Machine Learning

Matteo Gambera

1 aprile 2020

What is a prediction

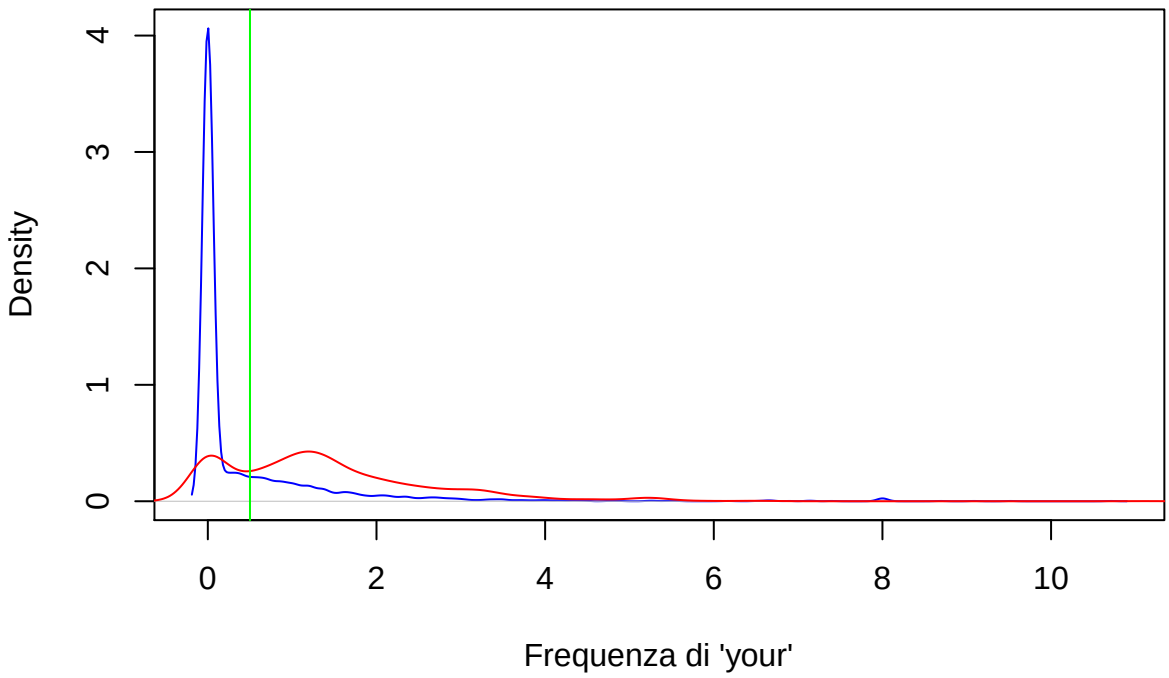
```
library(kernlab)
data(spam)
str(spam)
```

```
## 'data.frame':    4601 obs. of  58 variables:
##  $ make             : num  0 0.21 0.06 0 0 0 0 0 0.15 0.06 ...
##  $ address          : num  0.64 0.28 0 0 0 0 0 0 0 0.12 ...
##  $ all              : num  0.64 0.5 0.71 0 0 0 0 0 0.46 0.77 ...
##  $ num3d            : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ our              : num  0.32 0.14 1.23 0.63 0.63 1.85 1.92 1.88 0.61 0.19 ...
##  $ over             : num  0 0.28 0.19 0 0 0 0 0 0 0.32 ...
##  $ remove           : num  0 0.21 0.19 0.31 0.31 0 0 0 0.3 0.38 ...
##  $ internet         : num  0 0.07 0.12 0.63 0.63 1.85 0 1.88 0 0 ...
##  $ order            : num  0 0 0.64 0.31 0.31 0 0 0 0.92 0.06 ...
##  $ mail             : num  0 0.94 0.25 0.63 0.63 0 0.64 0 0.76 0 ...
##  $ receive          : num  0 0.21 0.38 0.31 0.31 0 0.96 0 0.76 0 ...
##  $ will             : num  0.64 0.79 0.45 0.31 0.31 0 1.28 0 0.92 0.64 ...
##  $ people           : num  0 0.65 0.12 0.31 0.31 0 0 0 0 0.25 ...
##  $ report           : num  0 0.21 0 0 0 0 0 0 0 0 ...
##  $ addresses        : num  0 0.14 1.75 0 0 0 0 0 0 0.12 ...
##  $ free             : num  0.32 0.14 0.06 0.31 0.31 0 0.96 0 0 0 ...
##  $ business         : num  0 0.07 0.06 0 0 0 0 0 0 0 ...
##  $ email            : num  1.29 0.28 1.03 0 0 0 0.32 0 0.15 0.12 ...
##  $ you              : num  1.93 3.47 1.36 3.18 3.18 0 3.85 0 1.23 1.67 ...
##  $ credit           : num  0 0 0.32 0 0 0 0 0 3.53 0.06 ...
##  $ your             : num  0.96 1.59 0.51 0.31 0.31 0 0.64 0 2 0.71 ...
##  $ font             : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ num000           : num  0 0.43 1.16 0 0 0 0 0 0 0.19 ...
##  $ money            : num  0 0.43 0.06 0 0 0 0 0 0.15 0 ...
##  $ hp               : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hpl              : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ george           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ num650           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ lab              : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ labs             : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ telnet           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ num857           : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ data             : num  0 0 0 0 0 0 0 0 0.15 0 ...
##  $ num415           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ num85            : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ technology       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ num1999          : num  0 0.07 0 0 0 0 0 0 0 0 ...
##  $ parts            : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ pm               : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ direct           : num  0 0 0.06 0 0 0 0 0 0 0 ...
##  $ cs               : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ meeting          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ original         : num  0 0 0.12 0 0 0 0 0 0.3 0 ...
##  $ project          : num  0 0 0 0 0 0 0 0 0 0.06 ...
##  $ re               : num  0 0 0.06 0 0 0 0 0 0 0 ...
##  $ edu              : num  0 0 0.06 0 0 0 0 0 0 0 ...
##  $ table            : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ conference       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ charSemicolon    : num  0 0 0.01 0 0 0 0 0 0 0.04 ...
##  $ charRoundbracket : num  0 0.132 0.143 0.137 0.135 0.223 0.054 0.206 0.271 0.03 ...
##  $ charSquarebracket: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ charExclamation  : num  0.778 0.372 0.276 0.137 0.135 0 0.164 0 0.181 0.244 ...
##  $ charDollar       : num  0 0.18 0.184 0 0 0 0.054 0 0.203 0.081 ...
##  $ charHash         : num  0 0.048 0.01 0 0 0 0 0 0.022 0 ...
##  $ capitalAve       : num  3.76 5.11 9.82 3.54 3.54 ...
##  $ capitalLong      : num  61 101 485 40 40 15 4 11 445 43 ...
##  $ capitalTotal     : num  278 1028 2259 191 191 ...
##  $ type             : Factor w/ 2 levels "nonspam","spam": 2 2 2 2 2 2 2 2 2 2 ...
```

