Analisi di regressione in R

Table of Contents

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(readxl)  
library(HH)

## Loading required package: lattice  
## Loading required package: grid  
## Loading required package: latticeExtra  
##   
## Attaching package: 'latticeExtra'  
##   
## The following object is masked from 'package:ggplot2':  
##   
## layer  
##   
## Loading required package: multcomp  
## Loading required package: mvtnorm  
## Loading required package: survival  
## Loading required package: TH.data  
## Loading required package: MASS  
##   
## Attaching package: 'MASS'  
##   
## The following object is masked from 'package:dplyr':  
##   
## select  
##   
##   
## Attaching package: 'TH.data'  
##   
## The following object is masked from 'package:MASS':  
##   
## geyser  
##   
## Loading required package: gridExtra  
##   
## Attaching package: 'gridExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine  
##   
##   
## Attaching package: 'HH'  
##   
## The following object is masked from 'package:lubridate':  
##   
## interval  
##   
## The following object is masked from 'package:purrr':  
##   
## transpose

Per questa analisi, usiamo un dataset relativo allo studio di Panagakis et al. (2017), nel quale gli autori analizzano l’effetto dell’età di prima riproduzione (AFR, *Age of First Reproduction*) su *early reproductive success*, longevità, *late reproductive success* e sui tassi di senescenza, tenendo contemporaneamente conto delle condizioni ambientali natali (*density at birth*) e della qualità individuale. Lo studio è condotto sulla capra delle nevi, in Canada (*Canadian Rocky Mountains, west-central Alberta*).

Come prima cosa, verifichiamo appunto la presenza di effetti della densità alla nascita sulla longevità degli animali. Ci interessa anche sapere se questo effetto è diverso per gli animali che si riproducono e gli animali che non si riproducono.

# Uno sguardo ai dati

Come prima cosa, usiamo la funzione read\_excel per caricare i dati:

mydata <- read\_excel("data/CawRidge\_LifetimeData.xlsx")  
head(mydata)

## # A tibble: 6 × 19  
## goat YOB agecaptured longevity AFR ALR ers34birth ers34wean  
## <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr> <chr>   
## 1 1 1983 4 13 NA 10 NA NA   
## 2 2 1984 4 15 NA 15 NA NA   
## 3 3 1983 5 11 NA 11 NA NA   
## 4 4 1984 4 11 NA 8 NA NA   
## 5 5 1988 1 10 4 10 1 1   
## 6 6 1989 1 10 4 10 1 1   
## # ℹ 11 more variables: ers3456birth <chr>, ers3456wean <chr>,  
## # ers123birth <chr>, ers123wean <chr>, LRSbirth <chr>, LRSwean <chr>,  
## # lateRSbirth <chr>, lateRSwean <chr>, meddensity <chr>, densityYOB <chr>,  
## # quality <chr>

# table(is.na(mydata$densityYOB))

mydata$densityYOB <- as.numeric(mydata$densityYOB)

## Warning: NAs introduced by coercion

# table(is.na(mydata$densityYOB))

Le variabili che ci interessano sono densityYOB (*density at the Year Of Birth*) e longevity. Usiamo inoltre la colonna ALR (*Age at last reproduction; never=known to never reproduce*) per discriminare tra le femmine che si sono riprodotte almeno una volta e quelle che non si sono mai riprodotte. Creo a questo scopo una variabile *ad hoc*, che divide gli animali in due gruppi:

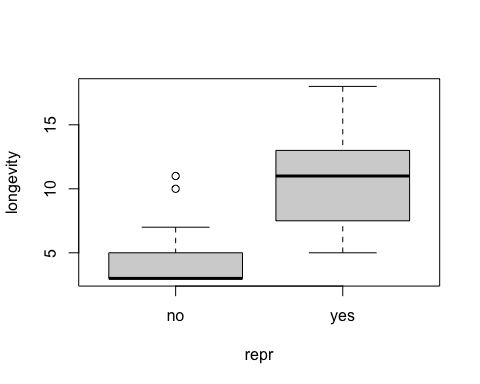
mydata %>%   
 dplyr::select(densityYOB, longevity, ALR) %>%   
 mutate(repr = "yes") %>%   
 na.omit() -> mydata  
mydata$repr[mydata$ALR == "never"] <- "no"  
mydata$repr <- as.factor(mydata$repr)  
  
mydata %>%   
 filter(repr == "yes") -> mydata\_repr  
mydata %>%   
 filter(repr == "no") -> mydata\_norepr

