Progetto di Biostatistica

Simone Farallo 889719 & Michele Salvaterra 891109

Abstract

Il nostro progetto ha come obiettivo l'analisi di un dataset riguardante dei pazienti anziani. La prima fase del progetto è stata dedicata alla pulizia dei dati e al loro processing, al fine di rendere il dataset utilizzabile per le successive fasi dell'analisi.

Successivamente, è stata eseguita un'analisi esplorativa dei dati (EDA), al fine di individuare eventuali tendenze o correlazioni tra le variabili del dataset. Questa fase ha permesso di acquisire una migliore comprensione delle caratteristiche dei pazienti anziani in oggetto e di individuare le variabili più significative.

Infine, è stata effettuata un'analisi di sopravvivenza, al fine di valutare la relazione tra le variabili del dataset e la sopravvivenza dei pazienti.

Introduction

La nostra analisi riguarda un dataset di 64 osservazioni riguardanti pazienti anziani. Dopo aver effettuato la pulizia e il preprocessing dei dati, abbiamo eseguito un'analisi esplorativa per identificare le variabili più importanti per la salute degli anziani. In particolare, abbiamo individuato SOFA e CCscore come le variabili più rilevanti per l'analisi della sopravvivenza. Queste variabili sono state utilizzate per valutare il rischio di disfunzione di organi e la gravità della malattia nei pazienti anziani.

Il punteggio "SOFA"(Sequential Organ Failure Assessment) che è uno strumento utilizzato per valutare la gravità dell'insufficienza di uno o più organi in pazienti in terapia intensiva. Esso viene utilizzato come strumento di valutazione per monitorare il miglioramento o il deterioramento della funzione dell'organo nel tempo.

Il CCscore, invece, è un indice di comorbidità utilizzato per valutare la presenza di altre malattie croniche o condizioni preesistenti in un paziente. Il punteggio CCscore viene utilizzato per valutare la gravità della malattia e il rischio di mortalità in pazienti critici.

L'EDA (Exploratory Data Analysis) è una fase fondamentale dell'analisi dei dati che prevede l'esplorazione del dataset per comprendere le caratteristiche e le relazioni tra le variabili, in questo specifico dataset l'EDA ha incluso:

Analisi univariata: esaminare ogni variabile singolarmente per capire la loro distribuzione, valori anomali e statistiche riassuntive.

Analisi bivariata: esaminare la relazione tra le diverse coppie di variabili per capire se esiste una correlazione o associazione tra di esse, abbiamo esaminato la relazione tra SOFA e CC-score, tra età e BMI e tra anno e MMSE-score.

Analisi di sopravvivenza: in questo caso, l'EDA include la creazione di grafici di Kaplan-Meier per visualizzare la sopravvivenza dei pazienti e il modello di Cox.

Infine, per quanto riguarda la presentazione dei risultati, abbiamo fatto in modo che il lettore possa comprendere facilmente i risultati ottenuti.

Variabile	Tipologia	Descrizione
PAZIENTE	*Numeric*	ID del paziente.
NASCITA	*Character*	Data di nascita del paziente.
SEX	*Numeric*	Sesso del paziente (1 = uomo ; 2 = donna).
STATOCIV	*Numeric*	Stato civile del paziente(1 = non sposato/a senza partner ;2 = coniugato/a; 3 = convivente; 4 = separato/divorziato; 5 = vedovo/a).
PESO	*Numeric*	Peso del paziente (Kg).
ALTEZZA	*Numeric*	Altezza del paziente (Cm).
CADUTE	*Numeric*	Numero di cadute del paziente.
CCSCORE	*Numeric*	Il Charlson Comorbidity Score è un indice di gravità della malattia che viene utilizzato per valutare il rischio di mortalità e morbilità associato a comorbidità in pazienti con malattie croniche.
SOFAING	*Numeric*	Il punteggio SOFA (Sequential Organ Failure Assessment) è un indice utilizzato per valutare la gravità della disfunzione di vari organi nei pazienti.
MMSE	*Character*	Il Mini-Mental State Examination (MMSE) è uno strumento di valutazione cognitiva ampiamente utilizzato per valutare la funzione cognitiva nei pazienti (-1 = Mancante ; -2 = Non valutabile).
ALB	*Character*	Albumina
CALC	*Character*	Calcio
VITD	*Character*	Vitamina D
HBING	*Character*	Emoglobina
TEMPRIC	*Numeric*	Durata del ricovero del paziente in minuti (-1 = Non disponibile ; -2 = Da altro ospedale)
DATDIM	*POSIXct*	Data dimissione del paziente.
DATINT	*POSIXct*	Data intervento del paziente.
INTDURAT	*Numeric*	Durata dell'intervento in minuti.

Variabile	Tipologia	Descrizione
ANEST	*Numeric*	Tipo di anestesia utilizzata (1 = generale; 2 = spinale; 3 = peridurale; 4 = plessica; 5 = combinata; 6 = sedazione; 7 = locale assisitita; 8 = altro)
DATA DECESSO	*Numeric*	Data decesso del paziente.

Preliminary Operations

Carichiamo le librerie utili alle nostre analisi.

```
#Management
library(readxl)
library(stringr)
#Visualization
library(ggplot2)
library(psych)
```

```
##
## Caricamento pacchetto: 'psych'
```

```
## I seguenti oggetti sono mascherati da 'package:ggplot2':
##
##     %+%, alpha
```

```
#Descriptive
library(table1)
```

```
##
## Caricamento pacchetto: 'table1'
```

```
## I seguenti oggetti sono mascherati da 'package:base':
##
##     units, units<-
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
#Analisi di sopravvivenza
library(survival)
library(survminer)
```

```
## Caricamento del pacchetto richiesto: ggpubr
```

```
##
## Caricamento pacchetto: 'survminer'
```

```
## Il seguente oggetto è mascherato da 'package:survival':
##
##     myeloma
```

```
library(ggfortify)
```

Impostiamo la directory.

```
setwd("C:/Users/Simone/Documents/Desktop/EDA_SOFA-Biostatistic")
```

Data loading

Carichiamo i dati.

```
data <- read_excel("SOFA.xlsx")
head(data)
```

```
## # A tibble: 6 x 20
##   PAZIENTE NASCITA   SEX STATCIV  PESO ALTEZZA CADUTE CCSCORE SOFAING MMSE
##      <dbl> <chr>   <dbl>   <dbl> <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <chr>
## 1        1 14816       1       2    75     170      1       5       0 22,3
## 2        2 10527       2       2    40     135      2       2       2 27.7
## 3        5 10264       1       2    70     170      1       3       2 0
## 4       11 8085        2       5    45     158      1       3       0 17.399999~
## 5       14 8869        2       5    67     170      2       0       0 29,8
## 6       16 6967        2       5    50     150      2       3       0 0
## # ... with 10 more variables: ALB <chr>, CALC <chr>, VITD <chr>, HBING <chr>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

Struttura del dataset.

```
#Struttura del dataset
str(data)
```

```
## tibble [64 x 20] (S3: tbl_df/tbl/data.frame)
##  $ PAZIENTE    : num [1:64] 1 2 5 11 14 16 16 20 22 23 ...
##  $ NASCITA     : chr [1:64] "14816" "10527" "10264" "8085" ...
##  $ SEX         : num [1:64] 1 2 1 2 2 2 2 1 2 1 ...
##  $ STATCIV     : num [1:64] 2 2 2 5 5 5 2 2 2 2 ...
##  $ PESO        : num [1:64] 75 40 70 45 67 50 57 85 60 62 ...
##  $ ALTEZZA     : num [1:64] 170 135 170 158 170 150 165 180 165 179 ...
##  $ CADUTE      : num [1:64] 1 2 1 1 2 2 1 1 2 2 ...
##  $ CCSCORE     : num [1:64] 5 2 3 3 0 3 5 3 4 3 ...
##  $ SOFAING     : num [1:64] 0 2 2 0 0 0 0 1 0 3 ...
##  $ MMSE        : chr [1:64] "22,3" "27.7" "0" "17.399999999999999" ...
##  $ ALB         : chr [1:64] "3,6" "3.37" "3" "2.6" ...
##  $ CALC        : chr [1:64] "9,9" "9.6" "9" "7.3" ...
##  $ VITD        : chr [1:64] "-1" "3.1" "9.1" "3" ...
##  $ HBING       : chr [1:64] "13,2" "14.1" "13" "13.3" ...
##  $ TEMPRIC     : num [1:64] 55 60 120 60 144 120 220 450 200 270 ...
##  $ DATDIM      : POSIXct[1:64], format: "2011-11-02" NA ...
##  $ DATINT      : POSIXct[1:64], format: "2011-10-19" "2012-01-13" ...
##  $ INTDURAT    : num [1:64] 40 95 35 100 80 55 140 100 50 75 ...
##  $ ANEST       : num [1:64] 2 2 4 5 1 1 1 4 4 1 ...
##  $ DATA DECESSO: chr [1:64] NA "40927" "41010" "40947" ...
```

Si può notare come ci sia la presenza di alcuni formati errati e valori anomali.

```
#Statistiche descrittive
summary(data)
```

```
##     PAZIENTE       NASCITA               SEX           STATCIV
##  Min.   :  1.0   Length:64          Min.   :1.000   Min.   :1.000
##  1st Qu.: 36.5   Class :character   1st Qu.:2.000   1st Qu.:2.000
##  Median : 78.0   Mode  :character   Median :2.000   Median :5.000
##  Mean   : 82.2                      Mean   :1.844   Mean   :3.656
##  3rd Qu.:127.5                      3rd Qu.:2.000   3rd Qu.:5.000
##  Max.   :160.0                      Max.   :2.000   Max.   :5.000
##
##       PESO       ALTEZZA          CADUTE          CCSCORE         SOFAING
##  Min.   :-1   Min.   : -1.0   Min.   :-1.000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:50   1st Qu.:154.0   1st Qu.: 1.000   1st Qu.:0.750   1st Qu.:0.0000
##  Median :60   Median :160.0   Median : 2.000   Median :3.000   Median :0.0000
##  Mean   :59   Mean   :155.2   Mean   : 1.625   Mean   :2.562   Mean   :0.5156
##  3rd Qu.:70   3rd Qu.:165.0   3rd Qu.: 2.000   3rd Qu.:4.000   3rd Qu.:1.0000
##  Max.   :95   Max.   :180.0   Max.   : 2.000   Max.   :6.000   Max.   :4.0000
##  NA's   :1    NA's   :1
##      MMSE               ALB               CALC              VITD
##  Length:64          Length:64          Length:64          Length:64
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     HBING              TEMPRIC          DATDIM
##  Length:64          Min.   : -2.0   Min.   :2011-10-14 00:00:00
##  Class :character   1st Qu.: 60.0   1st Qu.:2011-11-17 12:00:00
##  Mode  :character   Median : 90.0   Median :2012-12-28 00:00:00
##                     Mean   :111.6   Mean   :2012-01-15 18:27:41
##                     3rd Qu.:147.0   3rd Qu.:2012-02-25 06:00:00
##                     Max.   :450.0   Max.   :2012-06-22 00:00:00
##                     NA's   :1       NA's   :12
##      DATINT                         INTDURAT          ANEST
##  Min.   :2011-10-03 00:00:00   Min.   : 30.00   Min.   :1.000
##  1st Qu.:2011-11-20 06:00:00   1st Qu.: 50.00   1st Qu.:1.000
##  Median :2012-01-03 12:00:00   Median : 62.50   Median :2.000
##  Mean   :2012-01-26 00:45:00   Mean   : 70.24   Mean   :2.297
##  3rd Qu.:2012-02-12 18:00:00   3rd Qu.: 85.00   3rd Qu.:4.000
##  Max.   :2015-06-12 00:00:00   Max.   :195.00   Max.   :5.000
##                                NA's   :2
##  DATA DECESSO
##  Length:64
##  Class :character
##  Mode  :character
##
##
##
##
```

Confermiamo la presenza di valori anomali.

```
#Conta il numero di valori mancanti per ciascuna variabile del dataset
colSums(is.na(data))
```

```
##     PAZIENTE      NASCITA          SEX      STATCIV         PESO      ALTEZZA 
##            0            0            0            0            1            1 
##       CADUTE      CCSCORE      SOFAING         MMSE          ALB         CALC 
##            0            0            0            0            0            0 
##         VITD        HBING      TEMPRIC       DATDIM      DATINT     INTDURAT 
##            0            0            1           12            0            2 
##        ANEST DATA DECESSO 
##            0           45
```

Osservando i dati si può notare la presenza di valori mancanti, inoltre nelle distribuzioni delle variabili si notano valori anomali e valori che necessitano di una correzione, da queste premesse iniziamo la fase di cleaning e preprocessing del dataset.

Data Cleaning

Il data cleaning è il processo di identificazione e correzione di errori, incoerenze e dati mancanti all'interno di un dataset, per comodità, data la dimensione del dataset, abbiamo deciso di correggere ogni variabile manualmente.

Paziente

Controlliamo l'eventuale presenza dei duplicati.

```
data$PAZIENTE[duplicated(data$PAZIENTE)]
```

```
## [1] 16
```

```
data[data$PAZIENTE== 16,]
```

```
## # A tibble: 2 x 20
##   PAZIENTE NASCITA   SEX STATCIV  PESO ALTEZZA CADUTE CCSCORE SOFAING MMSE
##      <dbl> <chr>   <dbl>   <dbl> <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <chr>
## 1       16 6967        2       5    50     150      2       3       0 0   
## 2       16 6967        2       2    57     165      1       5       0 25,1
## # ... with 10 more variables: ALB <chr>, CALC <chr>, VITD <chr>, HBING <chr>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

Si nota la presenza di due pazienti che hanno lo stesso numero identificativo, ma i valori delle altre colonne sono diverse, deduciamo che si tratti di due pazienti distinti, dunque procediamo con la correzione.

```
data[7,1] = 17
data[7,1]
```

```
## # A tibble: 1 x 1
##   PAZIENTE
##      <dbl>
## 1       17
```

```
data[data$PAZIENTE==16 | data$PAZIENTE==17,]
```

```
## # A tibble: 2 x 20
##   PAZIENTE NASCITA   SEX STATCIV  PESO ALTEZZA CADUTE CCSCORE SOFAING MMSE
##      <dbl> <chr>   <dbl>   <dbl> <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <chr>
## 1       16 6967        2       5    50     150      2       3       0 0
## 2       17 6967        2       2    57     165      1       5       0 25,1
## # ... with 10 more variables: ALB <chr>, CALC <chr>, VITD <chr>, HBING <chr>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

Nascite

```
data$NASCITA
```

```
##  [1] "14816"      "10527"      "10264"      "8085"       "8869"
##  [6] "6967"       "6967"       "6125"       "12812"      "8907"
## [11] "13770"      "3295"       "8049"       "10320"      "8809"
## [16] "11611"      "11957"      "12371"      "10511"      "11299"
## [21] "9138"       "10044"      "8279"       "5197"       "7286"
## [26] "7982"       "10179"      "10945"      "8121"       "11017"
## [31] "11460"      "6082"       "5307"       "13/07/1829" "13728"
## [36] "11382"      "7513"       "11945"      "10409"      "7319"
## [41] "10404"      "9026"       "16/031928"  "10851"      "4306"
## [46] "9807"       "9931"       "13177"      "8827"       "10345"
## [51] "12042"      "9695"       "9361"       "9162"       "8202"
## [56] "8587"       "9829"       "15119"      "11917"      "7707"
## [61] "04/07/1822" "8105"       "9711"       "8049"
```

Come avevamo già notato le date hanno dei valori insensati, questo è dovuto al fatto che R non legge bene i file excel, inoltre c'è un valore da correggere per il paziente 111 e tre valori che contengono errori di battitura.

```
(i <- data[data$PAZIENTE==111,])
```

```
## # A tibble: 1 x 20
##   PAZIENTE NASCITA     SEX STATCIV  PESO ALTEZZA CADUTE CCSCORE SOFAING MMSE
##      <dbl> <chr>     <dbl>   <dbl> <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <chr>
## 1      111 16/031928     2       2    68     160      2       4       0 4,4
## # ... with 10 more variables: ALB <chr>, CALC <chr>, VITD <chr>, HBING <chr>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

```
data$NASCITA[data$PAZIENTE==111]=sapply(Map(append, strsplit(i$NASCITA,""), after = nc
har(i$NASCITA) - 4, "/"), paste, collapse = "")
data[data$PAZIENTE==111,]
```

```
## # A tibble: 1 x 20
##   PAZIENTE NASCITA      SEX STATCIV  PESO ALTEZZA CADUTE CCSCORE SOFAING MMSE
##      <dbl> <chr>      <dbl>   <dbl> <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <chr>
## 1      111 16/03/1928     2       2    68     160      2       4       0 4,4
## # ... with 10 more variables: ALB <chr>, CALC <chr>, VITD <chr>, HBING <chr>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

Trasformiamo i numeri in data considerando come origine il "1899-12-30"

```
data$NASCITA = as.numeric(data$NASCITA)
```

```
## Warning: NA introdotti per coercizione
```

```
(data$NASCITA = as.Date(x = data$NASCITA,origin = "1899-12-30", ))
```

```
##  [1] "1940-07-24" "1928-10-26" "1928-02-06" "1922-02-18" "1924-04-12"
##  [6] "1919-01-27" "1919-01-27" "1916-10-07" "1935-01-28" "1924-05-20"
## [11] "1937-09-12" "1909-01-07" "1922-01-13" "1928-04-02" "1924-02-12"
## [16] "1931-10-15" "1932-09-25" "1933-11-13" "1928-10-10" "1930-12-07"
## [21] "1925-01-06" "1927-07-01" "1922-08-31" "1914-03-24" "1919-12-12"
## [26] "1921-11-07" "1927-11-13" "1929-12-18" "1922-03-26" "1930-02-28"
## [31] "1931-05-17" "1916-08-25" "1914-07-12" NA           "1937-08-01"
## [36] "1931-02-28" "1920-07-26" "1932-09-13" "1928-06-30" "1920-01-14"
## [41] "1928-06-25" "1924-09-16" NA           "1929-09-15" "1911-10-15"
## [46] "1926-11-06" "1927-03-10" "1936-01-28" "1924-03-01" "1928-04-27"
## [51] "1932-12-19" "1926-07-17" "1925-08-17" "1925-01-30" "1922-06-15"
## [56] "1923-07-05" "1926-11-28" "1941-05-23" "1932-08-16" "1921-02-05"
## [61] NA           "1922-03-10" "1926-08-02" "1922-01-13"
```

Procediamo con la correzione dei tre valori mancanti, che altro non sono che i tre errori di battitura che si possono osservare nel Dataset.