Quante volte appare “your” in una email

```
## Quante volte appare in una email non spam
plot(density(spam$your[spam$type == "nonspam"]),
     col = "blue", main = "" , xlab = "Frequenza di 'your'")
## Quante volte appare in una email spam
lines(density(spam$your[spam$type == "spam"]),
     col = "red")
## Aggiungo cutoff guardando immagine
abline(v=0.5, col = "green")
```



Scrivo un algoritmo per tagliare, se il numero di volte che appare your è maggiore di C Scelgo C guardando immagine

```
cutoff <- 0.5
prediction <- ifelse(spam$your > cutoff, "spam", "nonspam")
## E' un arrey con spam o non spam
## Comparo i risultati dell'algoritmo con quelli veri
confronto <- table(prediction, spam$type)/length(spam$type)
confronto
```

```
##
## prediction   nonspam      spam
##    nonspam 0.4590306 0.1017170
##    spam    0.1469246 0.2923278
```

```
## L'accuratezza è data dalla somma dei numeri sulla diagonale
accuratezza <- (confronto[1] + confronto[4])*100
paste0(round(accuratezza,1),"%")
```

```
## [1] "75.1%"
```

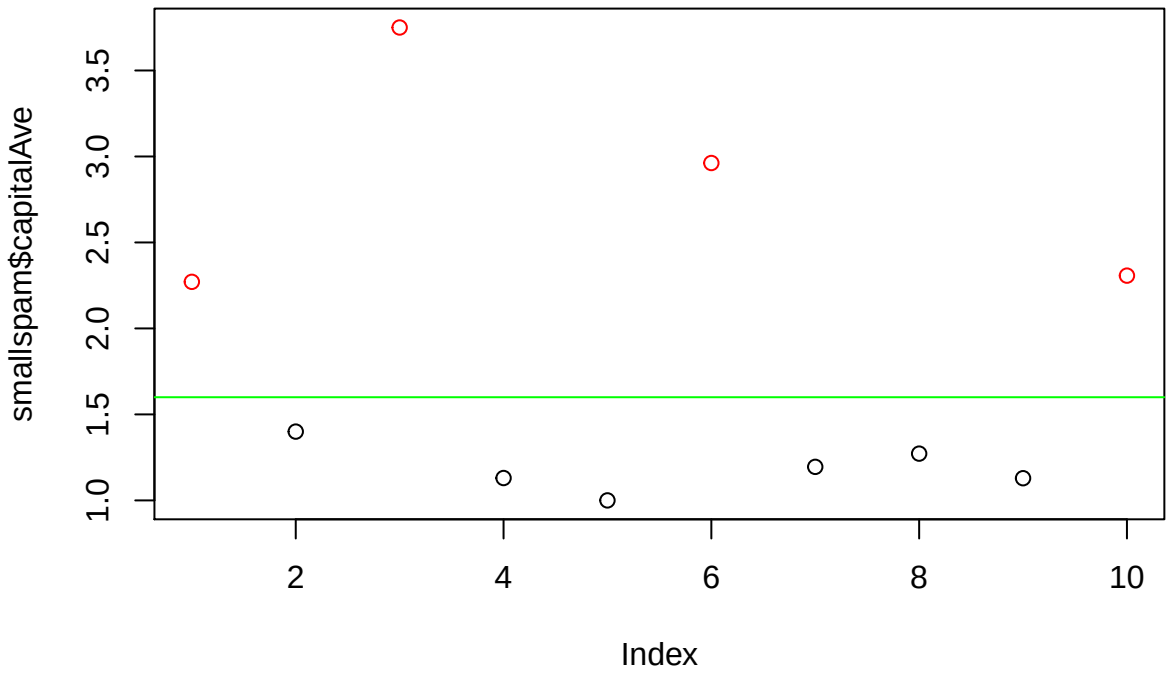
Relative importance of steps

question > input data > features > algorithm > parameters > evaluation

In sample and out of sample

in sample erro: errore che commetti su dati utilizzati anche per costruire il modello out of sample error: errore che commetti su nuovi dati

```
set.seed(1)
## Prendo 10 email
smallspam <- spam[sample(dim(spam)[1],size=10),]
## Li faccio diventare 1 e 2 in base alla tipologia
spamlabel <- (smallspam$type == "spam")*1 +1
plot(smallspam$capitalAve, col = spamlabel)
abline(h = 1.6 , col = "green")
```



Creiamo una funzione per categorizzare

```
regola <- function(x){
    prediction <- rep(NA, length(x))
    prediction[x > 1.6] <- "spam"
    prediction[x <= 1.6] <- "nonspam"
    return(prediction)
}
smalldf  <- table(regola(smallspam$capitalAve),smallspam$type)/length(smallspam$type)
smalldf
```

```
##
##           nonspam spam
```

```
##   nonspam     0.6  0.0
##   spam        0.0  0.4
```

```
paste0("accuratezza del ",round(smalldf[1] + smalldf[4],1)*100, "%")
```

```
## [1] "accuratezza del 100%"
```

Funziona in modo perfetto, ho lavorato su dati che già conoscevo Ora applichiamolo su tutte le email (che ancora non conosco)

```
bigdf <- table(regola(spam$capitalAve), spam$type)