# Confronto tra le longevità medie dei due gruppi: t-test e intervalli di confidenza

mydata %>%   
 group\_by(repr) %>%   
 summarise(media = mean(longevity))

## # A tibble: 2 × 2  
## repr media  
## <fct> <dbl>  
## 1 no 4.33  
## 2 yes 10.3

boxplot(longevity ~ repr, data = mydata)



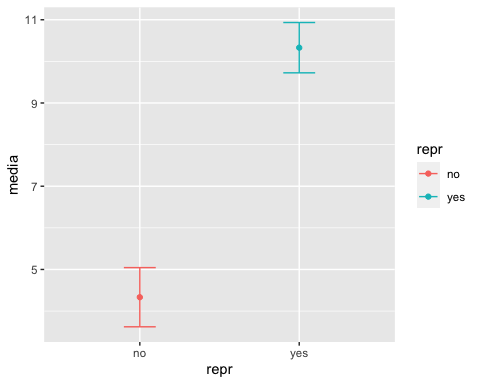
t.test(longevity ~ repr, data = mydata)

##   
## Welch Two Sample t-test  
##   
## data: longevity by repr  
## t = -10.847, df = 67.003, p-value < 2.2e-16  
## alternative hypothesis: true difference in means between group no and group yes is not equal to 0  
## 95 percent confidence interval:  
## -7.098856 -4.892372  
## sample estimates:  
## mean in group no mean in group yes   
## 4.333333 10.328947

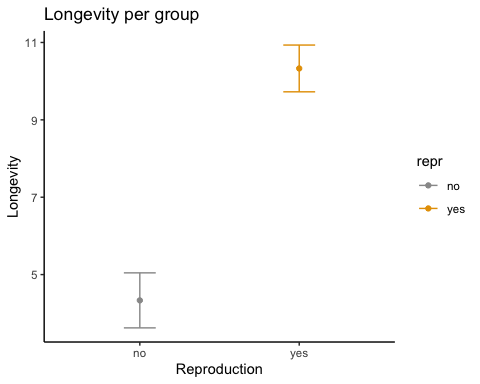
mydata %>%   
 group\_by(repr) %>%   
 summarise(media = mean(longevity),  
 SE = sd(longevity) / sqrt(length(longevity)),  
 n = n()) %>%   
 rowwise() %>%   
 mutate(lcl = (media + qt( c(0.05, 0.95), n - 1) \* SE)[1],  
 ucl = (media + qt( c(0.05, 0.95), n - 1) \* SE)[2]) -> intervalli  
intervalli

## # A tibble: 2 × 6  
## # Rowwise:   
## repr media SE n lcl ucl  
## <fct> <dbl> <dbl> <int> <dbl> <dbl>  
## 1 no 4.33 0.417 27 3.62 5.04  
## 2 yes 10.3 0.363 76 9.72 10.9

# Default line plot  
p <- ggplot(intervalli, aes(x=repr, y=media, color=repr)) +   
 # geom\_line() +  
 geom\_point()+  
 geom\_errorbar(aes(ymin=lcl, ymax=ucl), width=.2,  
 position=position\_dodge(0.05))  
print(p)



# Finished line plot  
p+labs(title="Longevity per group", x="Reproduction", y = "Longevity")+  
 theme\_classic() +  
 scale\_color\_manual(values=c('#999999','#E69F00'))



# Analisi di regressione

R stima la retta di regressione linear con la funzione *lm*.

mod <- lm(longevity ~ densityYOB, data = mydata\_repr)  
names(mod)