```
data$NASCITA[c(43, 61, 34)] = as.Date(c("1928/03/16","1922/07/04","1929/07/13"))
data$NASCITA
```

```
##  [1] "1940-07-24" "1928-10-26" "1928-02-06" "1922-02-18" "1924-04-12"
##  [6] "1919-01-27" "1919-01-27" "1916-10-07" "1935-01-28" "1924-05-20"
## [11] "1937-09-12" "1909-01-07" "1922-01-13" "1928-04-02" "1924-02-12"
## [16] "1931-10-15" "1932-09-25" "1933-11-13" "1928-10-10" "1930-12-07"
## [21] "1925-01-06" "1927-07-01" "1922-08-31" "1914-03-24" "1919-12-12"
## [26] "1921-11-07" "1927-11-13" "1929-12-18" "1922-03-26" "1930-02-28"
## [31] "1931-05-17" "1916-08-25" "1914-07-12" "1929-07-13" "1937-08-01"
## [36] "1931-02-28" "1920-07-26" "1932-09-13" "1928-06-30" "1920-01-14"
## [41] "1928-06-25" "1924-09-16" "1928-03-16" "1929-09-15" "1911-10-15"
## [46] "1926-11-06" "1927-03-10" "1936-01-28" "1924-03-01" "1928-04-27"
## [51] "1932-12-19" "1926-07-17" "1925-08-17" "1925-01-30" "1922-06-15"
## [56] "1923-07-05" "1926-11-28" "1941-05-23" "1932-08-16" "1921-02-05"
## [61] "1922-07-04" "1922-03-10" "1926-08-02" "1922-01-13"
```

SEX

Per la variabile "SEX" decidiamo di modificare i valori numerici 1 e 2 in due stringhe, utilizzando la documentazione.

```
data$SEX <- ifelse(data$SEX == 1, "uomo", "donna")
data$SEX
```

```
##  [1] "uomo"  "donna" "uomo"  "donna" "donna" "donna" "donna" "uomo"  "donna"
## [10] "uomo"  "uomo"  "donna" "donna" "uomo"  "donna" "uomo"  "donna" "donna"
## [19] "uomo"  "donna" "donna" "donna" "donna" "donna" "donna" "donna" "donna"
## [28] "donna" "donna" "donna" "donna" "donna" "donna" "donna" "donna" "donna"
## [37] "donna" "donna" "donna" "donna" "donna" "uomo"  "donna" "donna" "donna"
## [46] "donna" "donna" "donna" "donna" "uomo"  "donna" "donna" "donna" "donna"
## [55] "donna" "donna" "donna" "donna" "donna" "donna" "donna" "donna" "donna"
## [64] "donna"
```

STATCIV

Come fatto per la variabile precedente, procediamo in modo analogo e utilizzando la documentazione andiamo a modificare i valori numerici.

```
data$STATCIV <- ifelse(data$STATCIV == 1, "non_sposato\a",
                       ifelse(data$STATCIV == 2, "coniugato\a",
                              ifelse(data$STATCIV == 3, "convivente",
                                     ifelse(data$STATCIV == 4, "divorziato\a",
                                            ifelse(data$STATCIV == 5, "vedovo\a",'NA
N'))))))
data$STATCIV
```

```
##  [1] "coniugato\a"   "coniugato\a"   "coniugato\a"   "vedovo\a"
##  [5] "vedovo\a"      "vedovo\a"      "coniugato\a"   "coniugato\a"
##  [9] "coniugato\a"   "coniugato\a"   "coniugato\a"   "vedovo\a"
## [13] "vedovo\a"      "coniugato\a"   "vedovo\a"      "coniugato\a"
## [17] "vedovo\a"      "coniugato\a"   "coniugato\a"   "vedovo\a"
## [21] "non_sposato\a" "coniugato\a"   "vedovo\a"      "vedovo\a"
## [25] "vedovo\a"      "vedovo\a"      "coniugato\a"   "vedovo\a"
## [29] "vedovo\a"      "non_sposato\a" "coniugato\a"   "non_sposato\a"
## [33] "vedovo\a"      "vedovo\a"      "coniugato\a"   "vedovo\a"
## [37] "non_sposato\a" "coniugato\a"   "non_sposato\a" "vedovo\a"
## [41] "vedovo\a"      "coniugato\a"   "coniugato\a"   "vedovo\a"
## [45] "vedovo\a"      "vedovo\a"      "vedovo\a"      "vedovo\a"
## [49] "vedovo\a"      "coniugato\a"   "coniugato\a"   "vedovo\a"
## [53] "vedovo\a"      "vedovo\a"      "vedovo\a"      "vedovo\a"
## [57] "vedovo\a"      "vedovo\a"      "coniugato\a"   "vedovo\a"
## [61] "vedovo\a"      "vedovo\a"      "vedovo\a"      "vedovo\a"
```

Non ci sono valori nulli e tutti i campi sono stati corretti, bisogna dire che questa operazione non è sempre indispensabile.

Peso

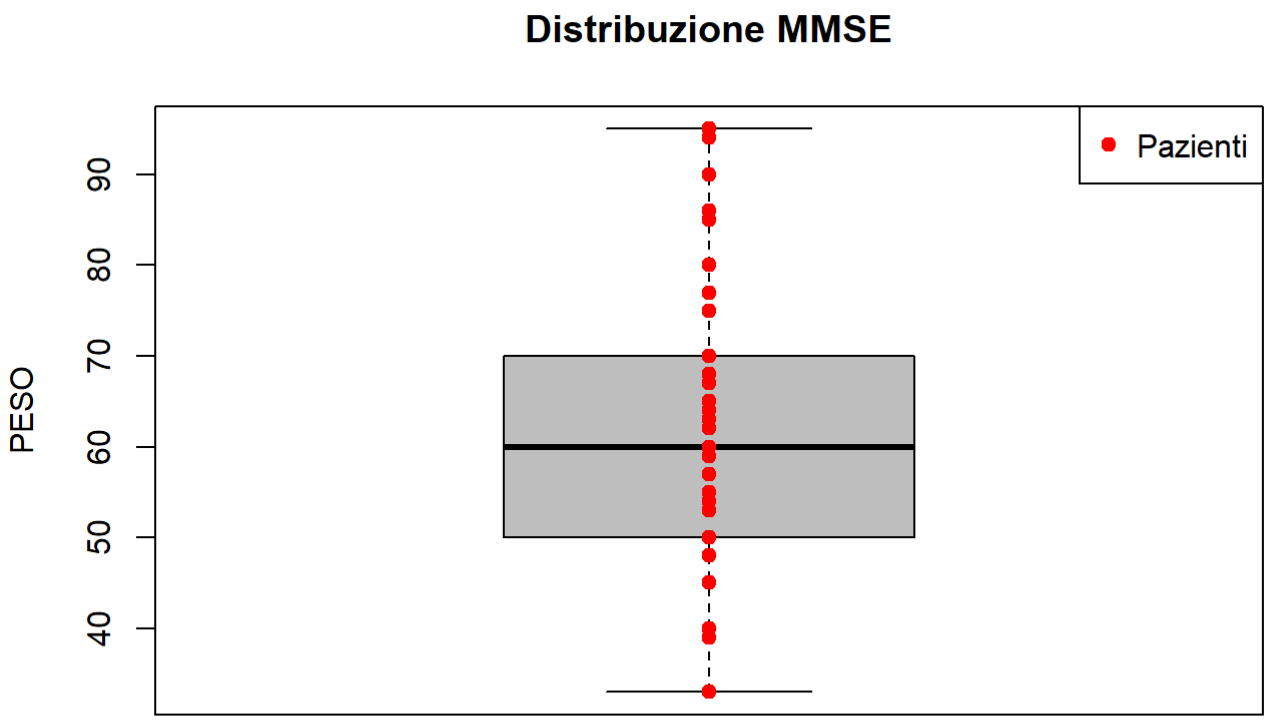
Osservando la distribuzione della variabile “PESO” sono presenti dei valori -1, andiamo a trasformare i valori mancanti e li contiamo come se fossero dei valori nulli.

```
data$PESO =ifelse(data$PESO==-1,NA,data$PESO)
sum(is.na(data$PESO) | data$PESO == "NA")
```

```
## [1] 3
```

Osserviamo la nuova distribuzione.

```
boxplot(data$PESO,
        col = "grey",
        border = "black",
        names = c("PESO"),
        ylab = "PESO",
        pch = 19,
        main= "Distribuzione MMSE")
points(x = rep(1, length(data$PESO)), y = data$PESO, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Notiamo come ci siano alcuni outlier, soprattutto una paziente ha un peso che equivale a 33kg. Osservando però i valori delle altre colonne ed anche l'età avanzata di molti pazienti, come si può vedere facilmente dando uno sguardo al boxplot, abbiamo deciso momentaneamente di non eliminare l'osservazione.

ALTEZZA

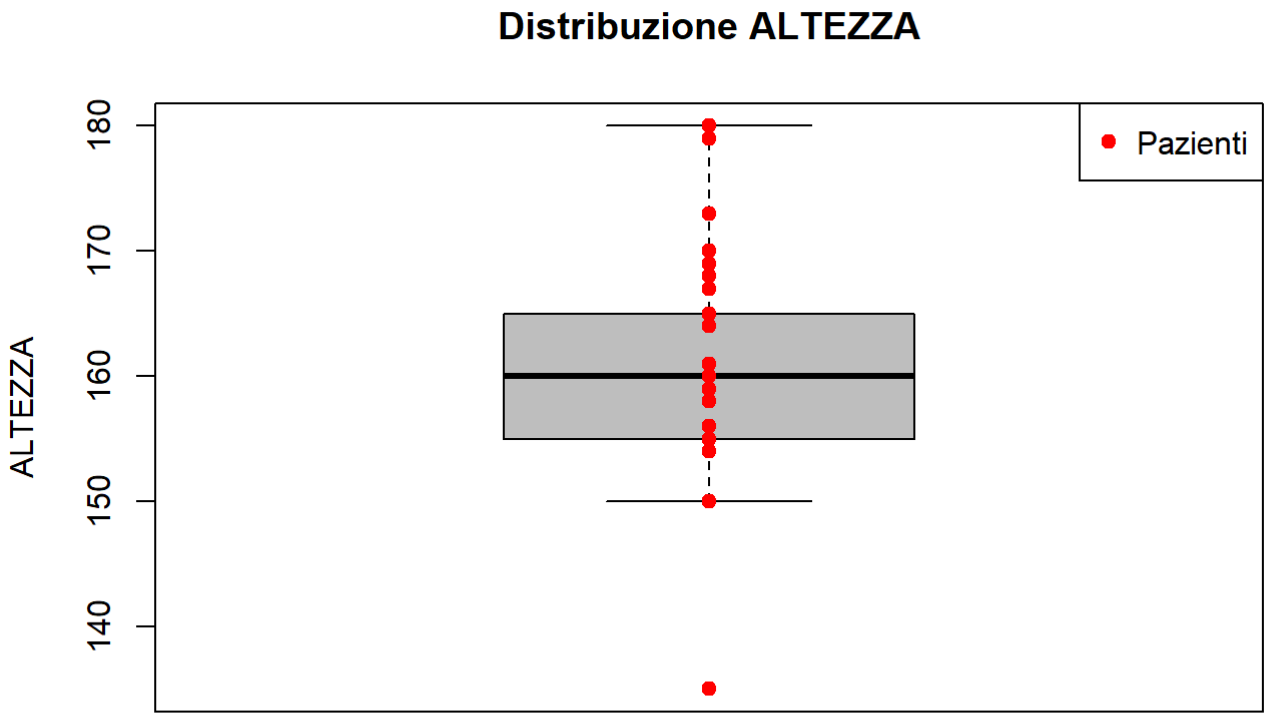
Osservando la variabile "ALTEZZA" si notano due valori "-1" e "9" che necessitano di una correzzione.

```
data$ALTEZZA = ifelse(data$ALTEZZA< 100,NA,data$ALTEZZA)
data$ALTEZZA
```

```
##  [1] 170 135 170 158 170 150 165 180 165 179 158 160 150 165 168  NA 160 169 170
## [20] 161 160 150 164 150 154 155 155 156 173 154 165 150 165 155 165 165 160 159
## [39] 160 150 170  NA 160 150 167 155 165 160 165 168  NA 160 160 155 160 150 160
## [58] 165 150 170 150 160 150 150
```

Osserviamo la distribuzione.

```
boxplot(data$ALTEZZA,
        col = "grey",
        border = "black",
        names = c("ALTEZZA"),
        ylab = "ALTEZZA",
        pch = 19,
        main= "Distribuzione ALTEZZA")
points(x = rep(1, length(data$ALTEZZA)), y = data$ALTEZZA, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



C'è un solo outlier di un paziente alto 135 cm, ma osservando il suo peso e altri valori, momentaneamente non lo eliminiamo.

CADUTE

La variabile "CADUTE" presenta dei valori anomali corrispondenti a -1, procediamo a corregerli.

```
data$CADUTE = ifelse(data$CADUTE<0,NA,data$CADUTE)
data$CADUTE
```

```
##  [1]  1  2  1  1  2  2  1  1  2  2  1  2 NA  2  2  2  2  2  2  2  2  2  1  2  2
## [26]  2  2  1  1  1  2  2  2  1  2  1  1  1  1  2  2  2  2  1  2  2  2  1  2  2
## [51]  2  2  2  1  1  2  2  2  2  2  2  2  1  1
```

CCSCORE

La variabile "CCSCORE" non sembra presentare problemi.

```
data$CCSCORE
```

```
##  [1] 5 2 3 3 0 3 5 3 4 3 6 3 2 3 0 6 3 2 2 6 0 5 5 4 0 1 0 0 3 3 3 0 0 0 3 0 6 1
## [39] 0 4 2 2 4 4 2 0 0 0 2 0 2 3 3 4 1 4 4 0 4 5 1 5 5 5
```

SOFAING

La variabile “SOFAING” non sembra presentare problemi.

```
data$SOFAING
```

```
##  [1] 0 2 2 0 0 0 0 1 0 3 1 1 0 0 0 2 1 0 0 0 0 1 0 0 0 1 0 0 2 0 0 0 0 0 0 0 2 0
## [39] 0 1 1 0 0 1 0 0 0 0 0 2 0 0 0 0 0 4 1 0 0 3 0 1 0 0
```

MMSE

La variabile “MMSE” presenta dei valori corrispondenti a “-1 = mancante” e “-2 = non valutabile”, inoltre alcuni valori sono separati da virgole ed altri da punti, procediamo a corregerli.

```
data$MMSE = ifelse(data$MMSE<0,NA,data$MMSE)
data$MMSE = str_replace(data$MMSE, ",", ".")
data$MMSE = as.numeric(data$MMSE)
```

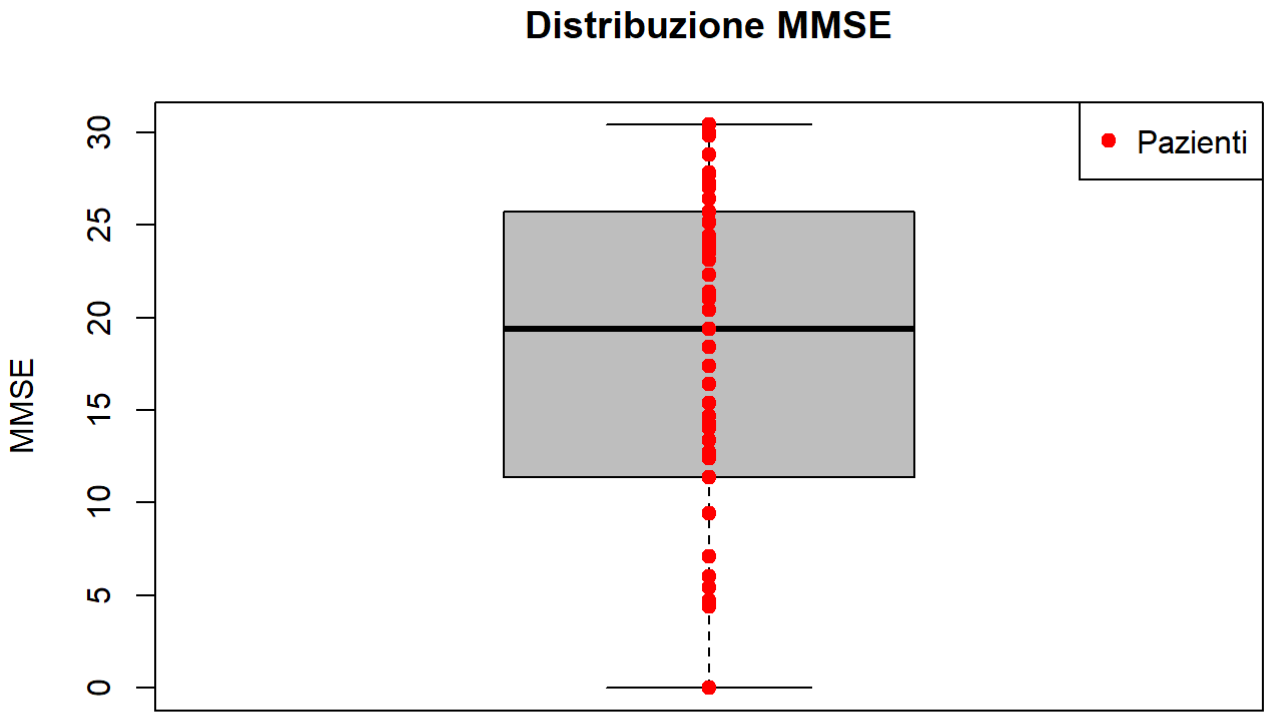
```
## Warning: NA introdotti per coercizione
```

```
data$MMSE
```

```
##  [1] 22.3 27.7  0.0 17.4 29.8  0.0 25.1  5.4  4.7 27.3 14.3  0.0 23.8 24.1 28.8
## [16] 12.7 27.7 27.0 21.1  0.0 16.4 23.5 19.4  0.0 13.4 26.4 17.4 21.4 11.4  7.1
## [31] 23.1 13.4   NA 20.4   NA 30.0 26.4 27.7 26.4 12.4 18.4  9.4  4.4  4.4 15.4
## [46] 24.4 25.7 25.7 30.4 24.1 14.7  6.0 18.4  0.0 21.2   NA 27.8 30.0 14.0 21.0
## [61] 18.4 25.2  0.0  0.0
```

Osserviamo la distribuzione.

