A.A 2021/2022

Laboratorio R per la Biostatistica

Delmiglio Claudia - 830882

Gadda Alberto - 824029

A.A 2021/2022

Dataset

Il dataset «MIOPI LABORATORIO» raccoglie informazioni relative ad alcuni pazienti che si sono sottoposti al test SD-OCT, un esame radiografico non invasivo che punta ad effettuare una scansione della retina.

Per ogni paziente abbiamo a disposizione i risultati di alcuni aspetti che sono stati rilevati durante quest'ultimo test.

L'obiettivo che il presente lavoro si pone è quello di studiare quale è l'insieme di variabili, tra le molte a disposizione, che sono utili a discriminare un paziente affetto da CNV da uno affetto da un'altra patologia.

In questa presentazione vengono riportate parti del codice R utilizzato durante l'analisi, tuttavia per consultare l'intero codice è possibile visitare il seguente link:

https://github.com/albertogadda/Laboratorio_R_per_la_Biostatistica/blob/main/code.R

Dataset

Il dataset è composto da 122 righe e 62 colonne.

Alcune di queste colonne, essendo relative allo stesso tema, sono state opportunamente aggregate. Solo due colonne (VASCULAR NETWORK on OCTA 1 e VASCULAR NETWORK on OCTA 2) presentavano missing values, ed essendo molti (76%) si è deciso che la strategia migliore per gestirli fosse quella di eliminare tali colonne. La colonna EVAL, contenente delle date, ha dovuto subire una fase di post-processing al fine di rendere uniformi i dati in essa contenuti. Al fine della nostra analisi è stata però rimossa dal dataset poiché poco utile.

La colonna FINAL DIAGNOSIS, il nostro outcome, è stata dicotomizzata assegnando valore «CNV» o «Other» ad ogni istanza.

Queste operazioni hanno portato ad un dataset composto da 122 righe e 45 colonne.

La maggior parte delle variabili categoriali assume i seguenti possibili esiti:

- Y: YES
- N: NO
- D: Il test ha prodotto un esito dubbio
- NAP: Not Applicable, non è stato svolto il test

EDA

Tabella Final Diagnosis – Sex - Age

	CNV (N=83)	Other (N=39)	Total (N=122)
Sex			
F	56 (67.5%)	27 (69.2%)	83 (68.0%)
M	27 (32.5%)	12 (30.8%)	39 (32.0%)
Age (years)			
Mean (SD)	67.7 (13.0)	62.5 (16.8)	66.0 (14.5)
Median [Min, Max]	68.0 [39.0, 89.0]	65.0 [22.0, 89.0]	67.0 [22.0, 89.0]

Tabella 1

```
label(data$SEX) <- "Sex"
label(data$AGE) <- "Age"
units(data$AGE) <- "years"
tab1<- table1(~ SEX + AGE  | data$FINAL_DIAGNOSIS, data=data, topclass="Rtable1-zebra", overall = "Total" )
```

EDA

Box Plot

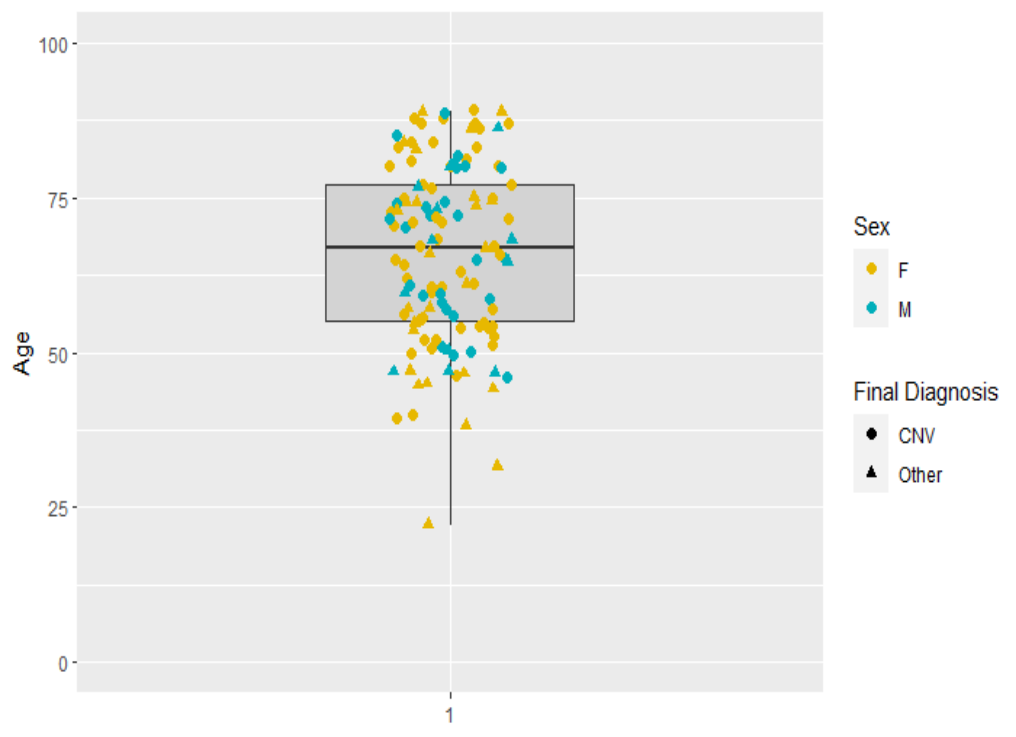


Grafico 1

```
#grafico1
grafico1 <- ggplot(data, aes(x = FINAL_DIAGNOSIS, y = AGE))
grafico1 + geom_boxplot()
grafico1 + geom_boxplot(notch = TRUE, fill = "lightgray")+
  stat_summary(fun.y = median, geom = "point",
               shape = 18, size = 2.5, color = "#FC4E07")

grafico1<- ggplot(data, aes(x = factor(1), y = AGE)) +
  geom_boxplot(width = 0.4, fill = "lightgray") +
  geom_jitter(aes(color = SEX, shape = FINAL_DIAGNOSIS),
              width = 0.1, size = 2) +
  scale_color_manual(values = c("#E7B800", "#00AFBB")) +
  labs(x = NULL)+
  labs(colour="Sex") +
  labs(shape="Final Diagnosis")+
  scale_y_continuous("Age", limits = c(0,100) )

grafico1
```