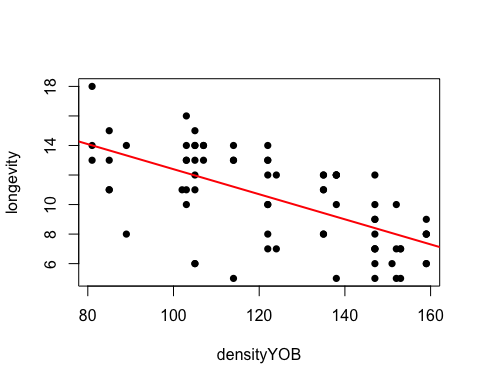
## [1] "coefficients" "residuals" "effects" "rank"   
## [5] "fitted.values" "assign" "qr" "df.residual"   
## [9] "xlevels" "call" "terms" "model"

coef(mod)

## (Intercept) densityYOB   
## 20.8802700 -0.0848572

Visualizziamo i dati:

plot(longevity ~ densityYOB, data = mydata\_repr, pch = 16)  
abline(coef(mod), col="red", lwd=2)



summary(mod)

##   
## Call:  
## lm(formula = longevity ~ densityYOB, data = mydata\_repr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6.2065 -1.4063 0.6029 1.8995 3.9932   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 20.88027 1.53377 13.61 < 2e-16 \*\*\*  
## densityYOB -0.08486 0.01212 -7.00 9.83e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.471 on 74 degrees of freedom  
## Multiple R-squared: 0.3984, Adjusted R-squared: 0.3902   
## F-statistic: 49 on 1 and 74 DF, p-value: 9.835e-10

## Esercizio

1. Interpretiamo la pendenza
2. Interpretiamo l’intercetta
3. Usiamo la retta di regressione per ottenere una stima puntuale della longevità media di un animale che è nato in un anno in cui la densità era pari a 130.

20.88 + (-0.085) \* (130) # point estimate

## [1] 9.83

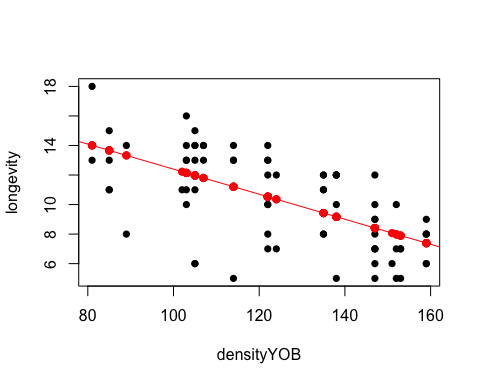
# Valori fittati e valori predetti

La stima di *Y* per un valore di *x* che corrisponde a un valore osservato è detta *fitted value*.

fitted(mod)

## 1 2 3 4 5 6 7 8   
## 14.006837 13.327979 13.667408 13.667408 13.327979 13.667408 13.667408 12.139979   
## 9 10 11 12 13 14 15 16   
## 12.139979 14.006837 12.139979 12.139979 11.970264 14.006837 11.970264 11.970264   
## 17 18 19 20 21 22 23 24   
## 12.139979 11.970264 11.970264 11.970264 11.206549 11.970264 11.206549 11.206549   
## 25 26 27 28 29 30 31 32   
## 11.206549 12.139979 11.800550 11.800550 11.970264 12.224836 11.800550 10.527692   
## 33 34 35 36 37 38 39 40   
## 10.527692 10.527692 10.527692 9.424548 9.424548 10.527692 9.424548 10.527692   
## 41 42 43 44 45 46 47 48   
## 9.424548 9.424548 9.169977 10.527692 10.527692 10.357977 9.169977 9.169977   
## 49 50 51 52 53 54 55 56   
## 9.169977 9.424548 9.169977 9.169977 8.406262 8.406262 8.406262 8.406262   
## 57 58 59 60 61 62 63 64   
## 10.357977 8.406262 8.406262 8.406262 8.406262 8.406262 8.406262 7.981976   
## 65 66 67 68 69 70 71 72   
## 7.981976 7.387976 7.981976 7.387976 7.897119 7.897119 7.897119 7.387976   
## 73 74 75 76   
## 7.387976 7.387976 7.387976 8.066833

plot(longevity ~ densityYOB, data = mydata\_repr, pch = 16)  
abline(coef(mod), col="red")  
points(mydata\_repr$densityYOB, fitted(mod), col="red", pch=19)