```
boxplot(data$MMSE,
        col = "grey",
        border = "black",
        names = c("MMSE"),
        ylab = "MMSE",
        pch = 19,
        main= "Distribuzione MMSE")
points(x = rep(1, length(data$MMSE)), y = data$MMSE, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Notiamo come ci sia un paziente con un valore uguale a “0” che sta a significare un massimo deficit cognitivo.

ALB

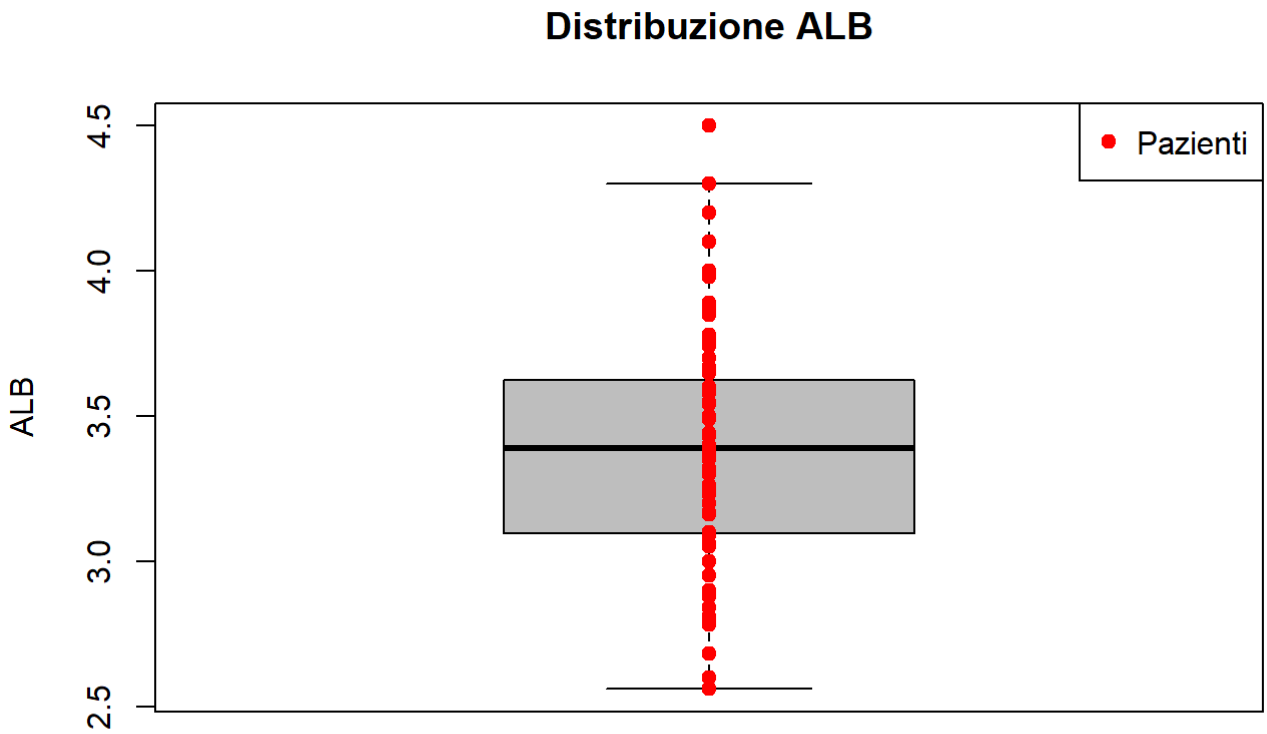
La variabile “ALB” presenta valori “-1” e inoltre alcuni valori sono separati da virgole ed altri da punti, procediamo a formalizzare il tutto.

```
data$ALB = ifelse(data$ALB==-1,NA,data$ALB)
data$ALB = str_replace(data$ALB, ",", ".")
data$ALB = as.numeric(data$ALB)
data$ALB
```

```
##  [1] 3.60 3.37 3.00 2.60 3.10 3.85 4.10 3.00 2.80 3.10 3.39 3.40 3.55 3.58 3.86
## [16] 3.60 4.50 3.98 3.49 3.26 3.89 3.36 3.76   NA 3.09 2.88 4.20 3.74 3.17 4.00
## [31] 3.78 3.40 2.78 2.81 4.30 3.16 3.00 3.44 2.90 2.84 3.70 3.10 3.05 3.30 3.32
## [46] 3.06 3.10 3.65 3.60 3.50 2.95 3.60 3.67 3.23 3.43 3.20 3.54 2.68 3.87 3.25
## [61] 3.50 3.35 3.60 2.56
```

Controlliamo la distribuzione.

```
boxplot(data$ALB,
        col = "grey",
        border = "black",
        names = c("ALB"),
        ylab = "ALB",
        pch = 19,
        main= "Distribuzione ALB")
points(x = rep(1, length(data$ALB)), y = data$ALB, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Notiamo come ci sia molta differenza tra i valori di albumina.

CALC

La variabile “CALC” ha alcuni valori separati da virgole ed altri da punti, procediamo a formalizzare il tutto.

```
data$CALC = str_replace(data$CALC, ",", ".")
data$CALC = as.numeric(data$CALC)
data$CALC
```

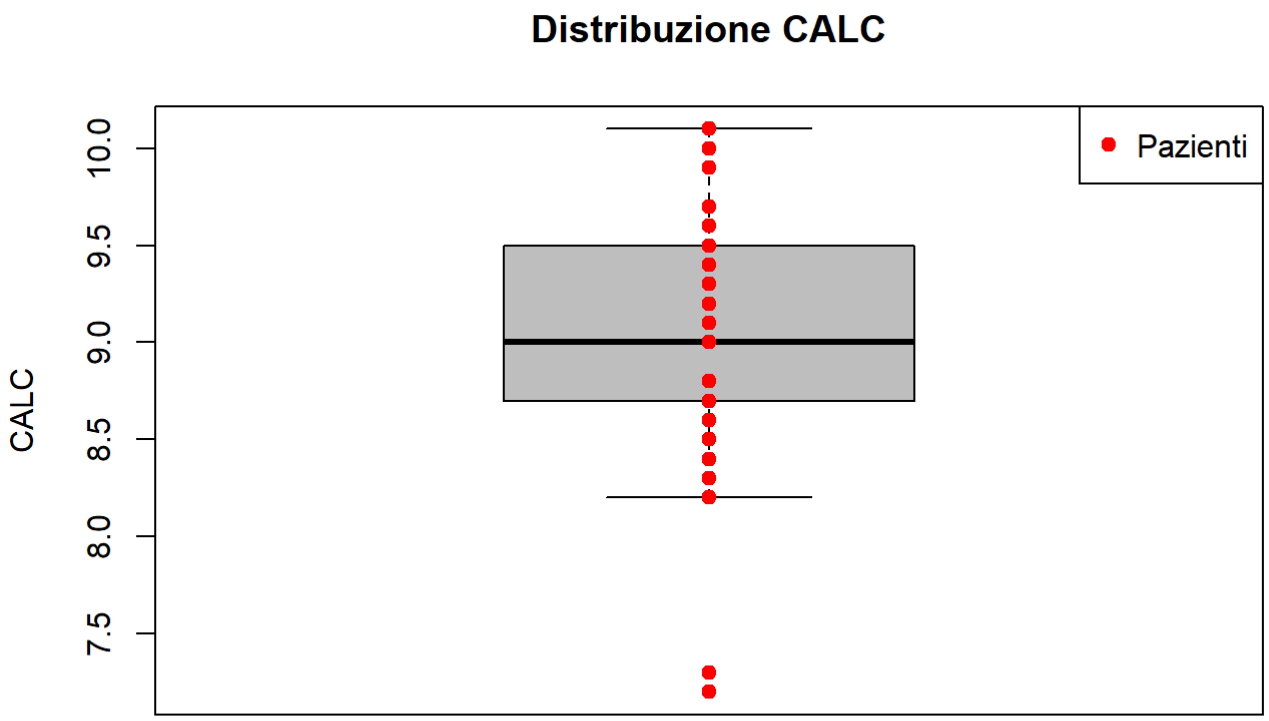
```
##  [1]   9.9   9.6   9.0   7.3   9.2   9.7   9.6   8.6   9.1   7.2   9.5   9.6
## [13]   8.5   8.8  10.1   9.2   9.9   9.4   8.8   8.6   9.4   9.1 880.0   9.1
## [25]   8.8   9.0   9.9   8.7   9.0   9.7   9.5   8.3   8.4   8.7  10.0   9.6
## [37]   9.0   9.7   8.5  10.0   9.0   8.5   9.3   9.1   8.8   9.0   9.7   9.7
## [49]   9.4   9.0   9.3   9.3   8.7   8.6   9.0   8.8   8.8   8.5   9.5   8.8
## [61]   8.5   8.7   8.8   8.2
```

Notiamo un valore pari a 880, probabilmente un errore di battitura, correggiamolo dividendolo per 100.

```
data$CALC[23]= data$CALC[23]/100
```

Osserviamo la distribuzione.

```
boxplot(data$CALC,
        col = "grey",
        border = "black",
        names = c("CALC"),
        ylab = "CALC",
        pch = 19,
        main= "Distribuzione CALC")
points(x = rep(1, length(data$CALC)), y = data$CALC, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Osserviamo la presenza di due outlier con valori molto bassi di calcio.

VITD

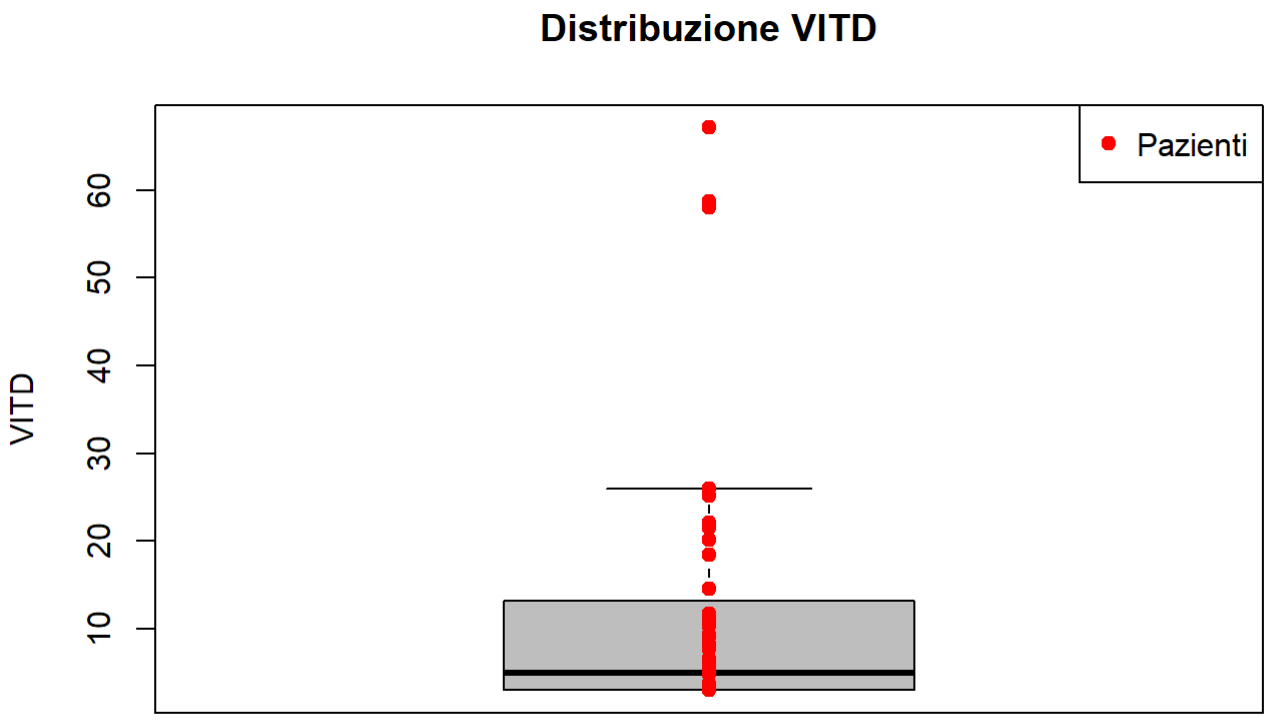
Ci sono diversi valori da correggere come i valori "-1" e sono presenti alcuni valori separati da virgole ed altri da punti, procediamo a formalizzare il tutto.

```
data$VITD = str_replace(data$VITD, ",", ".")
data$VITD = ifelse(data$VITD==-1,NA,data$VITD)
data$VITD = as.numeric(data$VITD)
data$VITD
```

```
##  [1]   NA  3.1  9.1  3.0 58.0   NA 11.7  5.7   NA  3.0   NA   NA   NA 21.5  3.0
## [16] 14.6  7.8 26.0  3.2  3.1  3.0  3.0  3.0  3.0  3.0  3.0   NA  3.3  3.0  9.3
## [31] 18.4   NA  3.0  3.0  3.2 21.6  4.9 25.2  3.0   NA 11.0   NA  8.3  3.7 22.1
## [46]  4.9   NA 58.7   NA  4.9  3.0   NA  3.1  7.7  6.0 21.8  6.6   NA 20.1   NA
## [61]  5.1 10.3   NA 67.1
```

Osserviamo la distribuzione.

```
boxplot(data$VITD,
        col = "grey",
        border = "black",
        names = c("VITD"),
        ylab = "VITD",
        pch = 19,
        main= "Distribuzione VITD")
points(x = rep(1, length(data$VITD)), y = data$VITD, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Osserviamo la presenza di 3 outlier, cioè tre pazienti che rispetto al resto del campione hanno dei valori di Vitamina D molto elevata.

HBING

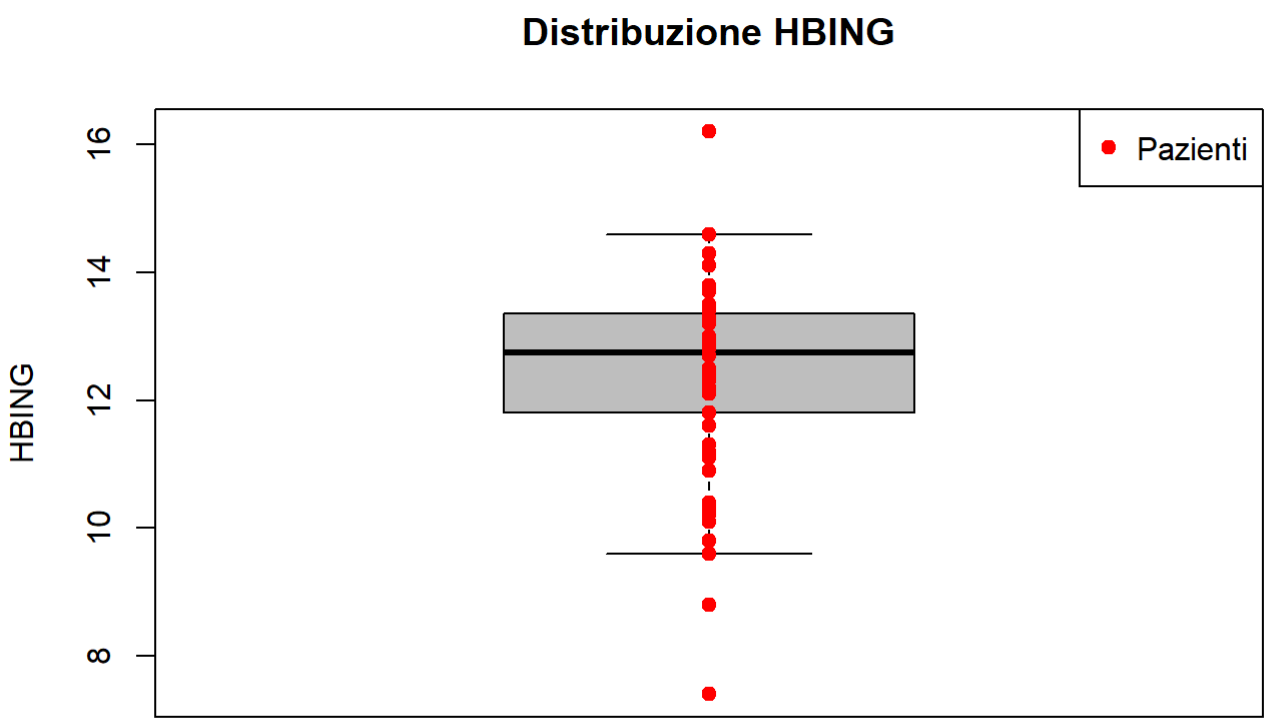
Notiamo che anche qui sono presenti alcuni valori separati da virgole ed altri da punti, procediamo a formalizzare il tutto.

```
data$HBING = str_replace(data$HBING, ",", ".")
data$HBING = as.numeric(data$HBING)
data$HBING
```

```
##  [1] 13.2 14.1 13.0 13.3 11.8 14.3 12.3 12.7 13.0 11.3 13.4 12.7 12.9 12.2 13.3
## [16] 13.0 11.6 12.8 14.6 12.4 12.7 13.3 12.2 16.2 14.3 12.2 13.2 14.1 10.1 12.4
## [31] 13.2 10.9 14.1 12.2 10.3 14.3  8.8 10.1 12.5 12.1 12.1 10.2 10.9 13.5 12.8
## [46] 10.4 13.4 12.7 13.4 12.9 11.8 13.7 12.9  9.6 12.8 14.3 12.2 11.2 14.6  9.8
## [61] 12.9 13.8 11.1  7.4
```

Osserviamo la distribuzione

```
boxplot(data$HBING,
        col = "grey",
        border = "black",
        names = c("HBING"),
        ylab = "HBING",
        pch = 19,
        main= "Distribuzione HBING")
points(x = rep(1, length(data$HBING)), y = data$HBING, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Anche in questo caso ci sono degli outlier sia oltre il limite superiore che inferiore dell'intera distribuzione.

