Ozone ~ ., data=airquality)
modBack <- stepAIC(modAll, direction="back", trace=0)
modBack
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp + Month, data = airquality)

Coefficients:
(Intercept)      Solar.R         Wind         Temp        Month
   -58.0538       0.0496      -3.3165       1.8709      -2.9916
```

```r
modAll <- lm(Ozone ~ ., data=airquality)
modBoth <- stepAIC(modAll, direction="both", trace=0)
modBoth
```

```

Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp + Month, data = airquality)

Coefficients:
(Intercept)      Solar.R         Wind         Temp        Month
   -58.0538       0.0496      -3.3165       1.8709      -2.9916
```

```r
summary(modBoth)
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp + Month, data = airquality)

Residuals:
    Min      1Q  Median      3Q     Max 
-35.870 -13.968  -2.671   9.553  97.918 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -58.05384   22.97114  -2.527   0.0130 *  
Solar.R       0.04960    0.02346   2.114   0.0368 *  
Wind         -3.31651    0.64579  -5.136 1.29e-06 ***
Temp          1.87087    0.27363   6.837 5.34e-10 ***
Month        -2.99163    1.51592  -1.973   0.0510 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 20.9 on 106 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared:  0.6199,	Adjusted R-squared:  0.6055 
F-statistic: 43.21 on 4 and 106 DF,  p-value: < 2.2e-16
```

```r
modAll <- lm(Ozone ~ . - Month, data=airquality)
modBoth2 <- stepAIC(modAll, direction="both", trace=0)
modBoth2
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp, data = airquality)

Coefficients:
(Intercept)      Solar.R         Wind         Temp  
  -64.34208      0.05982     -3.33359      1.65209  
```

```r
summary(modBoth2)
```

```
Call:
lm(formula = Ozone ~ Solar.R + Wind + Temp, data = airquality)

Residuals:
    Min      1Q  Median      3Q     Max 
-40.485 -14.219  -3.551  10.097  95.619 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -64.34208   23.05472  -2.791  0.00623 ** 
Solar.R       0.05982    0.02319   2.580  0.01124 *  
Wind         -3.33359    0.65441  -5.094 1.52e-06 ***
Temp          1.65209    0.25353   6.516 2.42e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 21.18 on 107 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared:  0.6059,	Adjusted R-squared:  0.5948 
F-statistic: 54.83 on 3 and 107 DF,  p-value: < 2.2e-16
```

Session info

```r
sessionInfo()
```

```
R version 3.4.1 (2017-06-30)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 16299)

Matrix products: default

locale:
[1] LC_COLLATE=Spanish_Spain.1252  LC_CTYPE=Spanish_Spain.1252
[3] LC_MONETARY=Spanish_Spain.1252 LC_NUMERIC=C
[5] LC_TIME=Spanish_Spain.1252

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] MASS_7.3-47     survival_2.41-3 mgcv_1.8-17     nlme_3.1-131
[5] car_2.1-6       knitr_1.20

loaded via a namespace (and not attached):
 [1] Rcpp_0.12.12       magrittr_1.5       splines_3.4.1      lattice_0.20-35
 [5] minqa_1.2.4        stringr_1.3.0      tools_3.4.1        nnet_7.3-12
 [9] pbkrtest_0.4-7     parallel_3.4.1     grid_3.4.1         quantreg_5.33
[13] MatrixModels_0.4-1 htmltools_0.3.6    yaml_2.1.16        lme4_1.1-13
[17] rprojroot_1.3-2    digest_0.6.12      Matrix_1.2-10      nloptr_1.0.4
[21] codetools_0.2-15   evaluate_0.10.1    rmarkdown_1.8      stringi_1.1.6
[25] compiler_3.4.1     backports_1.1.0    SparseM_1.77
```