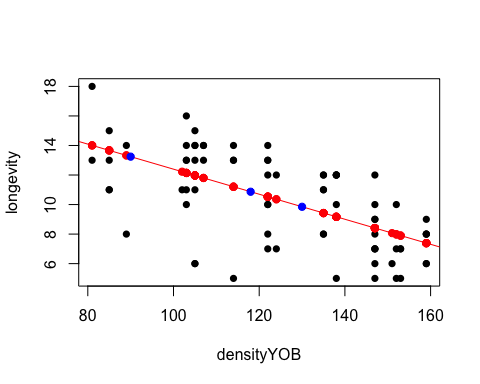


Le previsioni di *Y* in corrispondenza dei valori di *x* che non sono necessariamente parte dei dati osservati vengono fatte usando la funzione *predict*:

predict(mod, newdata = data.frame(densityYOB = c(90,118,130)))

## 1 2 3   
## 13.243122 10.867121 9.848834

plot(longevity ~ densityYOB, data = mydata\_repr, pch = 16)  
abline(coef(mod), col="red")  
points(mydata\_repr$densityYOB, fitted(mod), col="red", pch=19)  
points(c(90,118,130),  
 predict(mod, newdata = data.frame(densityYOB = c(90,118,130))),  
 col="blue", pch=19)



# Residui

I residui del modello possono essere visualizzati con la funzione *residuals*:

residuals(mod)

## 1 2 3 4 5 6   
## -0.006836956 0.672020624 -2.667408166 -0.667408166 -5.327979376 -2.667408166   
## 7 8 9 10 11 12   
## 1.332591834 1.860021390 0.860021390 -1.006836956 -1.139978610 -2.139978610   
## 13 14 15 16 17 18   
## 0.029735785 3.993163044 -0.970264215 -5.970264215 3.860021390 3.029735785   
## 19 20 21 22 23 24   
## 2.029735785 2.029735785 2.793450562 -5.970264215 -6.206549438 1.793450562   
## 25 26 27 28 29 30   
## 1.793450562 0.860021390 1.199450180 2.199450180 1.029735785 -1.224835808   
## 31 32 33 34 35 36   
## 2.199450180 3.472308143 -0.527691857 2.472308143 -2.527691857 2.575451710   
## 37 38 39 40 41 42   
## 1.575451710 -0.527691857 -1.424548290 -3.527691857 2.575451710 -1.424548290   
## 43 44 45 46 47 48   
## -4.169976697 2.472308143 1.472308143 -3.357977462 2.830023303 2.830023303   
## 49 50 51 52 53 54   
## 0.830023303 1.575451710 2.830023303 2.830023303 0.593738081 0.593738081   
## 55 56 57 58 59 60   
## -1.406261919 -0.406261919 1.642022538 -1.406261919 3.593738081 1.593738081   
## 61 62 63 64 65 66   
## -3.406261919 -2.406261919 -1.406261919 2.018024068 -2.981975932 1.612024451   
## 67 68 69 70 71 72   
## -0.981975932 0.612024451 -0.897118734 -2.897118734 -0.897118734 0.612024451   
## 73 74 75 76   
## -1.387975549 -1.387975549 0.612024451 -2.066833129

# Residual standard error  
modsumry <- summary(mod)  
modsumry$sigma

## [1] 2.470618

# Sintesi del modello

summary(mod)

##   
## Call:  
## lm(formula = longevity ~ densityYOB, data = mydata\_repr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -6.2065 -1.4063 0.6029 1.8995 3.9932   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 20.88027 1.53377 13.61 < 2e-16 \*\*\*  
## densityYOB -0.08486 0.01212 -7.00 9.83e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.471 on 74 degrees of freedom  
## Multiple R-squared: 0.3984, Adjusted R-squared: 0.3902   
## F-statistic: 49 on 1 and 74 DF, p-value: 9.835e-10