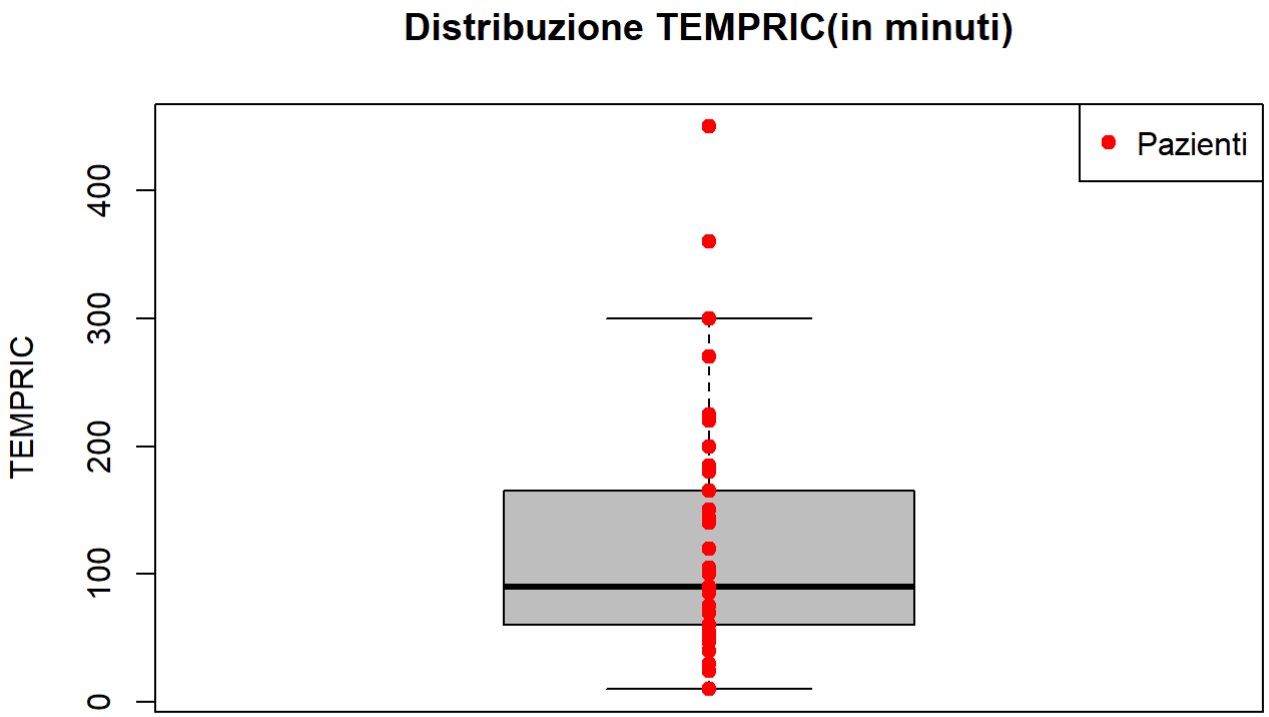
TEMPRIC

Nella variabile “TEMPRIC” ci sono dei “-1” riferiti al valore non disponibile e dei “-2” che indicano la provenienza del paziente da un altro ospedale, procediamo con la correzione trattandoli come “NA”.

```
data$TEMPRIC = ifelse(data$TEMPRIC<0,NA,data$TEMPRIC)
data$TEMPRIC
```

```
##  [1]  55  60 120  60 144 120 220 450 200 270 165  60 180  90 120  70  50  60  NA
## [20]  60 100 225  75 200 120 185  75  40  90  90  60 140 120  10  NA 100 105  60
## [39]  NA  70 300  60 120  75  24 165  75 270  85  90 180  NA  90  70 360  NA  85
## [58] 200  70  30 150  NA  47  70
```

```
boxplot(data$TEMPRIC,
        col = "grey",
        border = "black",
        names = c("TEMPRIC"),
        ylab = "TEMPRIC",
        pch = 19,
        main= "Distribuzione TEMPRIC(in minuti)")
points(x = rep(1, length(data$TEMPRIC)), y = data$TEMPRIC, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



Notiamo come per la maggior parte dei pazienti la durata del ricovero sia compresa tra i 0 e i 300 minuti.

DATADIM

La variabile “DATADIM” non presenta valori da correggere.

```
data$DATDIM
```

```
##  [1] "2011-11-02 UTC" NA               "2011-12-28 UTC" NA
##  [5] "2011-12-01 UTC" "2011-12-09 UTC" "2011-12-16 UTC" "2011-11-04 UTC"
##  [9] "2012-02-29 UTC" NA               "2011-10-24 UTC" "2012-01-10 UTC"
## [13] "2012-06-18 UTC" "2011-11-29 UTC" "2012-06-22 UTC" NA
## [17] "2012-01-05 UTC" "2011-11-10 UTC" "2012-03-09 UTC" NA
## [21] "2011-11-16 UTC" "2012-02-29 UTC" NA               "2012-03-09 UTC"
## [25] NA               "2012-02-24 UTC" "2011-12-12 UTC" "2011-10-14 UTC"
## [29] "2011-12-27 UTC" "2011-11-28 UTC" "2011-11-16 UTC" "2011-12-19 UTC"
## [33] NA               "2012-03-08 UTC" "2011-12-16 UTC" "2012-02-23 UTC"
## [37] NA               "2012-01-03 UTC" "2012-02-23 UTC" "2012-01-10 UTC"
## [41] "2011-11-14 UTC" "2012-06-19 UTC" "2011-11-18 UTC" "2011-10-14 UTC"
## [45] "2011-11-07 UTC" "2012-06-13 UTC" "2011-10-28 UTC" "2012-02-17 UTC"
## [49] "2011-10-28 UTC" "2012-01-12 UTC" "2012-06-19 UTC" "2012-02-15 UTC"
## [53] "2011-12-05 UTC" "2012-02-08 UTC" "2012-06-13 UTC" "2011-11-11 UTC"
## [57] "2012-02-29 UTC" "2012-03-02 UTC" "2012-02-16 UTC" NA
## [61] NA               "2011-12-28 UTC" "2011-12-21 UTC" NA
```

DATAINT

La variabile “DATAINT” non presenta errori da corregere, ma confrontando alcune osservazioni con la colonna della data di dimissione notiamo come per due pazienti questa è prima della data di intervento.

```
errors <- ifelse(difftime(data$DATDIM, data$DATINT, units = "days") >= 0, TRUE, FALSE)
indici <- which(!errors)
data[indici, ]
```

```
## # A tibble: 2 x 20
##   PAZIENTE NASCITA    SEX   STATCIV    PESO ALTEZZA CADUTE CCSCORE SOFAING  MMSE
##      <dbl> <date>     <chr> <chr>     <dbl>   <dbl>  <dbl>   <dbl>   <dbl> <dbl>
## 1       29 1922-01-13 donna "vedovo\~    65     150     NA       2       0  23.8
## 2      145 1923-07-05 donna "vedovo\~    39     150      2       4       4  NA
## # ... with 10 more variables: ALB <dbl>, CALC <dbl>, VITD <dbl>, HBING <dbl>,
## #   TEMPRIC <dbl>, DATDIM <dttm>, DATINT <dttm>, INTDURAT <dbl>, ANEST <dbl>,
## #   `DATA DECESSO` <chr>
```

```
nrow(data)
```

```
## [1] 64
```

Possiamo eliminare le righe dei due pazienti o eliminare la data di dimissione, procediamo sostituendo con “NA” la data di dimissione.

```
#Cancellare le righe
#data <- subset(data, !(PAZIENTE %in% c(29, 145)))
# Seleziona le righe in cui data$PAZIENTE è uguale a 29 o 145
indice <- which(data$PAZIENTE == 29 | data$PAZIENTE == 145)

# Trasforma i valori della colonna data$DATDIM in NA per le righe selezionate
data$DATDIM[indice] <- NA
data$DATDIM
```

```
##  [1] "2011-11-02 UTC" NA               "2011-12-28 UTC" NA
##  [5] "2011-12-01 UTC" "2011-12-09 UTC" "2011-12-16 UTC" "2011-11-04 UTC"
##  [9] "2012-02-29 UTC" NA               "2011-10-24 UTC" "2012-01-10 UTC"
## [13] NA               "2011-11-29 UTC" "2012-06-22 UTC" NA
## [17] "2012-01-05 UTC" "2011-11-10 UTC" "2012-03-09 UTC" NA
## [21] "2011-11-16 UTC" "2012-02-29 UTC" NA               "2012-03-09 UTC"
## [25] NA               "2012-02-24 UTC" "2011-12-12 UTC" "2011-10-14 UTC"
## [29] "2011-12-27 UTC" "2011-11-28 UTC" "2011-11-16 UTC" "2011-12-19 UTC"
## [33] NA               "2012-03-08 UTC" "2011-12-16 UTC" "2012-02-23 UTC"
## [37] NA               "2012-01-03 UTC" "2012-02-23 UTC" "2012-01-10 UTC"
## [41] "2011-11-14 UTC" "2012-06-19 UTC" "2011-11-18 UTC" "2011-10-14 UTC"
## [45] "2011-11-07 UTC" "2012-06-13 UTC" "2011-10-28 UTC" "2012-02-17 UTC"
## [49] "2011-10-28 UTC" "2012-01-12 UTC" "2012-06-19 UTC" "2012-02-15 UTC"
## [53] "2011-12-05 UTC" "2012-02-08 UTC" "2012-06-13 UTC" NA
## [57] "2012-02-29 UTC" "2012-03-02 UTC" "2012-02-16 UTC" NA
## [61] NA               "2011-12-28 UTC" "2011-12-21 UTC" NA
```

```
nrow(data)
```

```
## [1] 64
```

INTDURAT

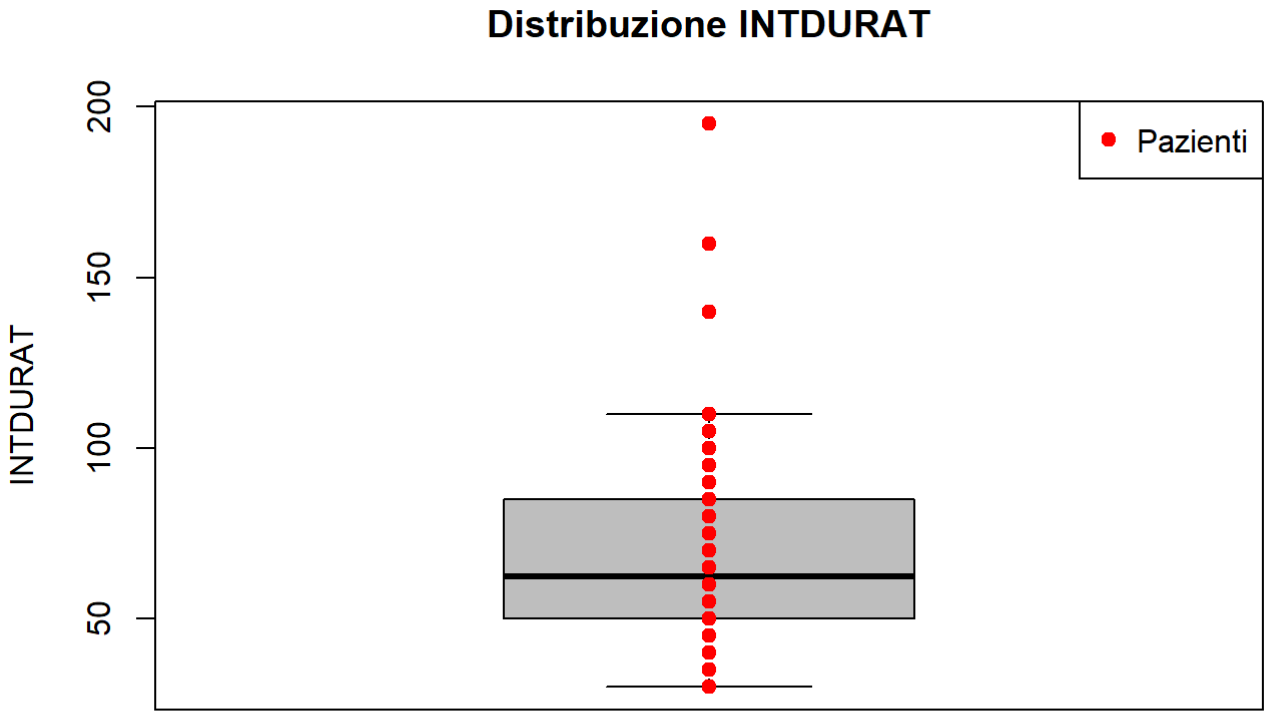
La variabile “INTDURAT” non sembra presentare errori anomali o valori mancanti.

```
data$INTDURAT
```

```
##  [1]  40  95  35 100  80  55 140 100  50  75  45  60  55  80  70  85 110 110  50
## [20]  90  55  70  40 100  75  60  40  45  60  55  60  35  85  30  40  90  70  95
## [39]  60  90  55  NA  70  70 105 160 195  70  60  30  NA  80  35  45  50  60  65
## [58]  80  55  75  50  35  90  40
```

Osserviamo la distribuzione.

```
boxplot(data$INTDURAT,
        col = "grey",
        border = "black",
        names = c("INTDURAT"),
        ylab = "INTDURAT",
        pch = 19,
        main= "Distribuzione INTDURAT")
points(x = rep(1, length(data$INTDURAT)), y = data$INTDURAT, pch = 19, col = "red")
legend("topright", legend = "Pazienti", pch = 19, col = "red")
```



ANEST

Cambiamo i valori della variabile “ANEST” utilizzando la documentazione.

```
nuovi_valori <- c("generale", "spinale", "peridurale", "plessica", "combinata", "sedaz
ione", "locale_assistita", "altro")

data$ANEST <- ifelse(data$ANEST == 1, nuovi_valori[1],
                     ifelse(data$ANEST == 2, nuovi_valori[2],
                            ifelse(data$ANEST == 3, nuovi_valori[3],
                                   ifelse(data$ANEST == 4, nuovi_valori[4],
                                          ifelse(data$ANEST == 5, nuovi_valori[5],
                                                 ifelse(data$ANEST == 6, nuovi_valori
[6],
                                                        ifelse(data$ANEST == 7, nuovi_
valori[7],
                                                               ifelse(data$ANEST == 8,
nuovi_valori[8],
                                                                      data$ANES
T))))))))
data$ANEST
```

```
##  [1] "spinale"   "spinale"   "plessica"  "combinata" "generale"  "generale"
##  [7] "generale"  "plessica"  "plessica"  "generale"  "plessica"  "combinata"
## [13] "generale"  "generale"  "spinale"   "plessica"  "generale"  "spinale"
## [19] "generale"  "spinale"   "generale"  "combinata" "plessica"  "plessica"
## [25] "generale"  "spinale"   "spinale"   "spinale"   "generale"  "spinale"
## [31] "generale"  "generale"  "spinale"   "spinale"   "generale"  "combinata"
## [37] "spinale"   "generale"  "spinale"   "combinata" "spinale"   "combinata"
## [43] "generale"  "spinale"   "spinale"   "combinata" "generale"  "combinata"
## [49] "spinale"   "generale"  "plessica"  "spinale"   "generale"  "generale"
## [55] "generale"  "generale"  "plessica"  "spinale"   "generale"  "plessica"
## [61] "spinale"   "generale"  "generale"  "spinale"
```

DATA DECESSO

La variabile “DATA DECESSO” presenta molteplici errori: innanzitutto ci sono valori uguali a “-1” che equivalgono a “Si è rifiutata”; dopo di che le date sono rappresentate con valori di tipo non numerico, procediamo con la correzione.

```
data$'DATA DECESSO'
```

```
##  [1] NA              "40927"         "41010"         "40947"
##  [5] NA              NA              NA              "41273"
##  [9] NA              NA              NA              "40949"
## [13] "-1"            NA              "-1"            "40563"
## [17] NA              NA              NA              NA
## [21] NA              NA              NA              NA
## [25] NA              NA              NA              "SI è RIFIUTATA"
## [29] "40904"         NA              NA              NA
## [33] NA              NA              NA              NA
## [37] NA              NA              NA              "40965"
## [41] NA              NA              "40194"         "40830"
## [45] NA              "-1"            NA              NA
## [49] NA              "40921"         "-1"            "41034"
## [53] NA              NA              NA              NA
## [57] NA              NA              NA              "40926"
## [61] NA              NA              NA              "40945"
```

La variabile “DATA DECESSO” presenta molteplici errori: innanzitutto ci sono valori uguali a “-1” che equivalgono a “Si è rifiutata”; dopo di che le date sono rappresentate con valori di tipo non numerico, procediamo con la correzione di tutti i valori.

```
data$'DATA DECESSO' = ifelse(data$`DATA DECESSO`=='SI è RIFIUTATA',NA,data$`DATA DECES
SO`)
data$'DATA DECESSO' = ifelse(data$`DATA DECESSO`== -1,NA,data$`DATA DECESSO`)
data$'DATA DECESSO' = as.numeric(data$`DATA DECESSO`)
(data$'DATA DECESSO' = as.Date(x = data$`DATA DECESSO`,origin = "1899-12-30", ))
```

```
##  [1] NA           "2012-01-19" "2012-04-11" "2012-02-08" NA
##  [6] NA           NA           "2012-12-30" NA           NA
## [11] NA           "2012-02-10" NA           NA           NA
## [16] "2011-01-20" NA           NA           NA           NA
## [21] NA           NA           NA           NA           NA
## [26] NA           NA           NA           "2011-12-27" NA
## [31] NA           NA           NA           NA           NA
## [36] NA           NA           NA           NA           "2012-02-26"
## [41] NA           NA           "2010-01-16" "2011-10-14" NA
## [46] NA           NA           NA           NA           "2012-01-13"
## [51] NA           "2012-05-05" NA           NA           NA
## [56] NA           NA           NA           NA           "2012-01-18"
## [61] NA           NA           NA           "2012-02-06"
```

```
data$'DATA DECESSO'
```

```
##  [1] NA           "2012-01-19" "2012-04-11" "2012-02-08" NA
##  [6] NA           NA           "2012-12-30" NA           NA
## [11] NA           "2012-02-10" NA           NA           NA
## [16] "2011-01-20" NA           NA           NA           NA
## [21] NA           NA           NA           NA           NA
## [26] NA           NA           NA           "2011-12-27" NA
## [31] NA           NA           NA           NA           NA
## [36] NA           NA           NA           NA           "2012-02-26"
## [41] NA           NA           "2010-01-16" "2011-10-14" NA
## [46] NA           NA           NA           NA           "2012-01-13"
## [51] NA           "2012-05-05" NA           NA           NA
## [56] NA           NA           NA           NA           "2012-01-18"
## [61] NA           NA           NA           "2012-02-06"
```

Abbiamo concluso la fase di cleaning, ogni variabile è stata ripulita, confermiamo la presenza di molti outlier, abbiamo deciso di non eliminarli per non ridurre ulteriormente la dimensione del dataset.

Pre Processing

In questa fase andiamo a creare delle nuove variabili per il nostro dataset che ci serviranno successivamente. Andiamo a definire nuove variabili dividendo i punteggi in classi.

CCSCORE_classe

Dividiamo il punteggio "CCSCORE" in tre classi: basso, medio, alto.