EDA

Recent Metamorphopsias

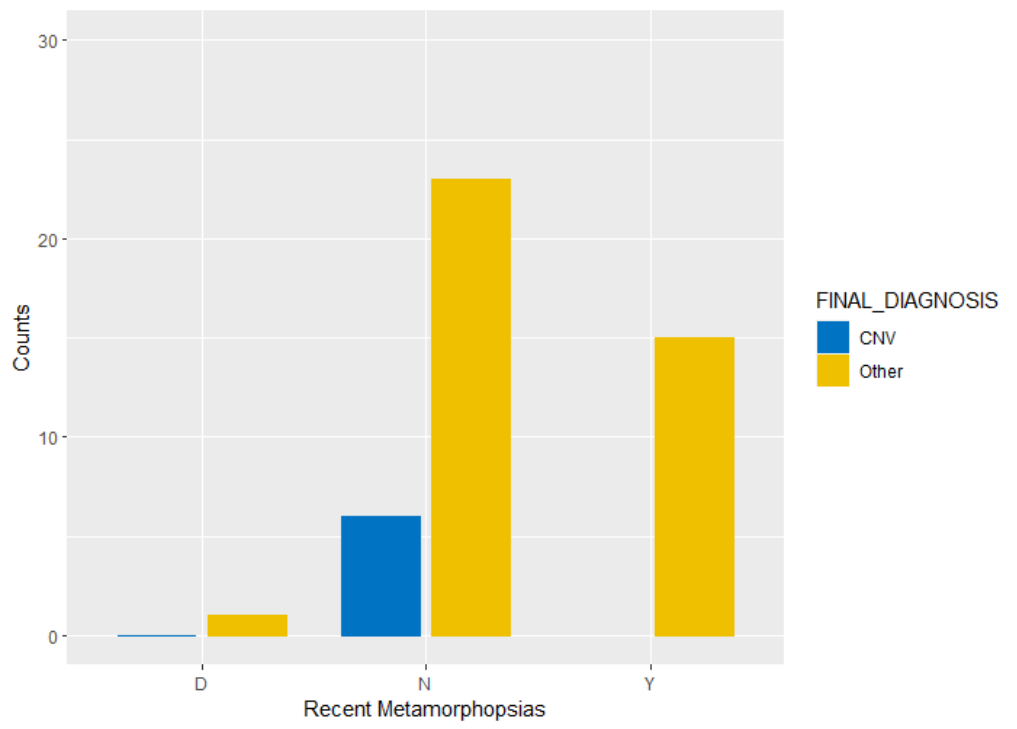


Grafico 2

```
#grafico2
table(data$FINAL_DIAGNOSIS)
df2 <- data %>%
 group_by(RECENT_METAMORPHOPSIAS, FINAL_DIAGNOSIS) %>%
 summarise(counts = n())
head(df2,4)
ap2=data.frame("D","CNV",0)
colnames(ap5)=colnames(df2)
df2 = rbind(df2,ap2 )
grafico2 <- ggplot(df5, aes(x = RECENT_METAMORPHOPSIAS, y =
counts)) +
 geom_bar(
  aes(color = FINAL_DIAGNOSIS, fill = FINAL_DIAGNOSIS),
  stat = "identity", position = position_dodge(0.8),
  width = 0.7,
 ) +
 scale_color_manual(values = c("#0073C2FF", "#EFC000FF"))+
 scale_fill_manual(values = c("#0073C2FF", "#EFC000FF"))+
 scale_x_discrete("Recent Metamorphopsias")+
 scale_y_continuous("Counts", limits = c(0,25))
grafico2
```

