

# Logistic Regression

Andrea Huerfano

September 4, 2019

## Dataset

For this statistical method, we are going to use the data set crabs available in R through the library MASS, its variables are:

- *sp*: species - “B” or “O” for blue or orange.
- *sex*: M for male and F for female
- *index*: 1:50 within each of the four groups.
- *FL*: frontal lobe size (mm).
- *RW*: rear width (mm).
- *CL*: carapace length (mm).
- *CW*: carapace width (mm).
- *BD*: body depth (mm).

The function summary will be used to have a first idea of the data structure. Notice that the variable indice is deleted because we will suppose that all of the crabs belongs to the same group.

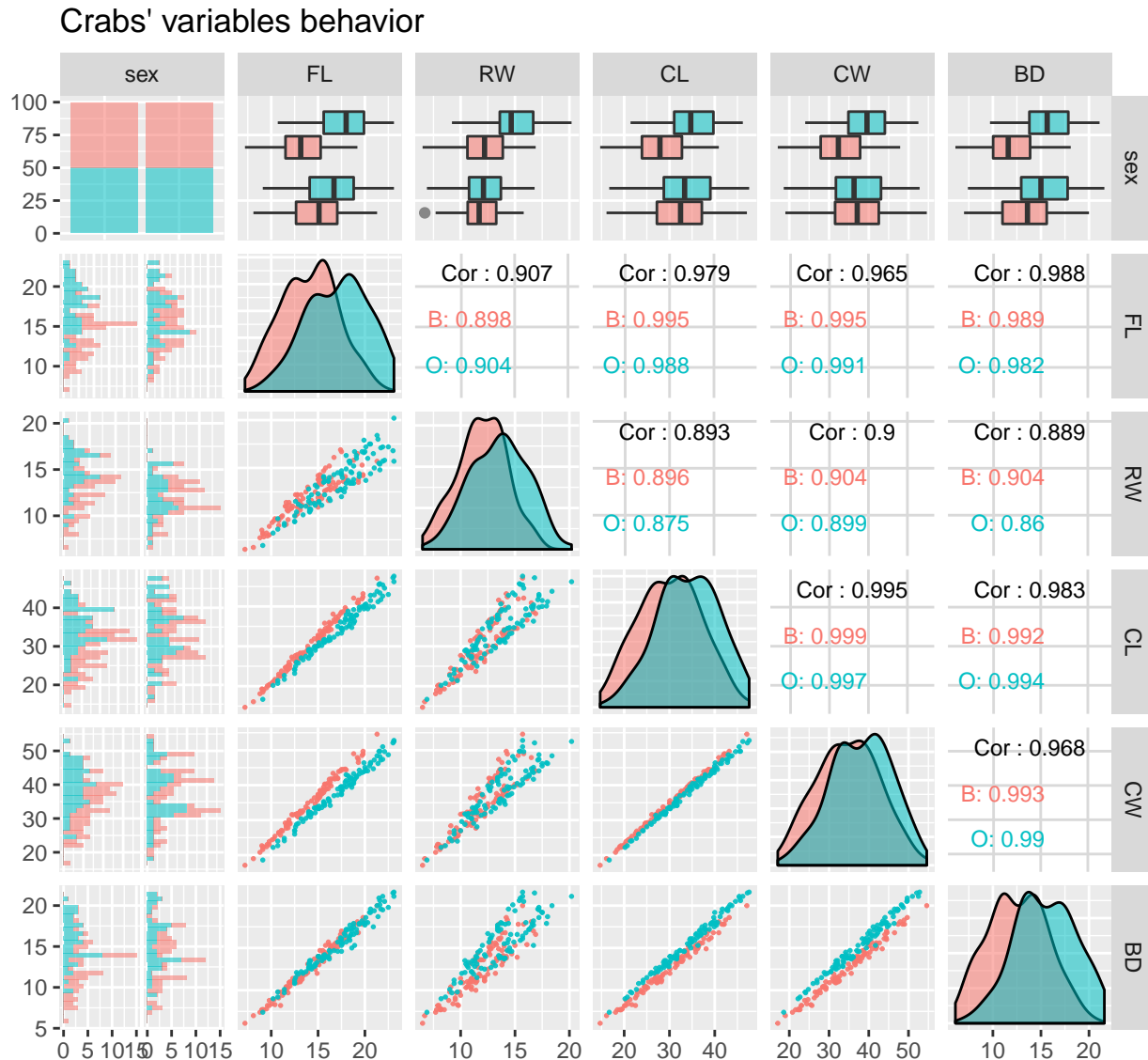
```
Packages <- c("MASS", "dplyr", "ggplot2", "readr", "pscl", "ROCR", "GGally")
lapply(Packages, library, character.only = TRUE)
set.seed(3)
```

```
data(crabs)
crabs<-crabs[,-3]
summary(crabs)
```

```
##  sp      sex      FL      RW      CL
##  B:100   F:100   Min.    : 7.20   Min.    : 6.50   Min.    :14.70
##  O:100   M:100   1st Qu.:12.90   1st Qu.:11.00   1st Qu.:27.27
##                      Median :15.55   Median :12.80   Median :32.10
##                      Mean    :15.58   Mean    :12.74   Mean    :32.11
##                      3rd Qu.:18.05   3rd Qu.:14.30   3rd Qu.:37.23
##                      Max.    :23.10   Max.    :20.20   Max.    :47.60
##      CW      BD
##  Min.    :17.10   Min.    : 6.10
##  1st Qu.:31.50   1st Qu.:11.40
##  Median :36.80   Median :13.90
##  Mean    :36.41   Mean    :14.03
##  3rd Qu.:42.00   3rd Qu.:16.60
##  Max.    :54.60   Max.    :21.60
```