```
data$CCSCORE<-as.numeric(data$CCSCORE)
data$CCSCORE_classe = ifelse((data$CCSCORE==0 |data$CCSCORE==1 |data$CCSCORE==2),'cc_b
asso',
ifelse((data$CCSCORE==3|data$CCSCORE==4),'cc_medio','cc_alto'))
(table(data$CCSCORE_classe))
```

```
##
##  cc_alto cc_basso cc_medio
##       12       29       23
```

SOFAING_classe

Dividiamo il punteggio "SOFAING" in due classi: basso e alto.

```
data$SOFAING_classe = ifelse(data$SOFAING== '0','sofa_basso', 'sofa_alto')
table(data$SOFAING_classe)
```

```
##
##  sofa_alto sofa_basso
##         20         44
```

MMSE_classe

Dividamo il punteggio "MMSE" in quattro classi: grave, moderato, lieve, normale.

```
data$MMSE_classe <- data$MMSE

# Raggruppare i punteggi in classi
data$MMSE_classe <- cut(data$MMSE, breaks = c(-1, 9, 18, 23, 30),
                                    labels = c("grave", "moderato", "lieve", "normal
e"))

#Conteggio delle osservazioni in ogni classe
table(data$MMSE_classe)
```

```
##
##    grave moderato    lieve  normale
##       14       13       10       23
```

BMI e BMI_classe

Andiamo a creare la variabile "BMI" e le sue classi. Il BMI (Body Mass Index) è un indice che indica la relazione tra il peso e l'altezza di una persona.

```
data$PESO<-as.numeric(data$PESO)
data$ALTEZZA<-as.numeric(data$ALTEZZA)
data$BMI<-(data$PESO)/((data$ALTEZZA/100)^2)
data$BMI_classi = ifelse(data$BMI<18.5,'sottopeso',ifelse(data$BMI<25,'normopeso','sov
rappeso'))
head(data$BMI)
```

```
## [1] 25.95156 21.94787 24.22145 18.02596 23.18339 22.22222
```

```
table(data$BMI_classi)
```

```
##
##  normopeso  sottopeso sovrappeso
##         33          8         19
```

AGE e AGE_classe

Calcoliamo l'età dei pazienti, assumendo la data di intervento come riferimento per il calcolo, anche questa variabile la dividiamo in classi.

```
# calculate the age in years
data$AGE <- as.numeric(difftime(data$DATINT, data$NASCITA, units = "weeks")) / 52.25
data$AGE <- trunc(data$AGE)
cat("data$AGE =", data$AGE, "\n")
```

```
## data$AGE = 71 83 83 89 87 92 92 94 76 87 73 102 93 83 88 80 79 77 83 81 86 84 89 97
91 90 83 81 89 81 80 95 97 82 74 80 91 79 83 91 83 87 83 81 99 85 84 75 87 83 79 85 86
86 89 88 85 70 79 90 89 89 85 89
```

Il range delle classi viene fissato a 5 anni.

```
breakpoints <- c(69, 75, 80, 85, 90, 95, 105)
data$AGE_classi <- cut(data$AGE, breakpoints, labels = c("70-75", "75-80", "80-85", "8
5-90", "90-95", "95-100+"))
table(data$AGE_classi)
```

```
##
##   70-75   75-80   80-85   85-90   90-95 95-100+
##       5       9      20      18       8       4
```

```
data$AGE_classi
```

```
##  [1] 70-75   80-85   80-85   85-90   85-90   90-95   90-95   90-95   75-80
## [10] 85-90   70-75   95-100+ 90-95   80-85   85-90   75-80   75-80   75-80
## [19] 80-85   80-85   85-90   80-85   85-90   95-100+ 90-95   85-90   80-85
## [28] 80-85   85-90   80-85   75-80   90-95   95-100+ 80-85   70-75   75-80
## [37] 90-95   75-80   80-85   90-95   80-85   85-90   80-85   80-85   95-100+
## [46] 80-85   80-85   70-75   85-90   80-85   75-80   80-85   85-90   85-90
## [55] 85-90   85-90   80-85   70-75   75-80   85-90   85-90   85-90   80-85
## [64] 85-90
## Levels: 70-75 75-80 80-85 85-90 90-95 95-100+
```

GGOSSERV e ANNIOSSERV

Infine andiamo a calcolare i giorni che il paziente rimane in osservazione dalla durata dell'intervento ad una data decisa a priori, assumendo che i paienti rimangano sotto osservazione della clinica; questa variabile ci servirà per la costruzione della curva di sopravvivenza.

```
data$fine_data = ifelse(is.na(data$`DATA DECESSO`),NA,data$`DATA DECESSO`)
data$fine_data =as.numeric(data$fine_data)
data$fine_data =as.Date(data$fine_data,origin = "1970-01-01", )
data$fine_data = ifelse(is.na(data$fine_data),as.Date("2021/12/31"),data$fine_data)
data$fine_data =as.Date(data$fine_data,origin = "1970-01-01", )
```

```
data$GGOSSERV = as.numeric(difftime(data$fine_data, data$DATINT, units = "days"))
data$GGOSSERV
```

```
##  [1] 3726    6  121   27 3693 3682 3696  443 3602 3637 3733   43 2394 3699 3486
## [16] -357 3655 3714 3594 3630 3705 3599 3623 3595 3644 3607 3679 3740    0 3696
## [31] 3707 3675 3601 3591 3679 3610 3637 3662 3607   61 3710 3487 -664   11 3719
## [46] 3500 3733 3612 3721   22 3487   93 3691 3621 3493 3690 3601 3599 3615    2
## [61] 3623 3668 3670   14
```

Osserviamo due valori negativi, quindi questo significa che ci sono due date incongruenti tra data decesso ed intervento, considerando che la data di decesso è un dato molto importante e significativo, decidiamo di eliminare tutti i dati riguardanti questi pazienti per non influire sulle nostre analisi.

```
data <- subset(data, GGOSSERV>=0)
```

Creiamo la variabile “ANNIOSSERV”.

```
data$ANNIOSSERV<- data$GGOSSERV/365
data$ANNIOSSERV <- trunc(data$ANNIOSSERV)
cat("data$ANNIOSSERV =", data$ANNIOSSERV, "\n")
```

```
## data$ANNIOSSERV = 10 0 0 0 10 10 10 1 9 9 10 0 6 10 9 10 10 9 9 10 9 9 9 9 9 10 10
0 10 10 10 9 9 10 9 9 10 9 0 10 9 0 10 9 10 9 10 0 9 0 10 9 9 10 9 9 9 0 9 10 10 0
```

Exploratory data analysis

L’EDA è una metodologia analitica che prevede l’esplorazione dei dati in modo visivo e statistico, al fine di comprendere le caratteristiche del dataset e le relazioni tra le variabili; può essere applicata a questo dataset per identificare eventuali relazioni tra le variabili, ad esempio per verificare se esiste una correlazione tra il punteggio SOFA e il CCScore, o se esiste una relazione tra le caratteristiche dei pazienti e i loro punteggi di gravità.

Missing values

Analizziamo la presenza di valori mancanti.

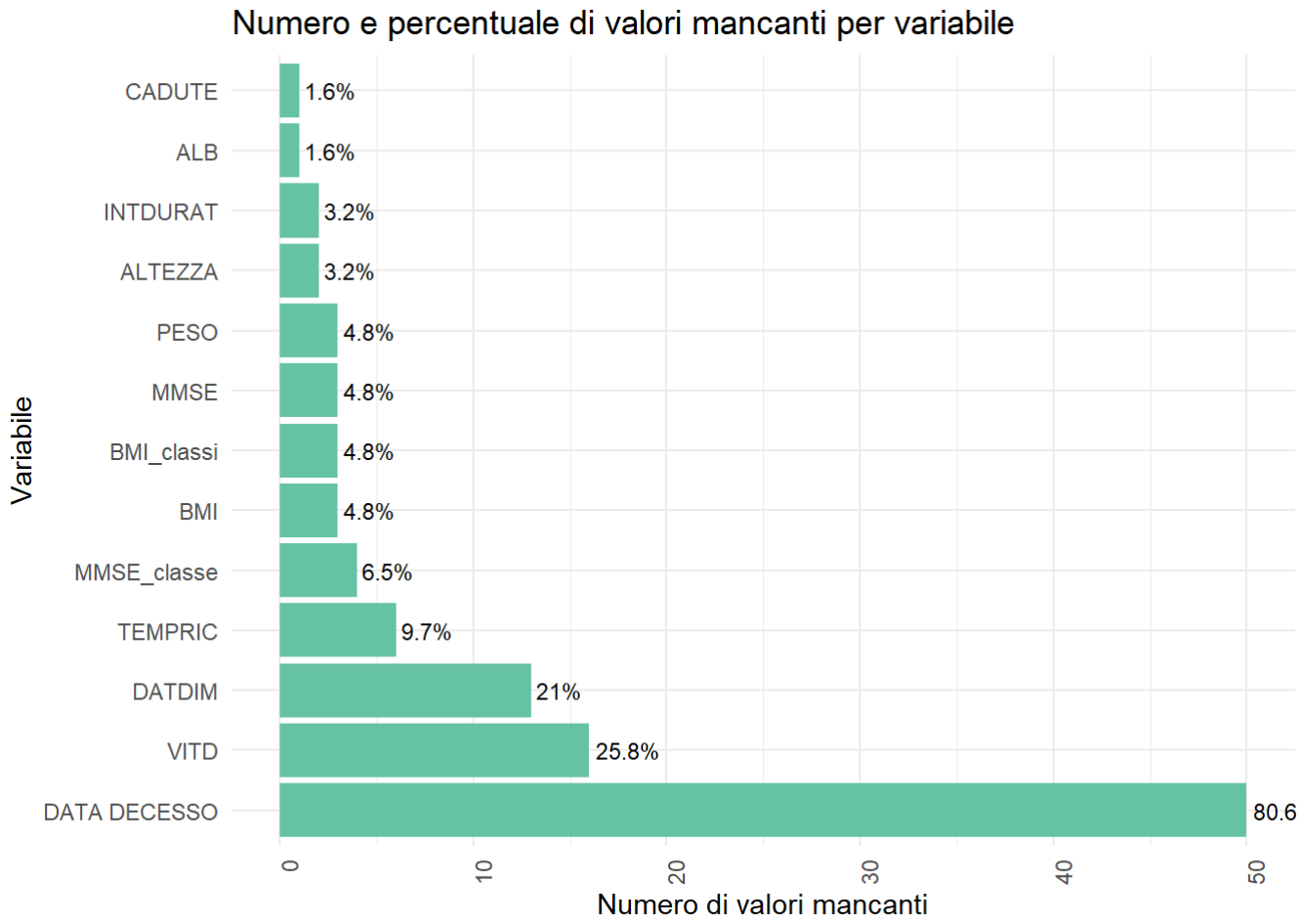
```
library(ggplot2)
library(dplyr)
```

```
##
## Caricamento pacchetto: 'dplyr'
```

```
## I seguenti oggetti sono mascherati da 'package:stats':
##
##     filter, lag
```

```
## I seguenti oggetti sono mascherati da 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# Calcola il numero di valori nulli per ogni colonna
null_counts <- sapply(data, function(x) sum(is.na(x)))

# Calcola le percentuali di valori nulli per ogni colonna
null_percents <- null_counts / nrow(data) * 100

# Crea un dataframe con i valori delle colonne
null_data <- data.frame(col = names(data), count = null_counts, percent = null_percent
s)

# Filtra le colonne con valore nulli
null_data <- null_data %>% filter(count > 0)

# Ordina il dataframe in base al numero di valori mancanti
null_data <- null_data[order(null_data$count, decreasing = TRUE),]

# Crea il grafico a barre orizzontale
ggplot(null_data, aes(x = reorder(col, -count), y = count, fill = "lightpink")) +
  geom_bar(stat = "identity") +
  ggtitle("Numero e percentuale di valori mancanti per variabile") +
  xlab("Variabile") +
  ylab("Numero di valori mancanti") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        legend.position = "none") +
  scale_fill_brewer(palette = "Set2") +
  geom_text(aes(label = paste0(round(percent, 1), "%")),
            hjust = -0.1, size = 3, color = "black") +
  coord_flip() +
  scale_y_continuous(limits = c(0, max(null_data$count)))
```



Nel caso della variabile "data decesso" che contiene valori nulli all'80%, questo può indicare che la maggior parte dei pazienti è in vita oppure che le informazioni sul decesso non sono state registrate o non sono disponibili. In questo caso, sarà necessario valutare attentamente come gestire questi dati mancanti.

La variabile "VITD" con valori nulli al 25% potrebbe comunque essere utilizzata nell'analisi, a patto che il numero di valori mancanti non influisca in modo significativo sulla potenza dell'analisi o sulla rappresentatività del campione, anche la variabile "DATDIM" che indica la data di dimissione contiente un numero significativo di valori nulli.

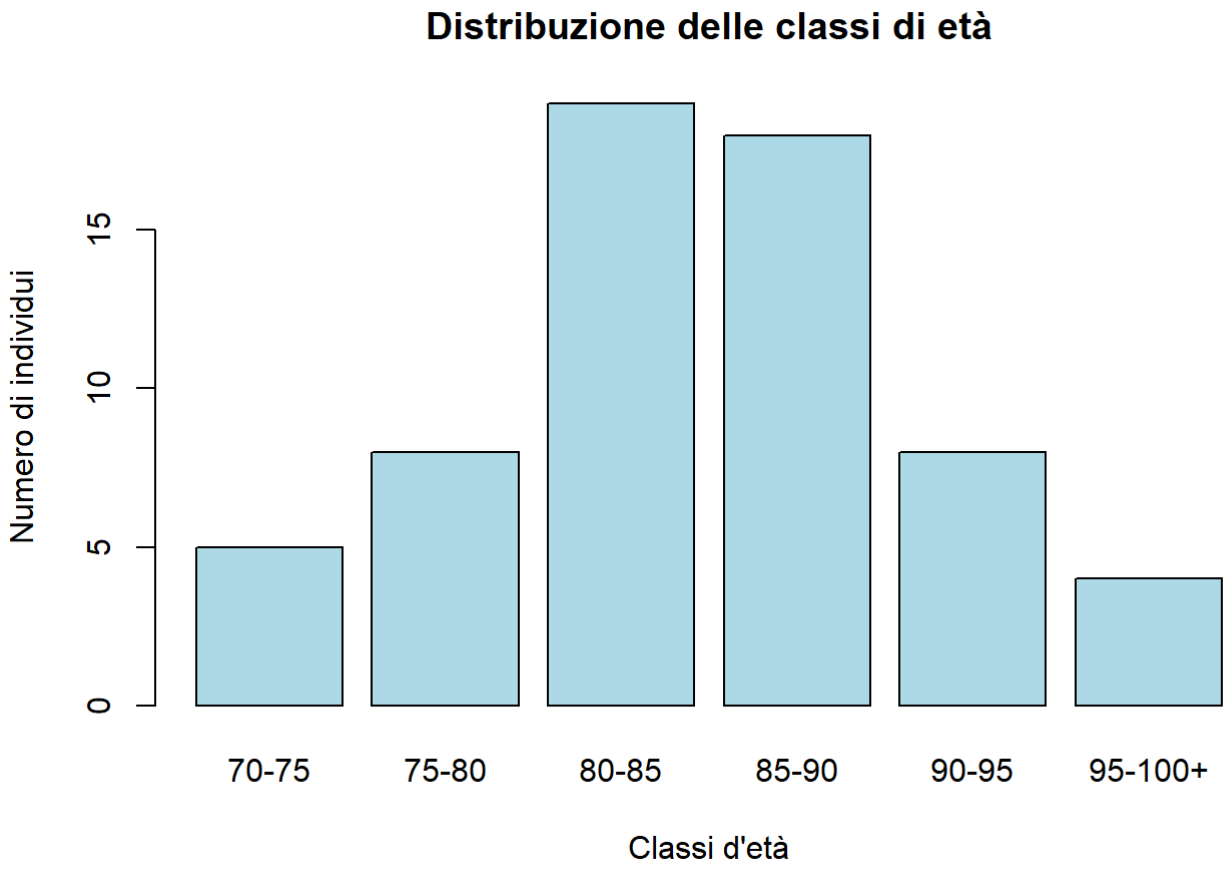
Infine, la presenza di pochi valori nulli (sotto il 5%) può essere considerata un valore accettabile e non influente sull'analisi, tuttavia, anche in questi casi, è importante verificare la ragione per cui questi valori sono mancanti.

Analisi delle principali variabili rispetto alle classi di età

Un'analisi delle principali variabili rispetto alle classi di età può aiutare a comprendere le differenze tra le varie fasce d'età in termini di variabili di interesse. Ciò può fornire informazioni preziose per la gestione delle malattie e per lo sviluppo di interventi di prevenzione e trattamento specifici per le diverse fasce d'età.

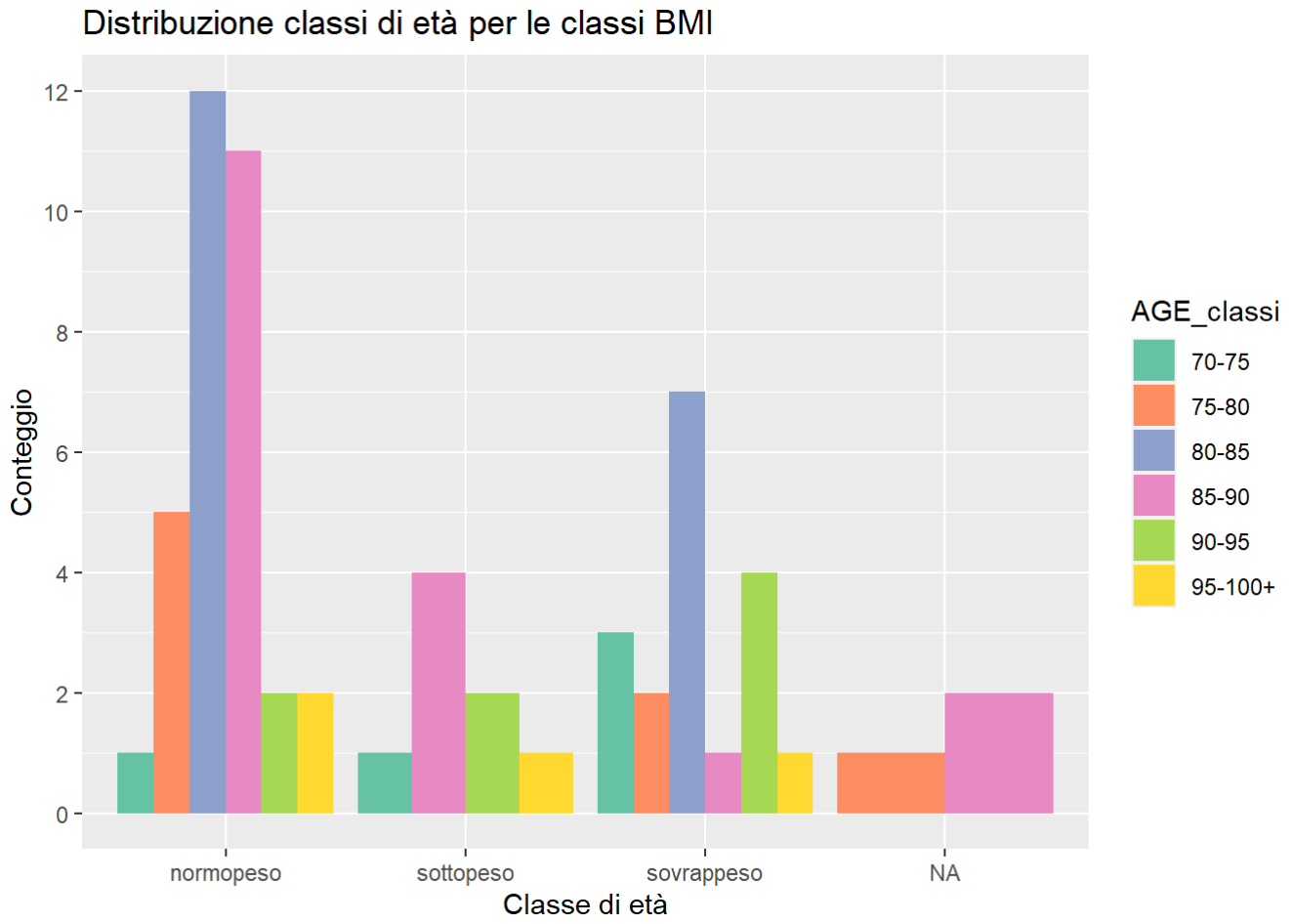
Andremo ad esaminare come variano "BMI", "SOFA", "CC-score", "MMSE-score" in base all'età dei pazienti.