EDA

Istogramma Fuzzy 1

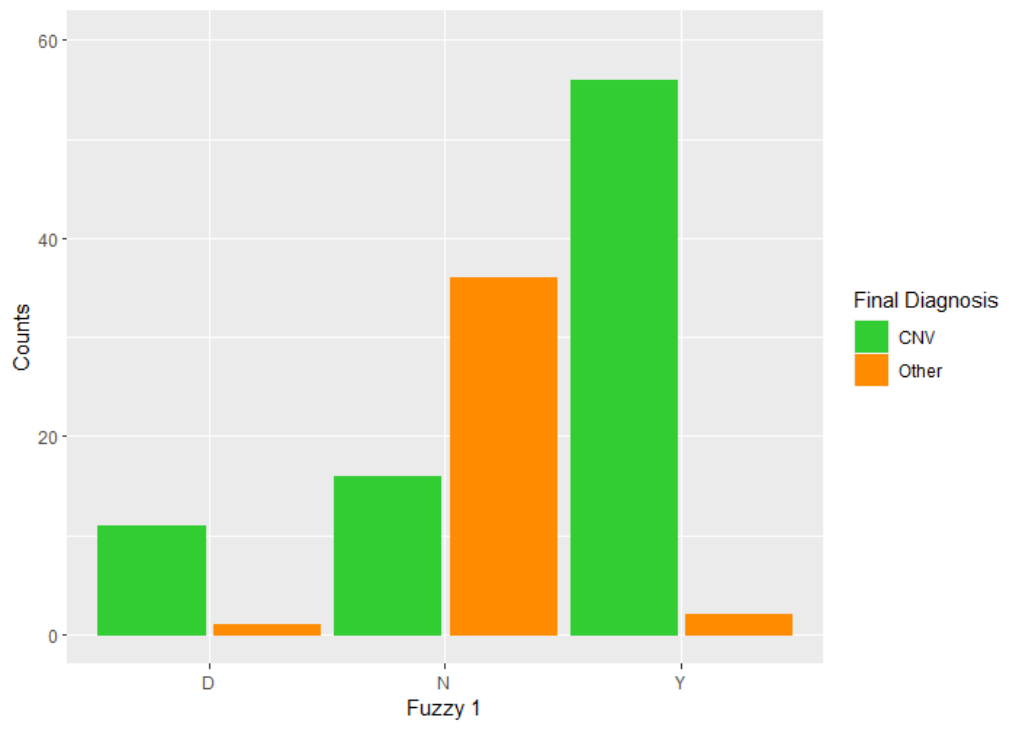


Grafico 3

Istogramma Fuzzy 2

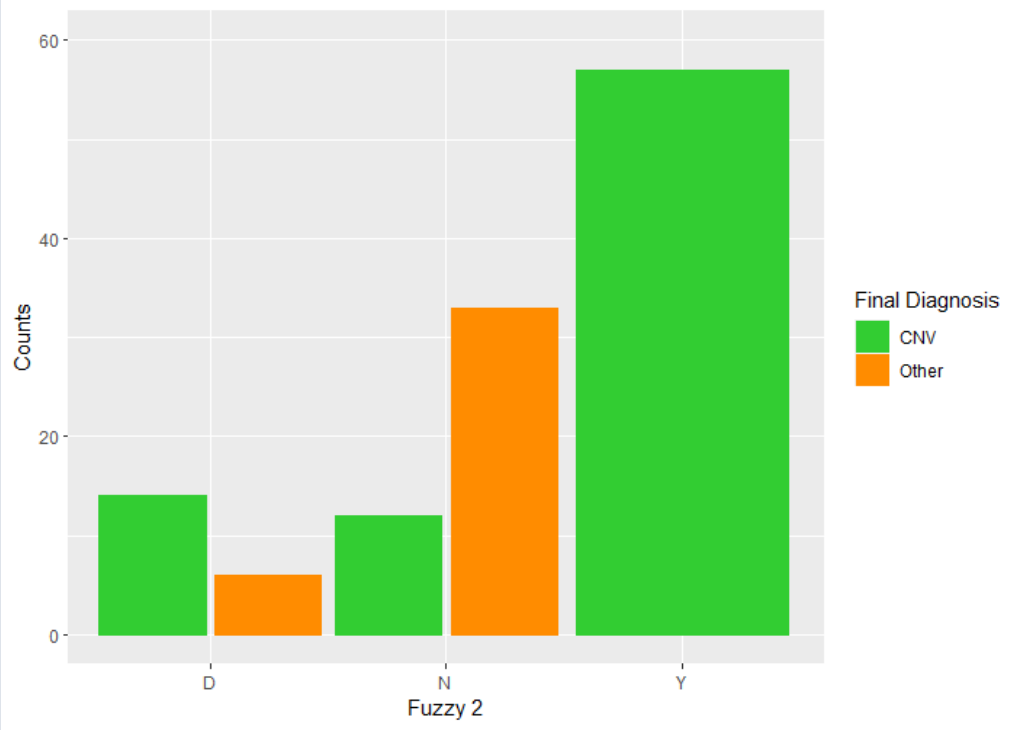


Grafico 4

EDA

Istogramma Shadow 1

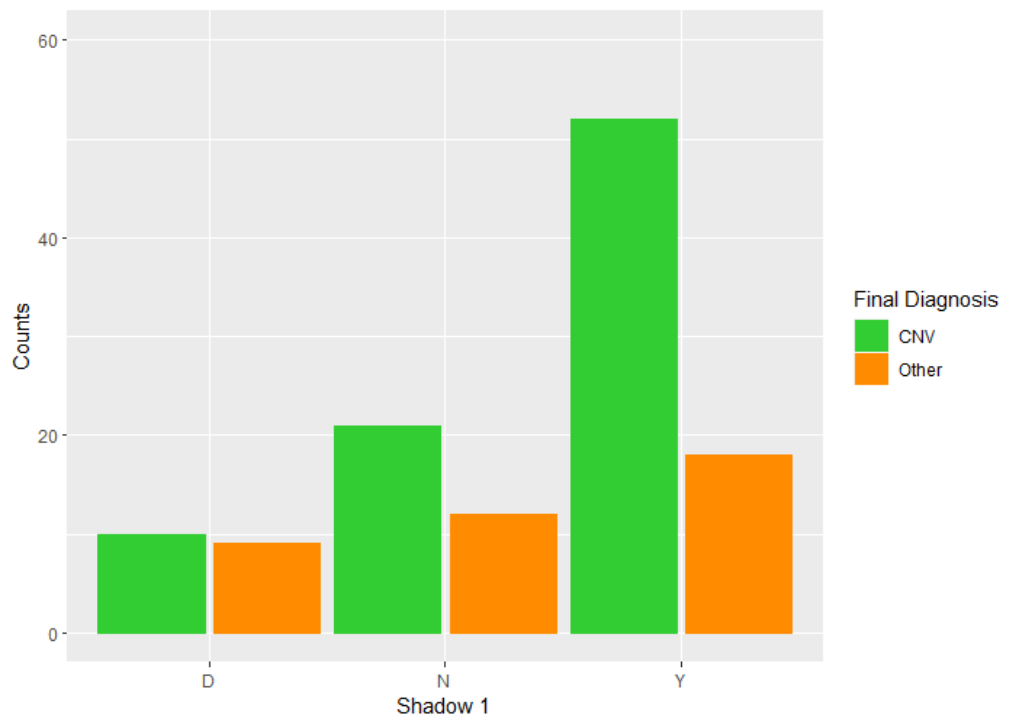


Grafico 5

Istogramma Shadow 2

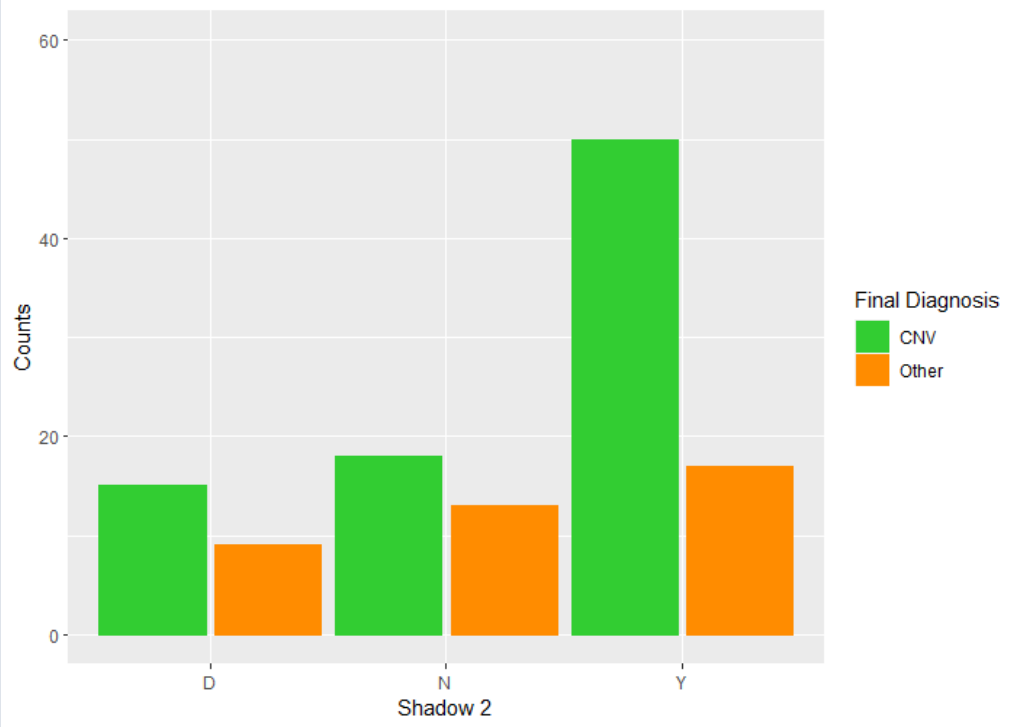


Grafico 6

EDA

Istogramma Leak Ste 1

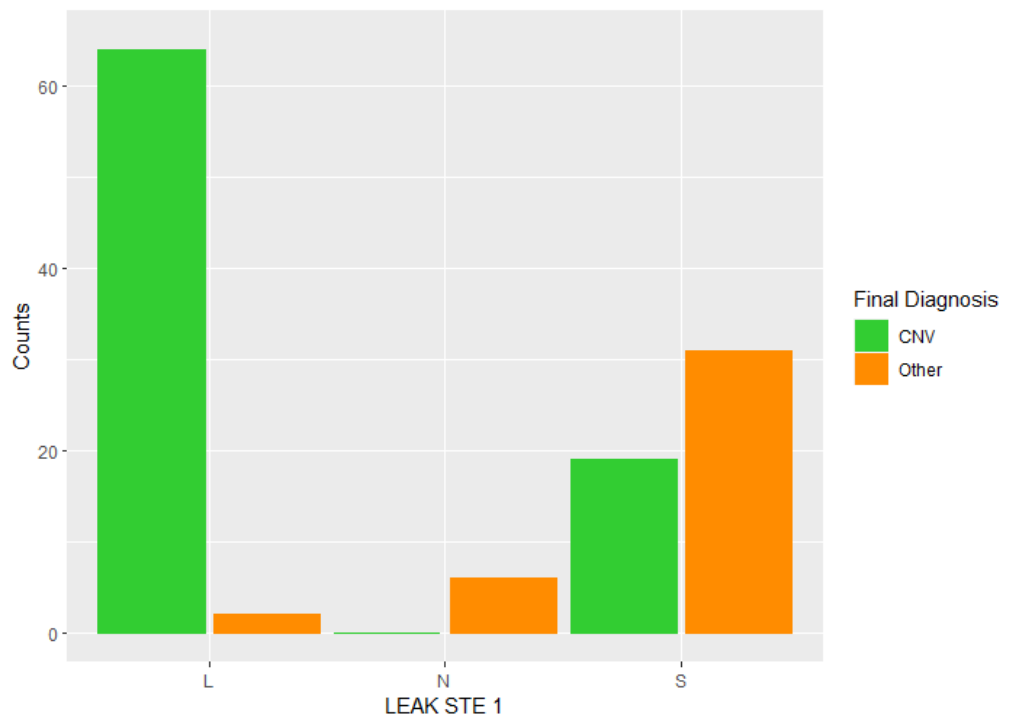


Grafico 7

Istogramma Leak Ste 2

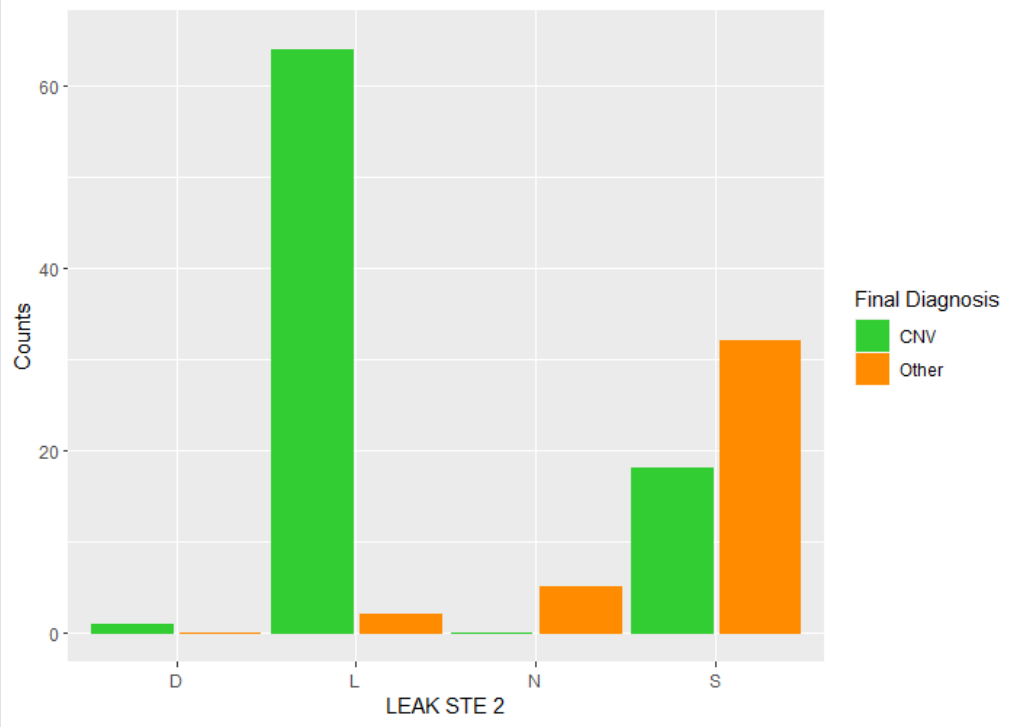


Grafico 8

EDA

Tabella di contingenza

	CNV (N=83)	Other (N=39)	Total (N=122)
PERSISTENT CHROID ON SD OCT 1			
D	3 (3.6%)	4 (10.3%)	7 (5.7%)
N	0 (0%)	2 (5.1%)	2 (1.6%)
Y	80 (96.4%)	33 (84.6%)	113 (92.6%)