In the data set we have two category variables: *sp* that is our target and *sex*, the others are quantitatives, that is the reason why in *sex* appears barplots and the other variables have density distribution plots and scatter plots. The two colors are associated with the break up of *sp* which has two levels, pink is associated with level B and blue with the second level (O). Notice that the *sex* distribution is the same for both of the levels because the two barplots have the same structure, with the box plots we can see that the behaviour of the other variables is different in the levels for *sex* and *sp*. Even when we haven't examine the correlation is easy to see that there is a strong linear relation between the variables and in some cases like *CL* vs. *RW* the dispersion increase when the variables take high values.

```
ggpairs(crabs, columns = 2:ncol(crabs), title = "Crabs' variables behavior",
  axisLabels = "show", mapping = aes(colour=sp,alpha = 5),
  upper = list(continuous = wrap("cor", size=3)),
  lower = list(continuous = wrap("points", alpha = 0.9, size=0.3)))
```



To create the model, the first step is preparing the split over the database to create two sets: training and testing datasets, in this case we are going to use 80 percent of the sample for training the model and the other 20 percent will be used to validate the model's quality. The distribution of the observation in the two set is made over a random simple sampling applied over the index.

```
train_index <- sample(1:nrow(crabs), 0.8 * nrow(crabs))
test_index <- setdiff(1:nrow(crabs), train_index)
# Build X_train and X_test
X_train <- crabs[train_index,]
X_test <- crabs[test_index,]
```

## Correlation

We can see that the variables have a strong correlation almost all of them, that is a problem if we would try to put in the model all of these variables. The just the variables sex,FL,RW and CL will keep to start the model.

```
cor(X_train[,3:ncol(crabs)])
```

```
##           FL           RW           CL           CW           BD
## FL 1.0000000 0.9027495 0.9800157 0.9661302 0.9897178
## RW 0.9027495 1.0000000 0.8918536 0.9007618 0.8879675
## CL 0.9800157 0.8918536 1.0000000 0.9951218 0.9830334
## CW 0.9661302 0.9007618 0.9951218 1.0000000 0.9681736
## BD 0.9897178 0.8879675 0.9830334 0.9681736 1.0000000
```