```
# Calcola il numero di donne e uomini
age_count <- table(data$AGE_classi)

# Crea il barplot
barplot(age_count,
        main = "Distribuzione delle classi di età",
        xlab = "Classi d'età",
        ylab = "Numero di individui",
        col = c( "lightblue"))
```



La maggior parte dei pazienti è compresa nell'intervallo di età da 80 a 90 anni.

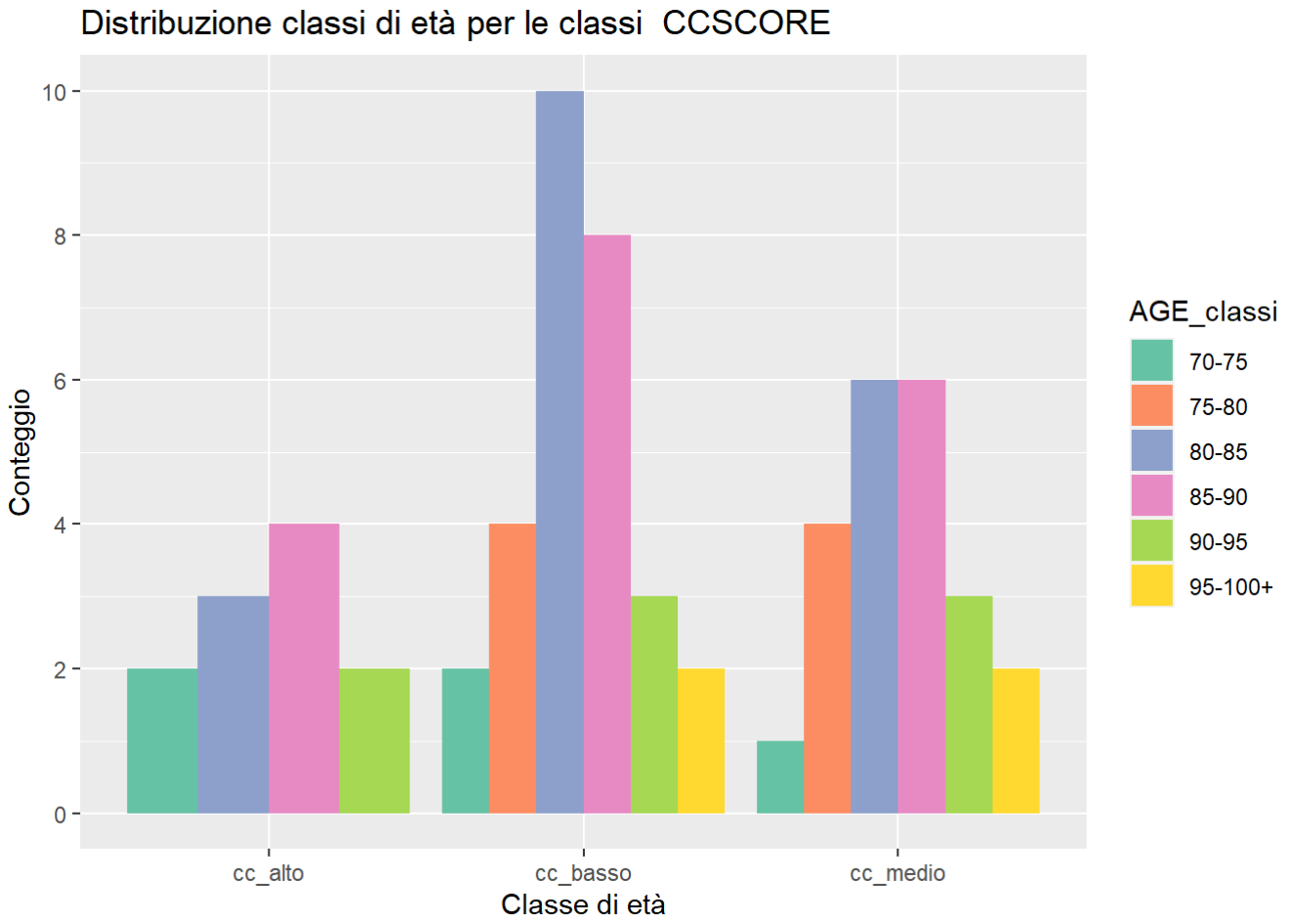
```
ggplot(data, aes(x = BMI_classi, fill = AGE_classi)) +
  geom_bar(position = "dodge") +
  ggtitle("Distribuzione classi di età per le classi BMI") +
  xlab("Classe di età") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(breaks = seq(0, 15, 2))
```



La maggior parte dei pazienti è normopeso, soprattutto nelle fasce d'età da 75 a 90 anni, questo suggerisce che il peso corporeo dei pazienti in queste fasce d'età sia generalmente adeguato rispetto all'età e alla statura, tuttavia, è importante notare che il peso corporeo può essere influenzato da fattori come l'attività fisica, la dieta, le condizioni di salute e i farmaci.

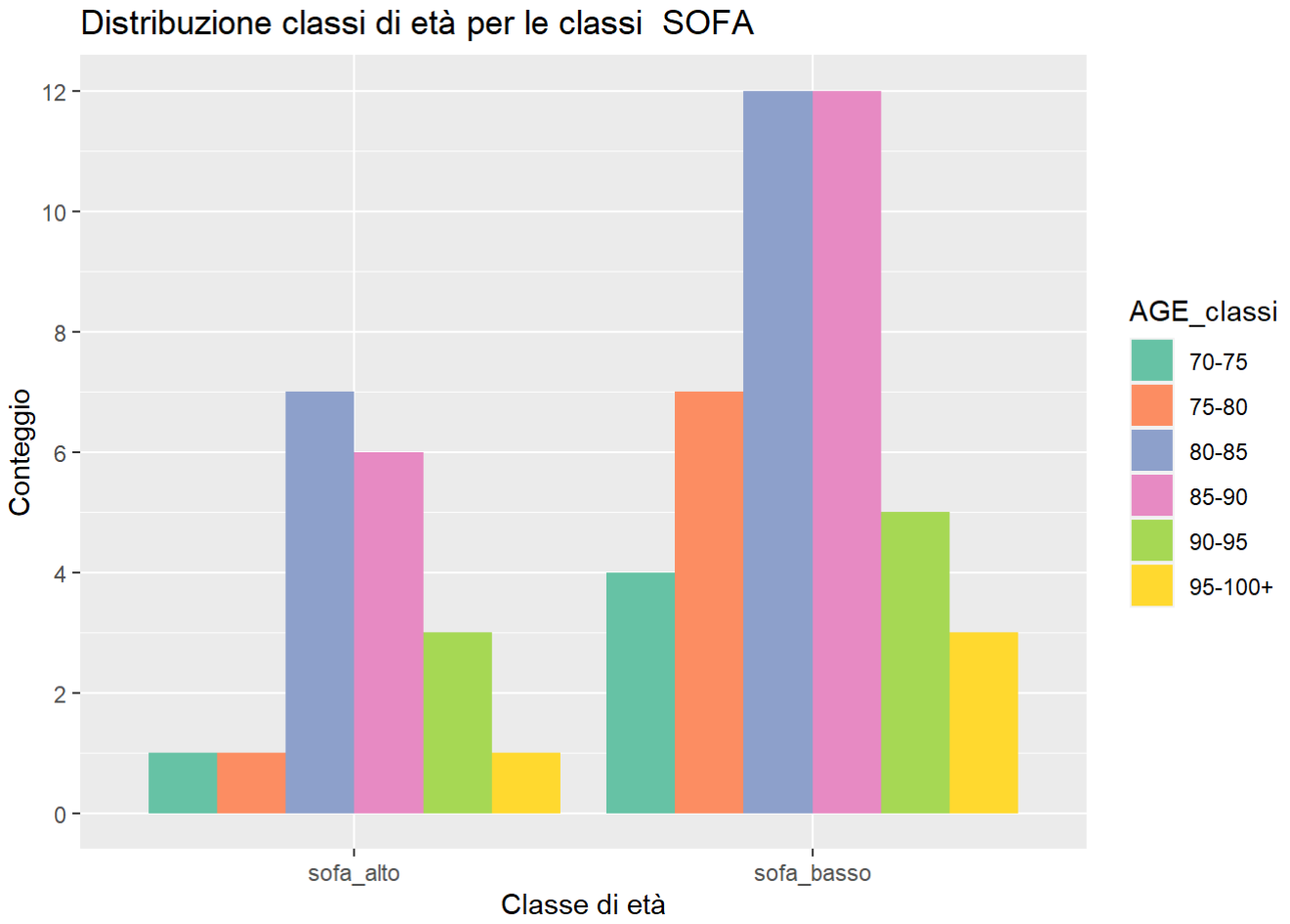
Inoltre, il fatto che circa 15 pazienti su 62 siano in sovrappeso, di cui 7 pazienti sono nella fascia d'età 80-85, suggerisce che l'aumento di peso può essere un problema significativo per alcune persone anziane.

```
ggplot(data, aes(x = CCSCORE_classe, fill = AGE_classi)) +
  geom_bar(position = "dodge") +
  ggtitle("Distribuzione classi di età per le classi  CCSCORE") +
  xlab("Classe di età") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(breaks = seq(0, 15, 2))
```



Circa la metà dei pazienti ha un “CC-score” basso, questo suggerisce che molti pazienti non presentano molte comorbidità. Al contrario, il fatto che circa 20 pazienti abbiano un “CC-score” medio suggerisce che un certo numero di pazienti potrebbe avere comorbilità moderate. Inoltre, l’assenza di pazienti con “CC-score” alto nella fascia d’età 75-80 suggerisce che potrebbero avere una buona salute generale.

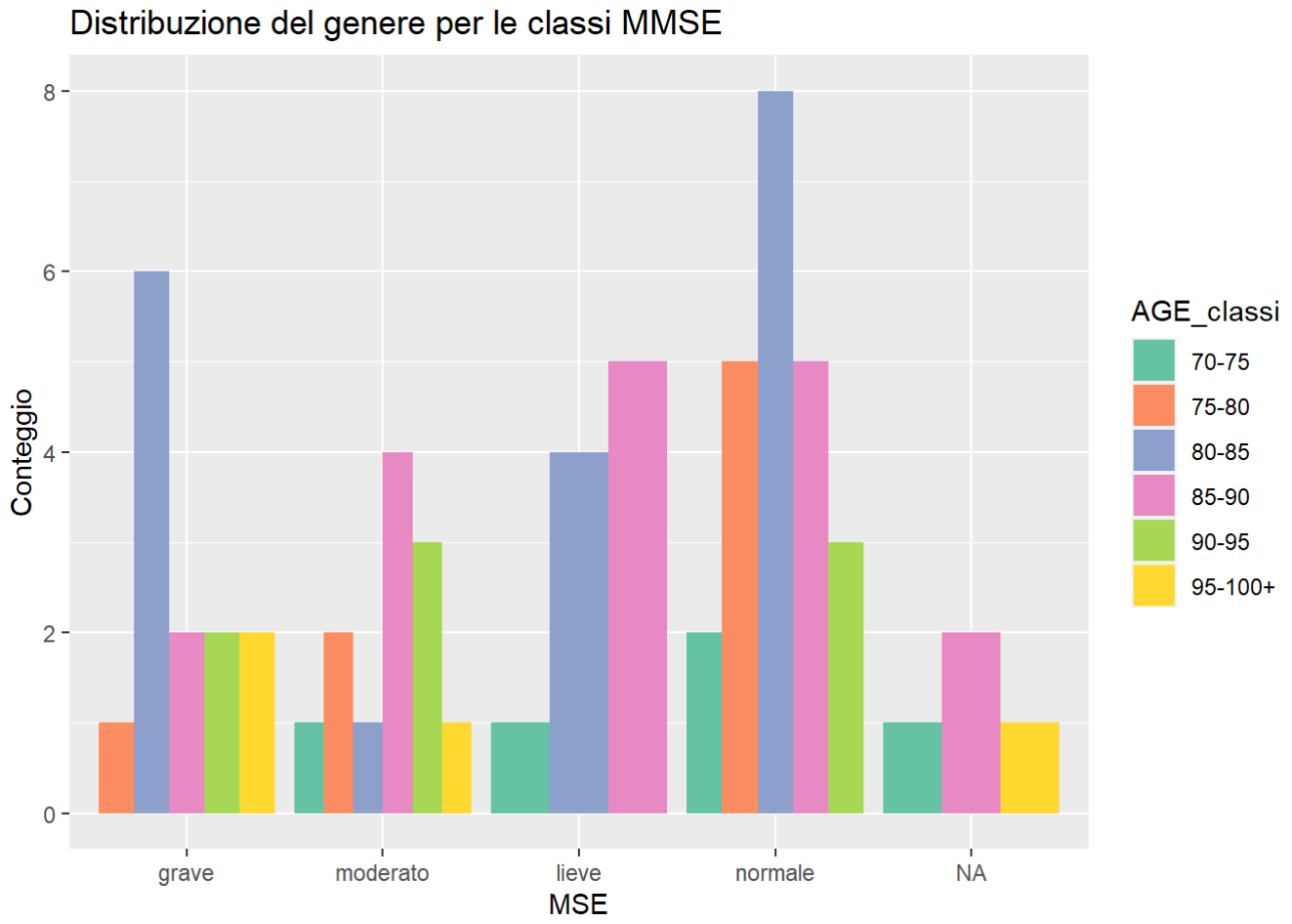
```
ggplot(data, aes(x = SOFAING_classe, fill = AGE_classi)) +
  geom_bar(position = "dodge") +
  ggtitle("Distribuzione classi di età per le classi  SOFA") +
  xlab("Classe di età") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(breaks = seq(0, 15, 2))
```



La maggior parte dei pazienti ha un "SOFA" basso, questo suggerisce che molti pazienti potrebbero non avere comorbilità significative, quindi potrebbe essere un fattore positivo per la loro salute e il loro benessere generale.

Inoltre, il fatto che ci siano solo due pazienti con uno score alto nella fascia d'età 70-80 suggerisce che i pazienti in questa fascia d'età potrebbero essere relativamente sani.