bigdf
```

```
##
##           nonspam spam
##   nonspam    1002  179
##   spam       1786 1634
```

```
paste0("accuratezza del ",round((bigdf[1] + bigdf[4])/length(spam$type),1)*100, "%")
```

```
## [1] "accuratezza del 60%"
```

Non è più così accurato, infatti

Prediction study design

1. definire errore
2. split data (training, test) and more

data dimension:

- training 60%
- test 20%
- validation 20%

Types of error

- true positive = persona malata, la diagnostico malata
- falso positivo = persona sana, la diagnostico malata **type_1 error**
- True negative = persona sana, la diagnostico sana
- False negative = persona malata, la diagnostico sana **type_2 error**

positive = ok, negative = rejected ### key quantities * sensitivity positive test/disease tp/(tp + fn) * specificity negative test/not disease * positive predicted value disease/ positive test * Accuracy correct outcome

```
bigdf
```

```
##
##           nonspam spam
##   nonspam    1002  179
##   spam       1786 1634
```

```
paste0("sensitivity ", round((bigdf[1]/(bigdf[3]+bigdf[1]))*100,1), "%")
```

```
## [1] "sensitivity 84.8%"
```

```
paste0("specificity ", round((bigdf[4]/(bigdf[2]+bigdf[4]))*100,1), "%")
```

```
## [1] "specificity 47.8%"
```

```
paste0("accuracy ", round(((bigdf[4]+bigdf[1])/(bigdf[1]+bigdf[2]+bigdf[3]+bigdf[4]))*100,1), "%")
```

```
## [1] "accuracy 57.3%"
```

```
paste0("Positive predicted value ", round((bigdf[1]/(bigdf[1]+bigdf[2]))*100,1), "%")
```

```
## [1] "Positive predicted value 35.9%"
```

```
paste0("Negative predicted value ", round((bigdf[4]/(bigdf[4]+bigdf[3]))*100,1), "%")
```

```
## [1] "Negative predicted value 90.1%"
```

ROC curves

Misura la bontà di un algoritmo di predizione

- asse x ho 1- specificity (probabilità di essere un falso positivo)
- asse y ho sensitivity (probsbilità di essere un vero positivo)

L'area sotto la curva rappresenta quanto è buono:

- AUC = 0.5 linea a 45°(random)
- AUC = 1 perfetto
- AUC ~ 0,8 considerato buono

Cross validation

utilizzata per identificare features rilevanti e creare modelli e stimare parametri
si tratta di dividere i dati... tipi di divisione:

- K-fold; classico, k grande = - errori + varianza
- random subsampling; prendo dei pezzettini a caso;
- leave one out

What data should you use