## Model identification

For the model selection we are going to use the Step aic and the R function glm will be used to compute de logistic regression, specifying the option family = binomial, that means the response variable is binary. AIC penalizes increasing the number of parameters into de model, and the best option will be the model with the smallest. Not all the variables are in the initial model because of the high correlation between them as I said before.

```
fit <- glm(sp ~ 1 + sex+FL+RW+CL+sex*FL+sex*RW, family=binomial, data=X_train)
stepAIC(fit)
```

```
## Start:  AIC=27.11
## sp ~ 1 + sex + FL + RW + CL + sex * FL + sex * RW
##
##           Df Deviance      AIC
## - sex:RW   1   13.224   25.224
## - sex:FL   1   13.467   25.467
## <none>           13.106   27.106
## - CL       1  130.058 142.058
##
## Step:  AIC=25.22
## sp ~ sex + FL + RW + CL + sex:FL
##
##           Df Deviance      AIC
## - sex:FL   1   13.945   23.945
## <none>           13.224   25.224
## - RW       1   20.569   30.569
## - CL       1  130.706 140.706
##
## Step:  AIC=23.95
## sp ~ sex + FL + RW + CL
##
##           Df Deviance      AIC
## <none>           13.945   23.945
## - RW       1   21.049   29.049
## - sex       1   24.175   32.175
## - CL       1  166.631 174.631
## - FL       1  200.546 208.546
##
## Call:  glm(formula = sp ~ sex + FL + RW + CL, family = binomial, data = X_train)
```

```
##
## Coefficients:
## (Intercept)      sexM      FL      RW      CL
##      -58.985      11.901      29.294      4.451     -14.158
##
## Degrees of Freedom: 159 Total (i.e. Null); 155 Residual
## Null Deviance:      221.6
## Residual Deviance: 13.95      AIC: 23.95
```

Well, let see the best model in detail:

```
fit<-glm(formula = sp ~ sex + FL + CL, family = binomial, data = X_train)
summary(fit)
```

```
##
## Call:
## glm(formula = sp ~ sex + FL + CL, family = binomial, data = X_train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.66885  -0.02286   0.00000   0.00221   2.13687
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -23.412      7.579  -3.089  0.00201 **
## sexM          2.101      1.229   1.710  0.08732 .
## FL           19.546      6.150   3.178  0.00148 **
## CL           -8.675      2.742  -3.163  0.00156 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 221.582  on 159  degrees of freedom
## Residual deviance:  21.049  on 156  degrees of freedom
## AIC: 29.049
##
## Number of Fisher Scoring iterations: 10
```

We would like to know the probability that the event occurs when we have a male, with FL=11.1 and CL=23.8.

```
x <- c(1,1,11.1,23.8)
eta <- sum(x*coef(fit))
prob <- exp(eta)/(1+exp(eta)) ;prob
```

```
## [1] 1.993698e-05
```

R has a function predict which enables us to calculate quickly probabilities just entry the values of each variable.

```
newdata <- data.frame( sex='M',FL=11.1,RW=9.9,CL=23.8,CW=27.1,BD=9.8)
probabilities <- fit %>% predict(newdata, type = "response")
probabilities
```

```
##              1
## 1.993698e-05
```

## Testing the model

It is using the probability that we are going to validate our model, predicting the value in the testing set for this is necessary define a threshold, in this case will be 0.4, that means that if the probability is greater than 0.4 the observation will be associated with the level *O* and in another case will be mark with *B*.

```
rev<-predict(fit,X_test,type = "response")
X_test$predicted.classes<- ifelse(rev > 0.4, "O", "B")
head(X_test,3)
```

```
##      sp sex  FL  RW  CL  CW  BD predicted.classes
## 1    B  M  8.1  6.7 16.1 19.0  7.0                B
## 6    B  M 10.8  9.0 23.0 26.5  9.8                B
## 10   B  M 11.8 10.5 25.2 29.3 10.3                B
```

```
tail(X_test,3)
```

```
##      sp sex  FL  RW  CL  CW  BD predicted.classes
## 186  0   F 19.7 16.7 39.9 43.6 18.2                0
## 190  0   F 20.1 17.2 39.8 44.1 18.6                0
## 193  0   F 20.6 17.5 41.5 46.2 19.2                0
```