```
ggplot(data, aes(x = MMSE_classe, fill = AGE_classi)) +
  geom_bar(position = "dodge") +
  ggtitle("Distribuzione del genere per le classi MMSE") +
  xlab("MSE") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")+
scale_y_continuous(breaks = seq(0, 15, 2))
```

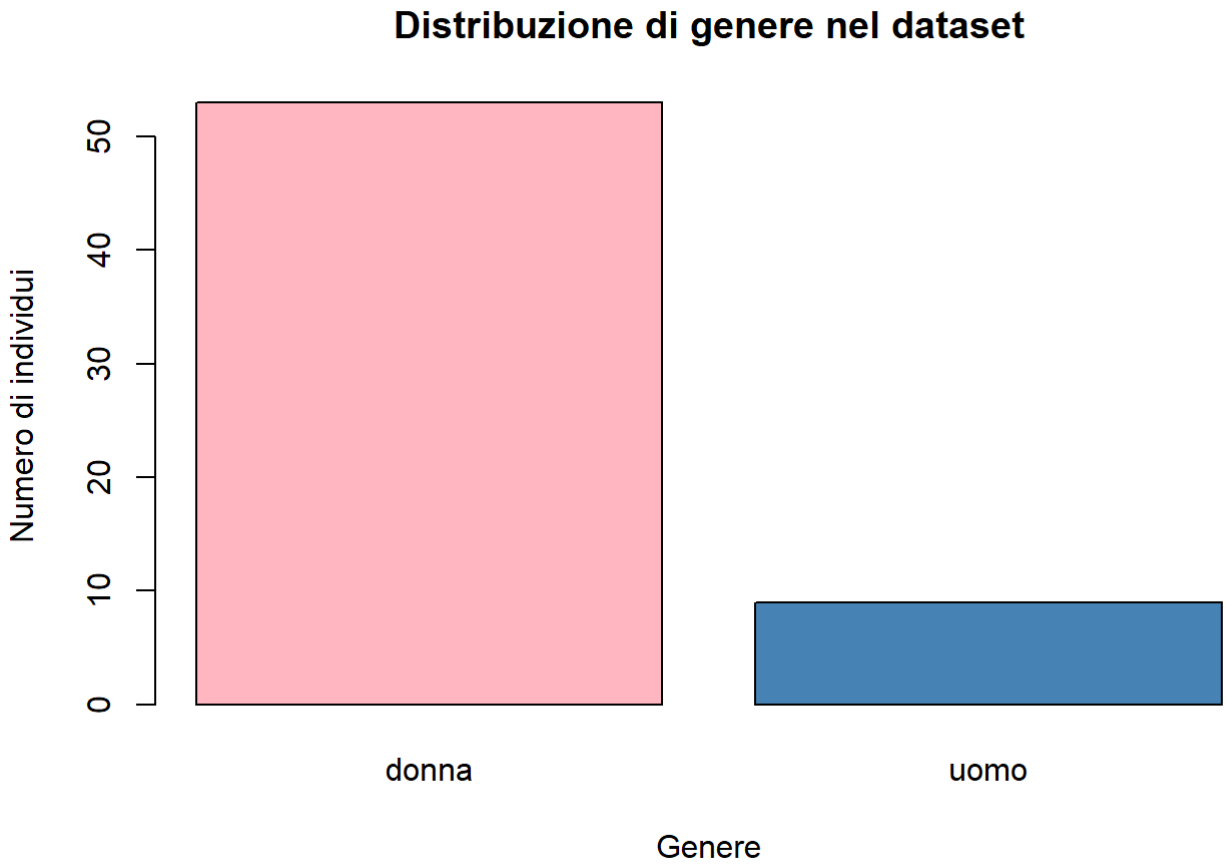


L'omogeneità della distribuzione delle classi di età rispetto ai livelli di "MMSE" suggerisce che l'età potrebbe non essere un fattore determinante nella capacità cognitiva dei pazienti. Tuttavia, il fatto che la maggior parte dei pazienti nella fascia d'età 70-80 abbia valori "MMSE" più normali e lievi rispetto alle altre classi potrebbe indicare una maggiore prevalenza di problemi cognitivi in età più avanzata.

Analisi delle principali variabili rispetto al sesso

Andiamo ad analizzare le distribuzioni di genere in base alle variabili più importanti: Età, BMI, CC score, SOFA e MMSE. Osserviamo il numero di donne e uomini presenti nel dataset.

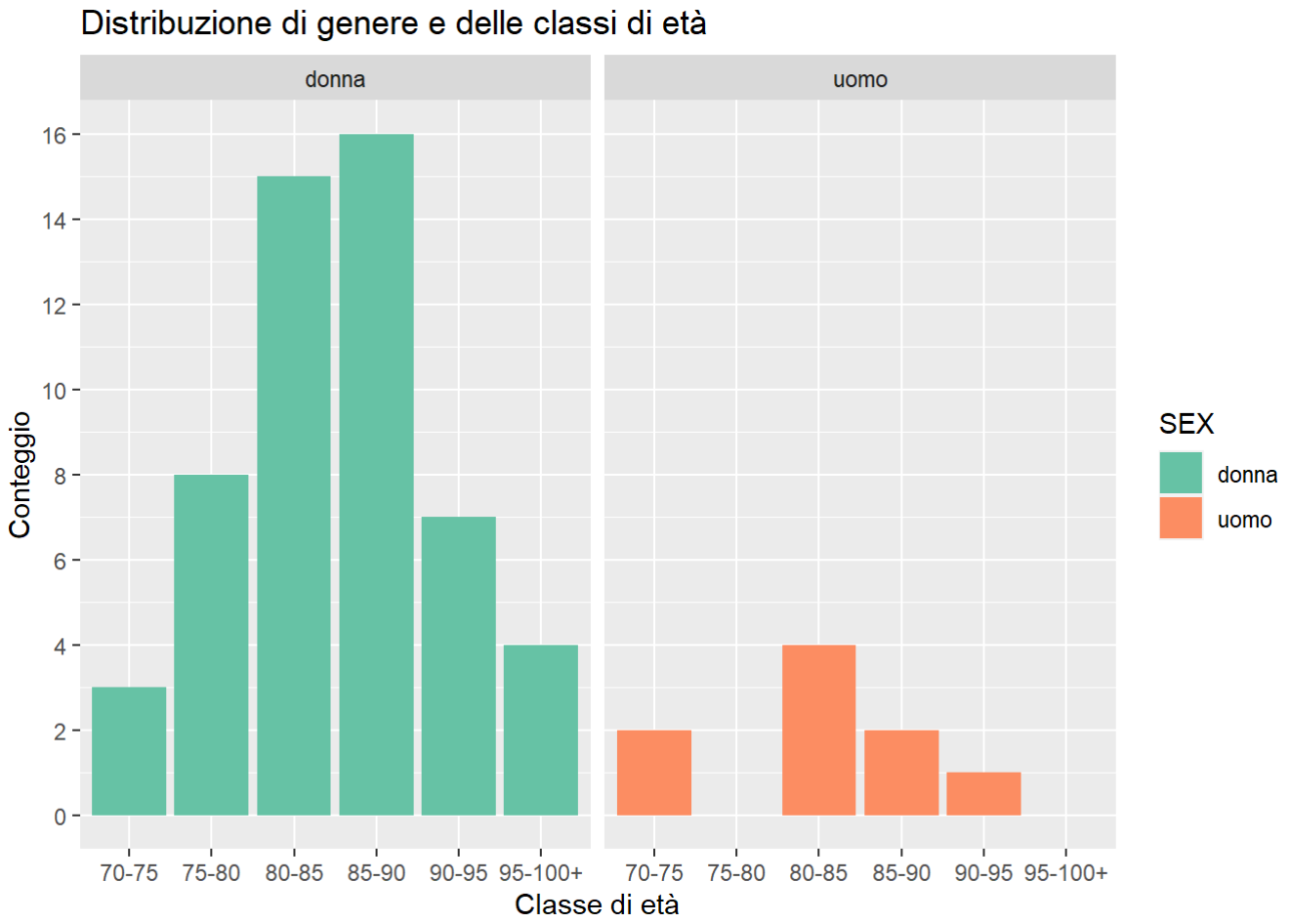
```
# Calcola il numero di donne e uomini
sex_counts <- table(data$SEX)

# Crea il barplot
barplot(sex_counts,
        main = "Distribuzione di genere nel dataset",
        xlab = "Genere",
        ylab = "Numero di individui",
        col = c("lightpink", "steelblue"))
```



Notiamo come la maggior parte dei pazienti sono donne, questa variabile risulta molto sbilanciata quindi successivamente sarà meglio svolgere analisi di genere per non creare distorsioni fra i dati.

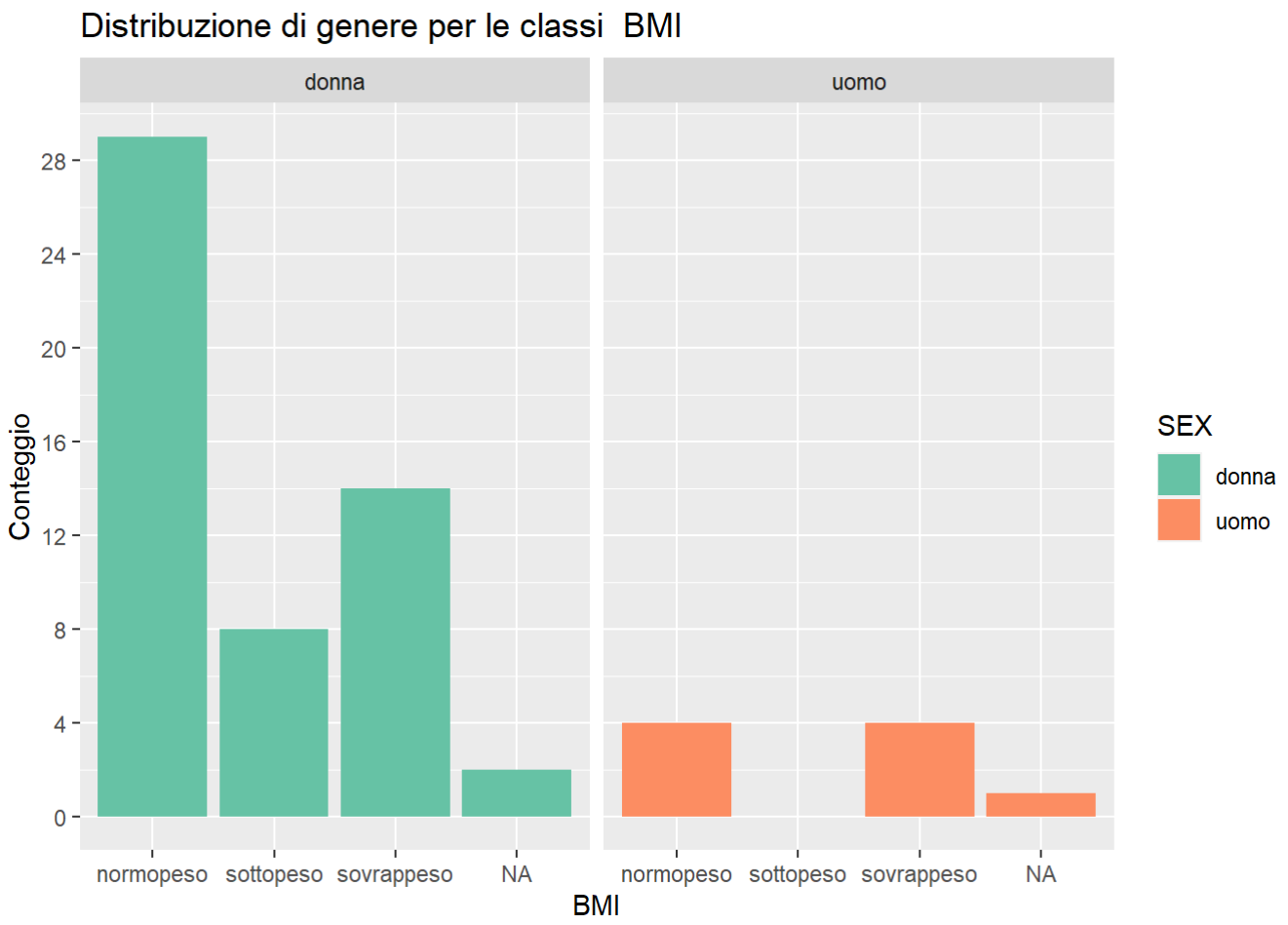
Osserviamo la distribuzioni di genere rispetto alle classi di età.

```
ggplot(data, aes(x = AGE_classi, fill = SEX)) +
  geom_bar(position = "dodge") +
  facet_grid(. ~ SEX) +
  ggtitle("Distribuzione di genere e delle classi di età") +
  xlab("Classe di età") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")+
  scale_y_continuous(breaks = seq(0, 20, 2))
```



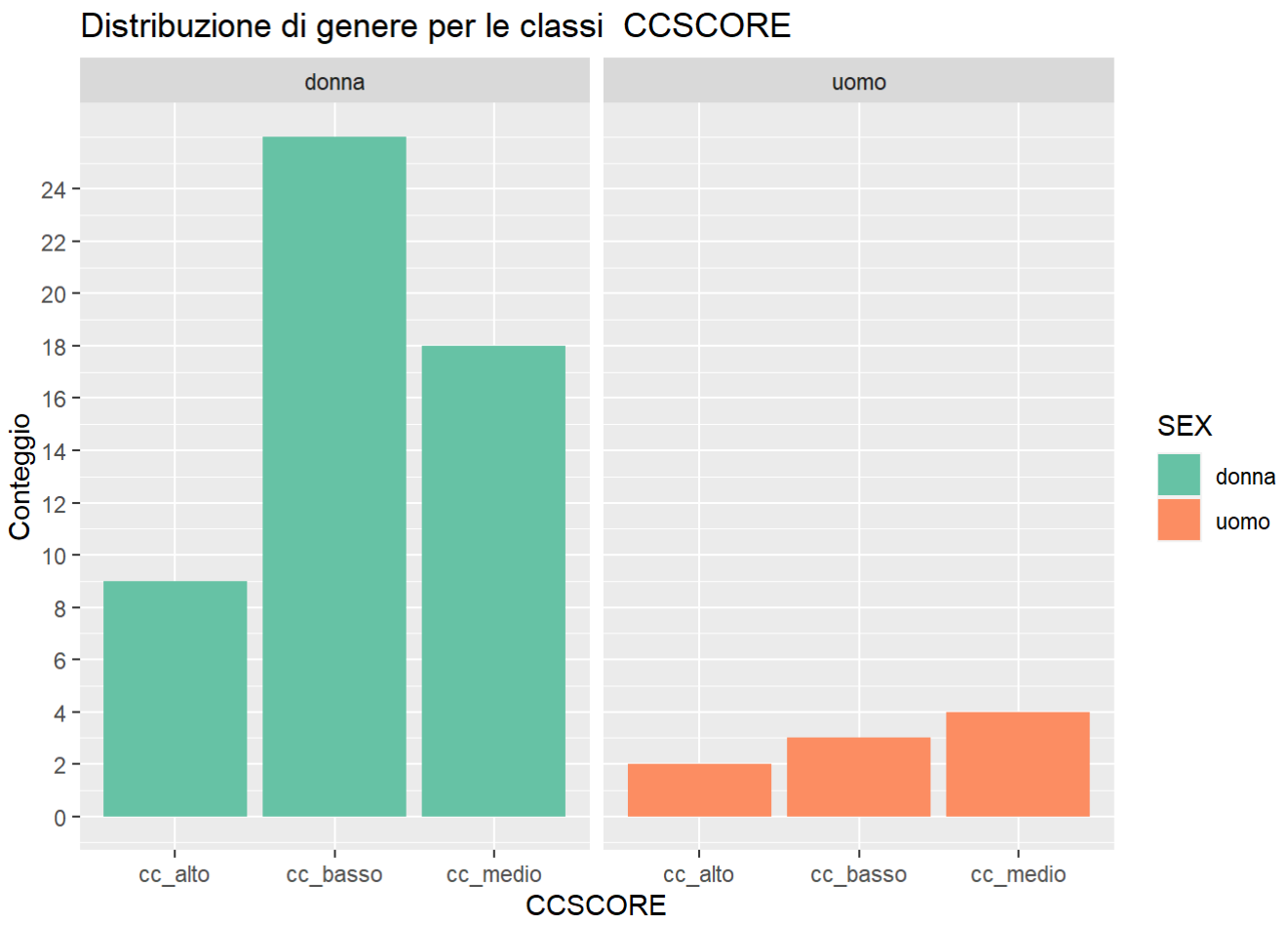
Come osservato precedentemente, i pazienti sono molto anziani e con un età maggiormente compresa fra 80 e 90 anni.