PERSISTENT CHROID ON SD OCT 2			
D	0 (0%)	5 (12.8%)	5 (4.1%)
N	1 (1.2%)	2 (5.1%)	3 (2.5%)
Y	82 (98.8%)	32 (82.1%)	114 (93.4%)
EZ 2			
D	5 (6.0%)	1 (2.6%)	6 (4.9%)
I	26 (31.3%)	7 (17.9%)	33 (27.0%)
N	52 (62.7%)	27 (69.2%)	79 (64.8%)
NI	0 (0%)	4 (10.3%)	4 (3.3%)

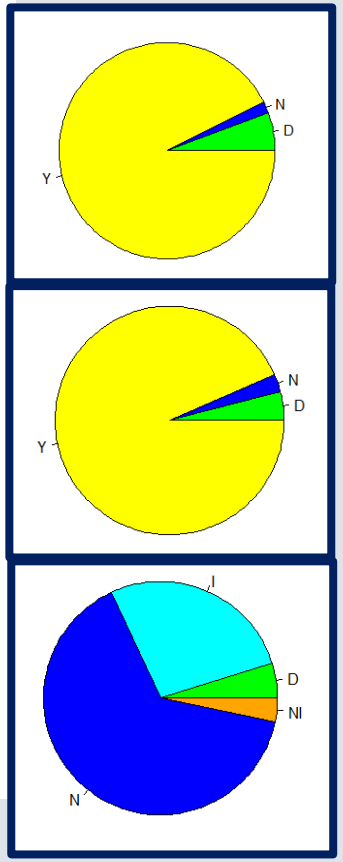


Tabella 2

A.A 2021/2022

Feature Selection

Boruta è una libreria che al suo interno sfrutta l'algoritmo Random Forest.

Per ogni variabile del dataset ne viene creata una copia, sulla quale vengono applicate delle permutazioni casuali.
Queste nuove variabili randomizzate **(shadow feature)** hanno perso, quindi, ogni tipo di correlazione con la variabile target.

L'idea è quella che, affinché una feature sia rilevante, la sua misura di importanza deve essere maggiore della massima misura di importanza ottenuta da queste variabili randomizzate.

A.A 2021/2022

Feature Selection

Il processo di selezione appena descritto viene allora iterato.

Dato che ogni singola iterazione ha esito binario: feature tenuta o feature rifiutata, l'esito di n iterazioni segue una distribuzione Binomiale.

Viene applicato un test statistico al fine di saggiare l'ipotesi nulla che: l'importanza della feature originale sia uguale all'importanza massima ottenuta dalle shadow feature.