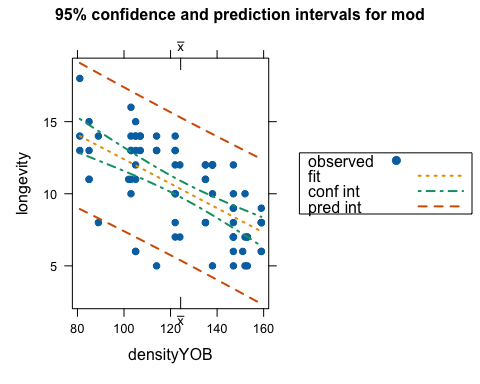
Nella sezione *Coefficients* troviamo le stime dei parametri e i loro errori standard, rispettivamente nella seconda e terza colonna. Possiamo anche ottenere gli intervalli di confidenza al 95%:

confint(mod)

## 2.5 % 97.5 %  
## (Intercept) 17.8241725 23.93636746  
## densityYOB -0.1090121 -0.06070227

Questi sono gli intervalli di confidenza per i parametri, ma, in generale, quanto sono buone le nostre sitme di *Y*? Quanto ‘crediamo’ al nostro modello?

ci.plot(mod)

 Notate che abbiamo due tipi di intervalli di confidenza. L’intervallo di confidenza è riferito alla media, mentre l’intervallo della *prediction* fa riferimento al singoo valore.

## Tipi di intervalli di confidenza

I seguenti sono i tipi di intervalli di confidenza utilizzati per le previsioni nella regressione e in altri modelli lineari:

* Intervallo di previsione (*prediction interval*) - Fornisce un intervallo di valori probabili per una singola risposta.
* Intervallo di confidenza della previsione - Fornisce un intervallo di valori probabili per la risposta media.

**Un intervallo di previsione è un intervallo che probabilmente conterrà il valore di risposta di una singola nuova osservazione, date le impostazioni specificate dei predittori nel modello** .

L’intervallo di confidenza del 95% della previsione è più ristretto: si può essere sicuri al 95% che questo intervallo includa la risposta media di tutte gli animali che nascono ad una data densità. **L’intervallo di confidenza della previsione è un intervallo che probabilmente contiene la risposta media, date le impostazioni specificate dei predittori nel modello. Proprio come gli intervalli di confidenza regolari, l’intervallo di confidenza della previsione presenta un intervallo per la media piuttosto che la distribuzione dei singoli punti di dati**.

L’intervallo di previsione è sempre più ampio dell’intervallo di confidenza a causa dell’incertezza aggiuntiva che comporta la previsione di una singola risposta rispetto alla risposta media.

# R2

Esistono diverse definizioni di R2 che solo a volte sono equivalenti. Quando è inclusa un’intercetta, è semplicemente il quadrato del coefficiente di correlazione campionaria (cioè r) tra gli esiti osservati e i valori osservati dei predittori. Se si includono ulteriori regressori, R2 è il quadrato del coefficiente di correlazione multipla. In entrambi i casi, il coefficiente di determinazione varia da 0 a 1.

L’uso di un R2 aggiustato è un tentativo di tenere conto del fenomeno dell’aumento automatico e spurio dell’R2 quando al modello vengono aggiunte ulteriori variabili esplicative. Si tratta di una modifica dell’R2 che aggiusta il numero di termini esplicativi in un modello rispetto al numero di punti dati. L’R2 corretto può essere negativo e il suo valore sarà sempre inferiore o uguale a quello dell’R2. A differenza dell’R2, l’R2 corretto aumenta solo quando l’aumento dell’R2 (dovuto all’inclusione di una nuova variabile esplicativa) è superiore a quello che ci si aspetterebbe di vedere per caso.

# I gradi di libertà

* Gradi di libertà corretti per il modello: parametri - 1
* Gradi di libertà per l’errore: numero di dati - parametri