To check the model accuracy we are going to see the percent associated with values that were classified right. The 92.5% of the observations were right classified.

```
mean(X_test$predicted.classes == X_test$sp)
```

```
## [1] 0.925
```

## Pseudo R2

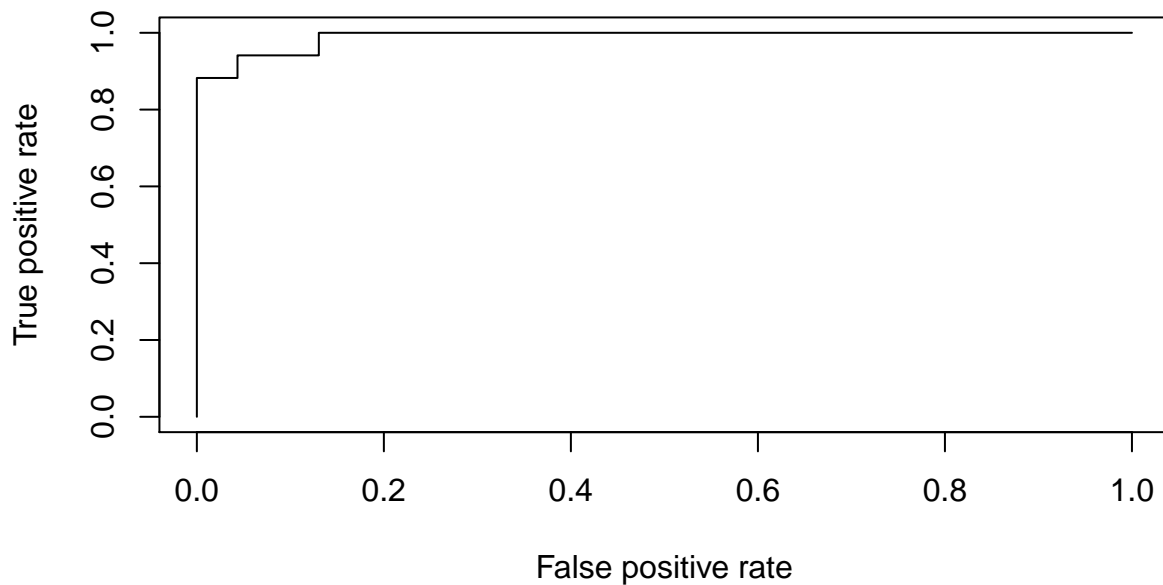
McFadden measure is 0.88 that is a really good value because this metric ranges from 0 to just under 1, with values closer to zero indicating that the model has no predictive power.

```
pR2(fit)
```

```
##          llh          llhNull          G2      McFadden          r2ML
## -10.5242979 -110.7910225    200.5334492    0.9050077    0.7144488
##          r2CU
##      0.9530456
```

## Roc curve.

```
prob <- predict(fit, newdata=X_test, type="response")
pred <- prediction(prob, X_test$sp)
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
plot(perf)
```



We examine the ROC curve which shows the trade off between the rate at which you can correctly predict something with the rate of incorrectly predicting something.