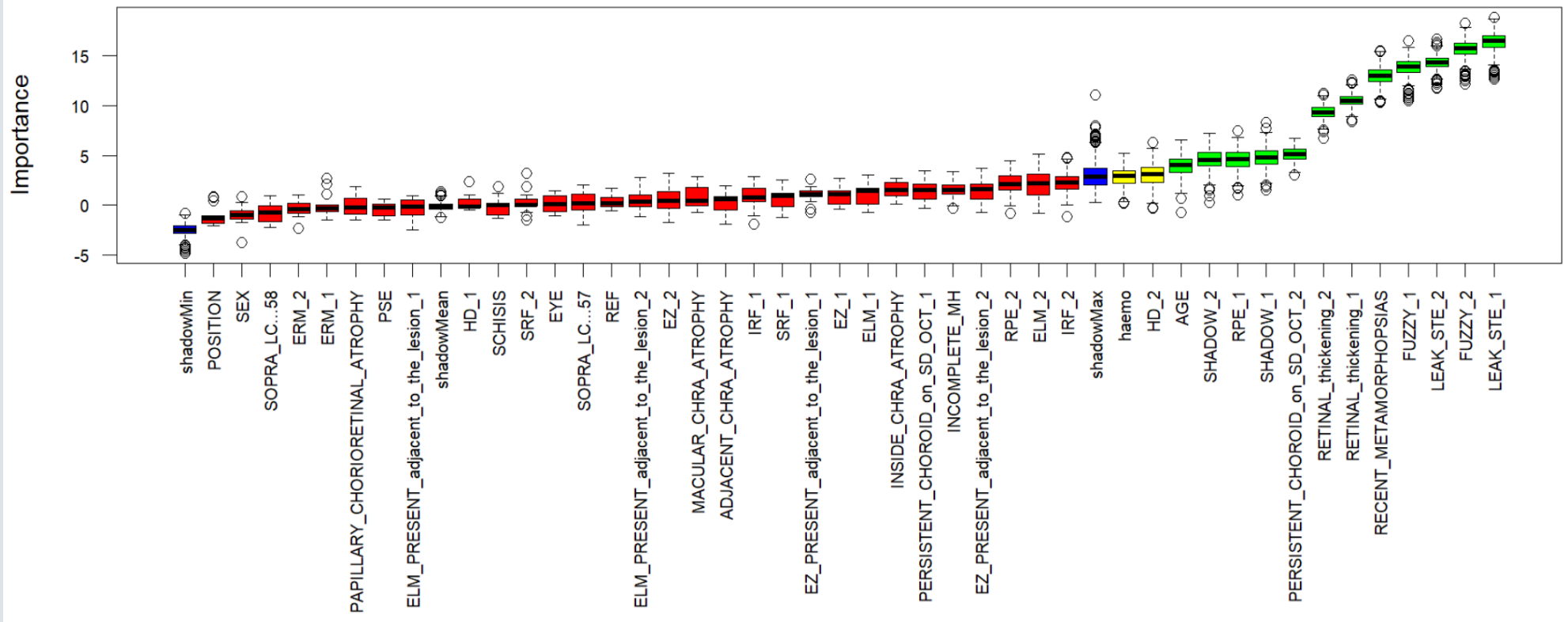
A.A 2021/2022

Feature Selection

```
boruta <- Boruta(FINAL_DIAGNOSIS ~ ., data = data, doTrace = 2,
                 maxRuns = 500)
print(boruta)
plot(boruta, las = 2, cex.axis = 0.7, xlab="")

importance_boruta =colnames(data)[which(boruta$finalDecision=="Confirmed")]
importance_borutadata_fs = data[ , which(names(data) %in% c(importance_boruta,
                 "FINAL_DIAGNOSIS"))]
```

A.A 2021/2022

Feature Selection



A.A 2021/2022

Classification

Dopo aver selezionato le feature più utili a discriminare un paziente affetto da CNV, si è deciso di testare empiricamente la bontà di tale procedura.

Si è quindi deciso addestrare due diversi classificatori prima sul dataset completo, e poi sul sottoinsieme di colonne estratte dalla feature selection, in modo da poter osservare la variazione delle performance.

I due classificatori scelti sono stati:

- RandomForest
- Multi Layer Perceptron

In entrambi i casi il dataset è stato diviso in due porzioni, in modo da poter addestrare il modello sul training-set(70%) e analizzare le performance sul test-set(30%).

Purtroppo la scarsa quantità di osservazioni ha impedito la possibilità di utilizzare una strategia di cross-validation, la quale avrebbe potuto fornire risultati più robusti.

A.A 2021/2022

Classification

Per quanto riguarda il **RandomForest** si è scelto di utilizzare la versione della library «randomForest» con i seguenti parametri:

```
rf_model=randomForest(FINAL_DIAGNOSIS~.,data=data,subset=train, mtry=7)
yhat_rf = predict(rf_model ,newdata=data[-train ,])

cm_rf <- caret::confusionMatrix(data=as.factor(yhat_rf),
                    reference = as.factor(data$FINAL_DIAGNOSIS)[-train])
draw_confusion_matrix(cm_rf, "RANDOM FOREST - Tutte le colonne")
```

Si è deciso di creare una funzione ad hoc per creare i grafici delle confusion matrix:

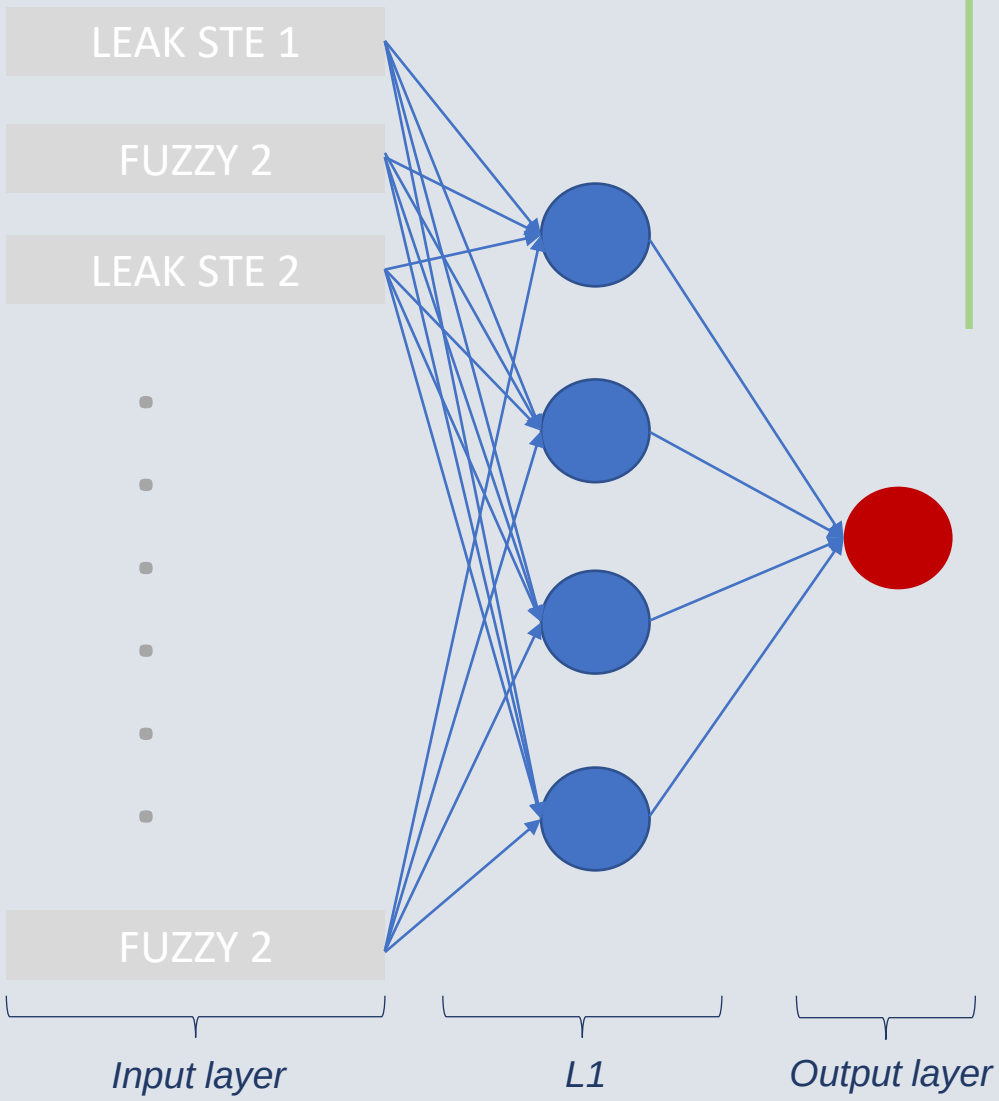
```
draw_confusion_matrix <- function(cm, title) {

  layout(matrix(c(1,1,2)))
  par(mar=c(2,2,2,2))
  plot(c(100, 345), c(300, 450), type = "n", xlab="", ylab="", xaxt='n', yaxt='n')
  title(title, cex.main=2)

  rect(150, 430, 240, 370, col='#3F97D0')
  text(195, 435, 'CNV', cex=1.2)
  rect(250, 430, 340, 370, col='#F7AD50')
  text(295, 435, 'Other', cex=1.2)
  text(125, 370, 'Predicted', cex=1.3, srt=90, font=2)
  text(245, 450, 'Actual', cex=1.3, font=2)
  rect(150, 305, 240, 365, col='#F7AD50')
  rect(250, 305, 340, 365, col='#3F97D0')
  text(140, 400, 'CNV', cex=1.2, srt=90)
  text(140, 335, 'Other', cex=1.2, srt=90)

  res <- as.numeric(cm$table)
  text(195, 400, res[1], cex=1.6, font=2, col='white')
  text(195, 335, res[2], cex=1.6, font=2, col='white')
  text(295, 400, res[3], cex=1.6, font=2, col='white')
  text(295, 335, res[4], cex=1.6, font=2, col='white')

  plot(c(100, 0), c(100, 0), type = "n", xlab="", ylab="", main = "DETAILS", xaxt='n', yaxt='n')
  text(10, 85, names(cm$byClass[1]), cex=1.2, font=2)
  text(10, 70, round(as.numeric(cm$byClass[1]), 3), cex=1.2)
  text(30, 85, names(cm$byClass[2]), cex=1.2, font=2)
  text(30, 70, round(as.numeric(cm$byClass[2]), 3), cex=1.2)
  text(50, 85, names(cm$byClass[5]), cex=1.2, font=2)
  text(50, 70, round(as.numeric(cm$byClass[5]), 3), cex=1.2)
  text(70, 85, names(cm$byClass[6]), cex=1.2, font=2)
  text(70, 70, round(as.numeric(cm$byClass[6]), 3), cex=1.2)
  text(90, 85, names(cm$byClass[7]), cex=1.2, font=2)
  text(90, 70, round(as.numeric(cm$byClass[7]), 3), cex=1.2)

  text(30, 35, names(cm$overall[1]), cex=1.5, font=2)
  text(30, 20, round(as.numeric(cm$overall[1]), 3), cex=1.4)
  text(70, 35, names(cm$overall[2]), cex=1.5, font=2)
  text(70, 20, round(as.numeric(cm$overall[2]), 3), cex=1.4)
}
```

A.A 2021/2022

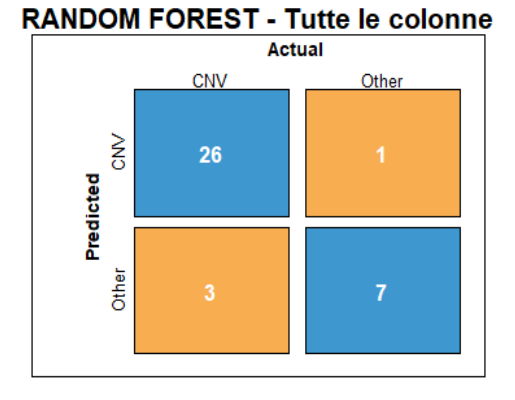
Classification

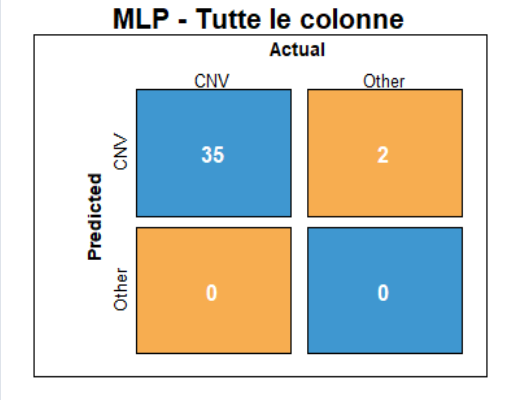
Per l'implementazione del **Multi Layer Perceptron** si è scelto di utilizzare la versione fornita dalla library «neuralnet»
Tutte le variabili categoriali sono state ricodificate tramite One Hot Encoding in modo da poter essere date in input alla rete neurale.
Quest'ultima è passata da una breve fase di fine tuning cha ha portato alla scelta dei seguenti parametri:

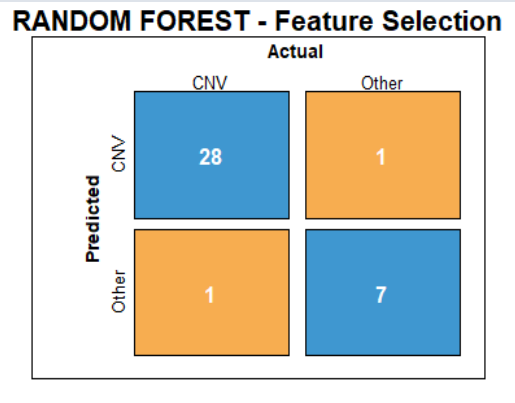
```
nmodel <- neuralnet(f,data=bnk_matrix[train, ],hidden=c(4),
                    threshold = 0.01,
                    rep=5,
                    learningrate=0.01,
                    learningrate.limit = NULL,
                    learningrate.factor = NULL,
                    algorithm = "rprop+")


output <- compute(nmodel, bnk_matrix[-train,-c(1,62)],rep=1)
pred = as.factor(as.numeric(output$net.result>0.5))
true_label = as.factor(bnk_matrix[-train,62])
cm_mlp <- caret::confusionMatrix(data = pred, reference = true_label)
draw_confusion_matrix(cm_mlp, "MLP - Tutte le colonne")
```

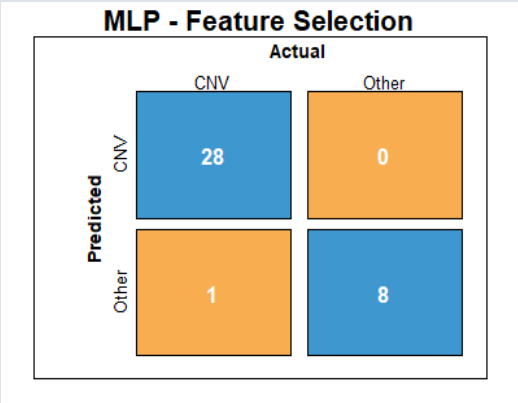


Classification







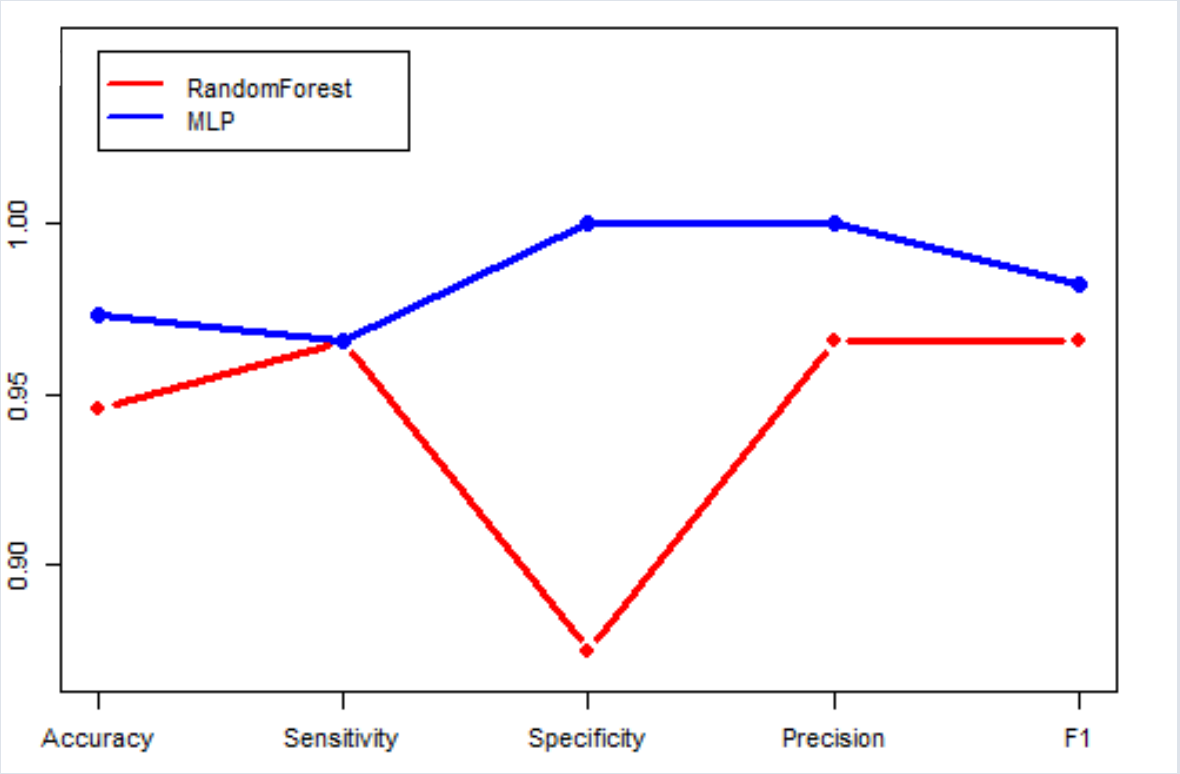


In entrambi i casi, si nota che la feature selection porta ad un miglioramento in termini di performance.

A.A 2021/2022

Classification

Nel seguente grafico si mostrano le **performance** complete dei due classificatori utilizzati:



A.A 2021/2022

Grazie per l'attenzione

Delmiglio Claudia - 830882

Gadda Alberto - 824029