## Parameters interpretation

```
fit$coefficients
```

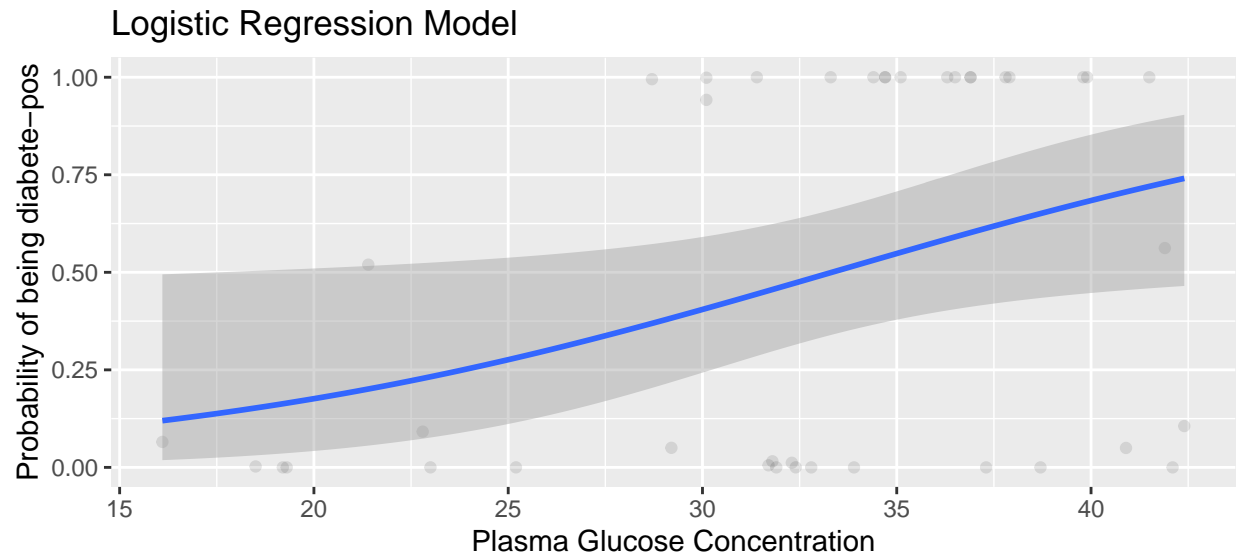
```
## (Intercept)      sexM          FL          CL
## -23.412346    2.100512   19.545675   -8.675129
```

Our final model is  $y = B_0 + B_1 \text{Sex}_{male} + B_2 \text{Fl} + B_3 \text{Cl}$

- $e^{B_1}$  when the crab is male the odds of success increase in  $e^{2.100512}$  8.166 times, that mean that males have 8.16 times of being  $O$  than females
- $e^{B_2}$  for each additional unit in FL the odds of success (being  $O$ ) increase in  $e^{19.55}$
- $e^{B_3}$  for each additional unit in CL the odds of being  $O$  decrease in  $e^{-8.87}$
- $e^{B_0}$  when the crab is female and have 0 value in FL and CL the odds of being  $O$  is  $e^{-23.41}$

Finally to check the well know s-shape for the logistic regression we are going to use the ggplot library

```
X_test$rev<-predict(fit,X_test,type = "response")
X_test %>%
  ggplot(aes(CL, rev)) +
  geom_point(alpha = 0.1) +
  geom_smooth(method = "glm", method.args = list(family = "binomial")) +
  labs(
    title = "Logistic Regression Model",
    x = "Plasma Glucose Concentration",
    y = "Probability of being diabete-pos"
  )
```



## References

Venables, W. N. & Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Hadley Wickham, Romain François, Lionel Henry and Kirill Müller (2019). *dplyr: A Grammar of Data Manipulation*. R package version 0.8.3. <https://CRAN.R-project.org/package=dplyr>

H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

Hadley Wickham, Jim Hester and Romain Francois (2018). *readr: Read Rectangular Text Data*. R package version 1.3.1. <https://CRAN.R-project.org/package=readr>

Simon Jackman (2017). *pscl: Classes and Methods for R*

Developed in the Political Science Computational Laboratory. United States Studies Centre, University of Sydney. Sydney, New South Wales, Australia. R package version 1.5.2. URL <https://github.com/atahk/pscl/>

Sing T, Sander O, Beerenwinkel N, Lengauer T (2005). “ROCR: visualizing classifier performance in R.” *Bioinformatics*, 21(20), 7881. <URL: <http://rocr.bioinf.mpi-sb.mpg.de>>.

Barret Schloerke, Jason Crowley, Di Cook, Francois Briatte, Moritz Marbach, Edwin Thoen, Amos Elberg and Joseph Larmarange (2018). *GGally: Extension to ‘ggplot2’*. R package version 1.4.0. <https://CRAN.R-project.org/package=GGally>