```
ggplot(data, aes(x = BMI_classi, fill = SEX)) +
  geom_bar(position = "dodge") +
  facet_grid(. ~ SEX) +
  ggtitle("Distribuzione di genere per le classi  BMI") +
  xlab("BMI") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")+
scale_y_continuous(breaks = seq(0, 30, 4))
```



Più del 50% dei pazienti ha un peso corporeo adeguato alla propria età, inoltre notiamo 15 pazienti donna in sovrappeso.

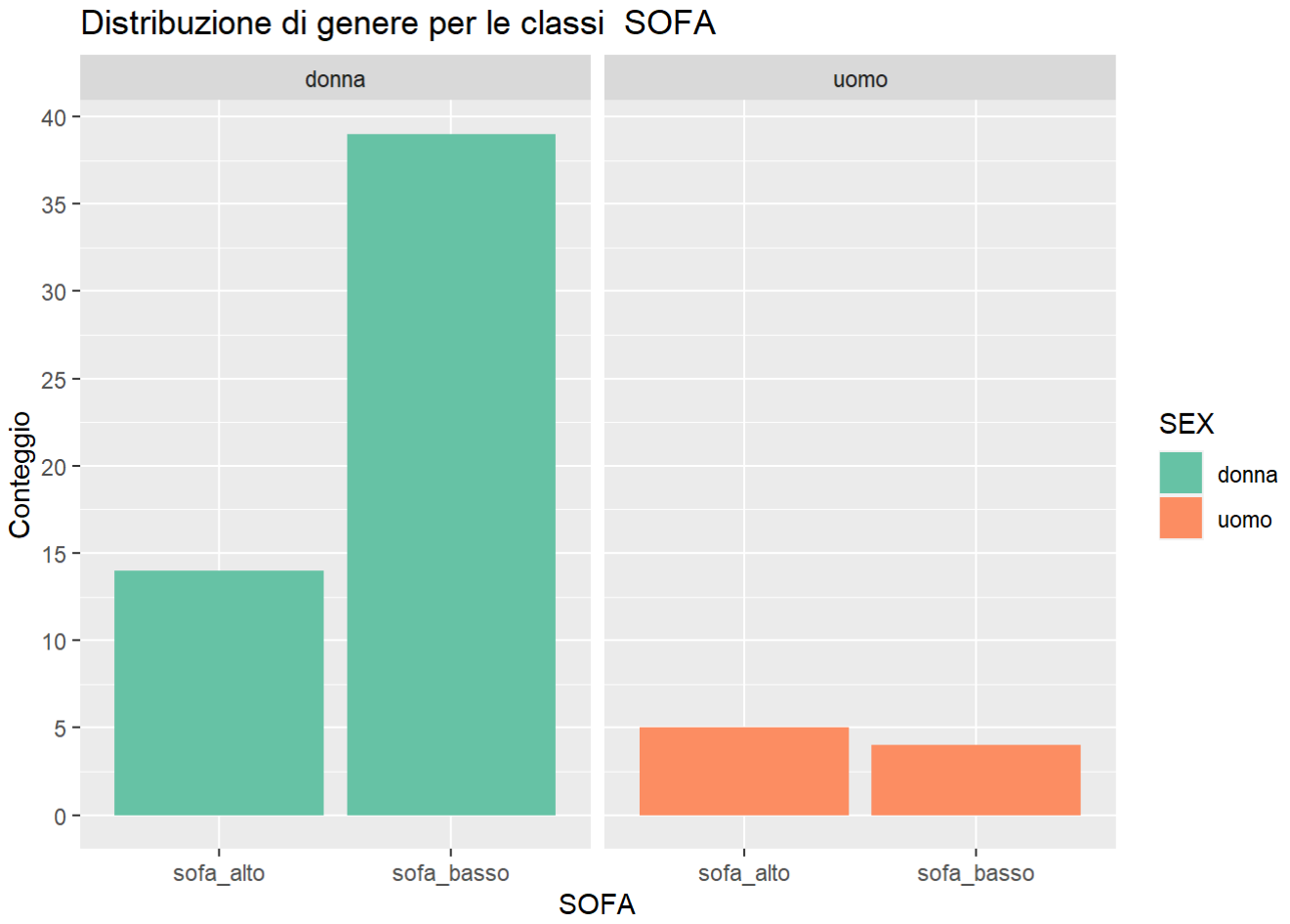
```
ggplot(data, aes(x = CCSCORE_classe, fill = SEX)) +
  geom_bar(position = "dodge") +
  facet_grid(. ~ SEX) +
  ggtitle("Distribuzione di genere per le classi  CCSCORE") +
  xlab("CCSCORE") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")+
scale_y_continuous(breaks = seq(0, 25, 2))
```



La maggior parte dei pazienti donna presenta un “CC-score” basso, questo può essere un indicatore di una migliore salute generale dell’individuo e di un minor rischio di complicanze mediche associate a malattie croniche, tuttavia, non significa necessariamente che l’individuo sia completamente privo di malattie o di altre condizioni mediche. 17 pazienti donne presentano un “CC-score” medio e 9 un “CC-score” alto, ciò potrebbe indicare che la popolazione in questione può avere un elevato rischio di malattie croniche.

Per gli uomini le classi sono più omogenee e non ci sono grandi differenze.

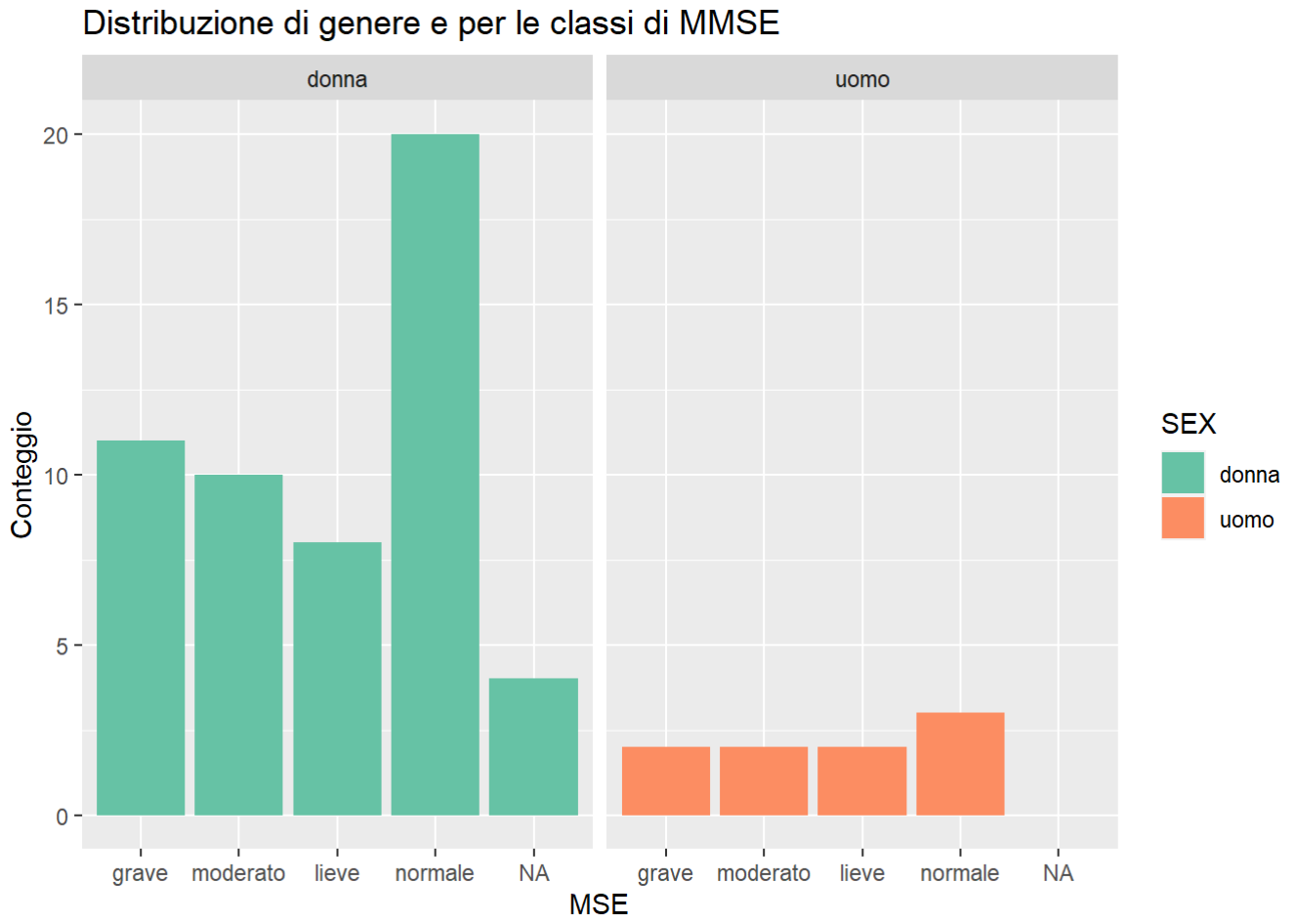
```
ggplot(data, aes(x = SOFAING_classe, fill = SEX)) +
  geom_bar(position = "dodge") +
  facet_grid(. ~ SEX) +
  ggtitle("Distribuzione di genere per le classi  SOFA") +
  xlab("SOFA") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")+
  scale_y_continuous(breaks = seq(0, 40, 5))
```



Un SOFA basso indica che la funzionalità degli organi è relativamente stabile e non ci sono segni di insufficienza grave, tuttavia, bisogna anche considerare il fatto che il SOFA non fornisce una valutazione completa dello stato di salute del paziente e non tiene conto di altri fattori importanti come la storia medica, la terapia attuale e il decorso della malattia.

La maggior parte dei pazienti donna ha un punteggio SOFA basso, ciò potrebbe suggerire che queste donne sono in una condizione relativamente stabile e non presentano una grave insufficienza d'organo.

```
ggplot(data, aes(x = MMSE_classe, fill = SEX)) +
  geom_bar(position = "dodge") +
  facet_grid(. ~ SEX) +
  ggtitle("Distribuzione di genere e per le classi di MMSE") +
  xlab("MSE") +
  ylab("Conteggio") +
  scale_fill_brewer(palette = "Set2")
```



La maggior parte dei pazienti ha un punteggio "MMSE" normale-lieve, ciò suggerisce che la loro funzione cognitiva è generalmente buona e che non vi sono gravi problemi di memoria, attenzione e ragionamento. Circa 15 pazienti su un totale di 62 hanno un punteggio "MMSE" moderato-grave, ciò suggerisce che questi pazienti potrebbero avere problemi significativi di memoria, attenzione e ragionamento.

Funzione cumulata per SOFA e CCSCORE

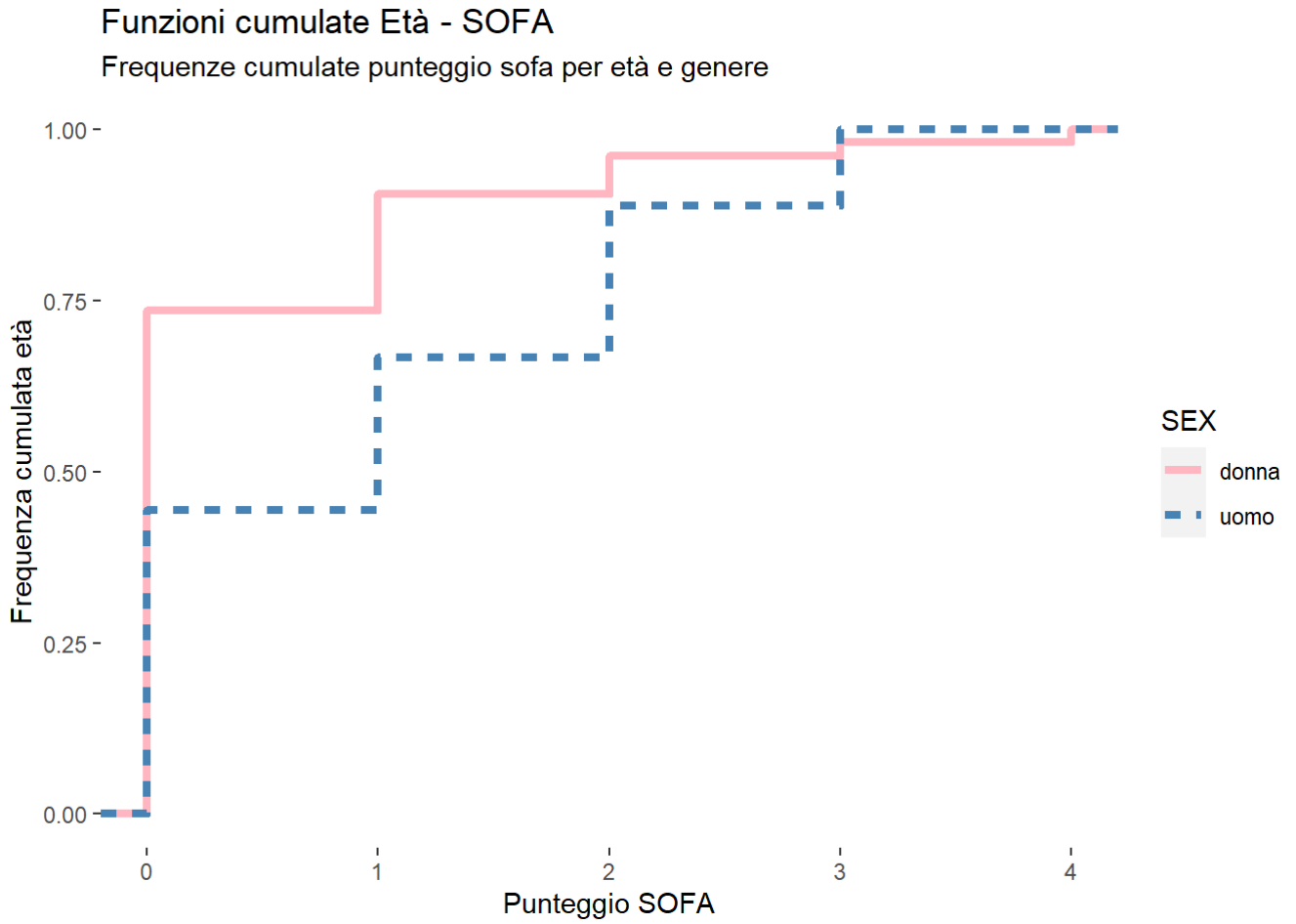
Una funzione cumulata dell'età rispetto allo "SOFA" può fornire informazioni sul modo in cui la salute dei pazienti cambia con l'età.

La seguente funzione rappresenta la distribuzione cumulata delle osservazioni rispetto alla variabile di età e al punteggio SOFA.

```
#funzione empirica cumulata dell'età divisa per sesso
data$SOFAING = as.numeric(data$SOFAING)
a <- ggplot(data, aes(x = SOFAING))+theme(panel.background = element_rect(fill = "whit
e" ))+ ggtitle("Plot of length \n by dose") + ylab("età cumulata") + xlab("Sofa")

#funzione empirica cumulata
pl =a + stat_ecdf(aes(color = SEX,linetype = SEX),

geom = "step", size = 1.5) +

scale_color_manual(values = c("lightpink", "steelblue"))+
labs(y = "f(eta)")+theme(panel.background = element_rect(fill = "white" ))
```

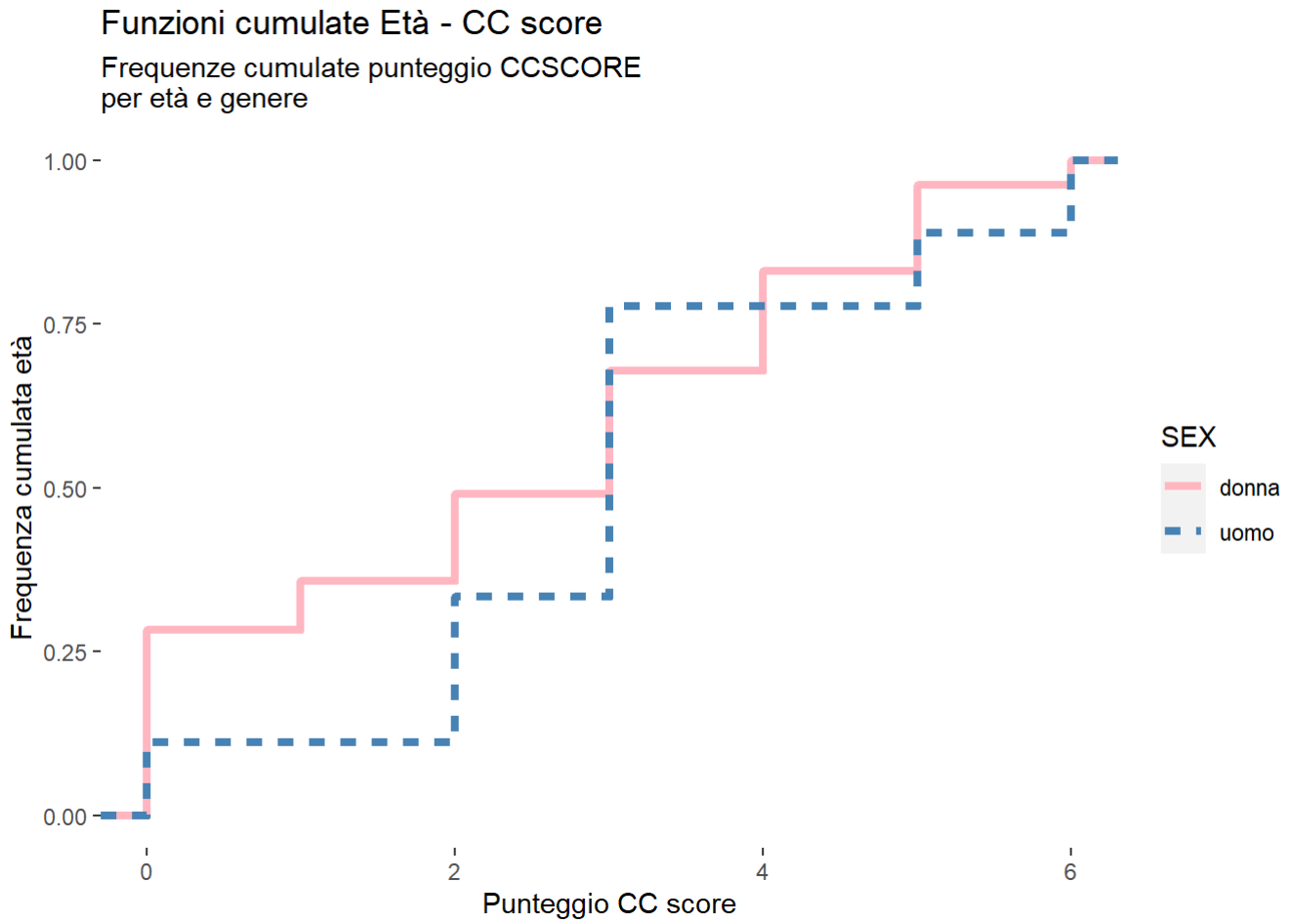
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```

```
bxp <- pl + labs(title = "Funzioni cumulate Età - SOFA",

subtitle = "Frequenze cumulate punteggio sofa per età e genere",
x = "Punteggio SOFA", y = "Frequenza cumulata età")

bxp
```



Il fatto che la funzione cumulata che rappresenta le donne sia sopra quella degli uomini per i punteggi SOFA da 0 a 2 potrebbe indicare che le donne in generale presentano meno comorbilità rispetto agli uomini nella stessa fascia di età. Tuttavia, il fatto che le funzioni si allineino per i punteggi SOFA da 2 a 4 potrebbe indicare che la relazione tra età e "SOFA" è simile per uomini e donne. L'analisi delle funzioni cumulate potrebbe suggerire la presenza di differenze di genere nella distribuzione dei punteggi SOFA in relazione all'età, ma è importante considerare ulteriori informazioni e condizioni specifiche del dataset per una corretta interpretazione.

```
#funzione empirica cumulata dell'età divisa per sesso
data$CCSCORE = as.numeric(data$CCSCORE)
a <- ggplot(data, aes(x = CCSCORE))+theme(panel.background = element_rect(fill = "whit
e" ))+ ggtitle("Plot of length \n by dose") + ylab("età cumulata") + xlab("CCSCORE")

#funzione empirica cumulata
pl =a + stat_ecdf(aes(color = SEX,linetype = SEX),

geom = "step", size = 1.5) +

scale_color_manual(values = c("lightpink", "steelblue"))+
labs(y = "f(eta)")+theme(panel.background = element_rect(fill = "white" ))

bxp <- pl + labs(title = "Funzioni cumulate Età - CC score",

subtitle = "Frequenze cumulate punteggio CCSCORE
per età e genere",
x = "Punteggio CC score", y = "Frequenza cumulata età")

bxp
```



Le funzioni cumulate per uomini e donne dopo il valore 2 si incrociano frequentemente, ciò potrebbe indicare che non vi è una forte associazione tra l'età e il CCscore e/o che non vi sono differenze significative tra uomini e donne.

Tuttavia, è importante considerare l'intervallo di età e il campione di pazienti inclusi nello studio. Ad esempio, se l'intervallo di età è ampio e il campione di pazienti è relativamente piccolo, le differenze tra gli uomini e le donne potrebbero non essere sufficientemente rappresentative.

Classificazione variabili principali rispetto a SOFA e CCSCORE

```
pvalue <- function(x, ...) {
    # Construct vectors of data y, and groups (strata) g
    y <- unlist(x)
    g <- factor(rep(1:length(x), times=sapply(x, length)))
    if (is.numeric(y)) {
        # For numeric variables, perform a standard 2-sample t-test
        p <- t.test(y ~ g)$p.value
    } else {
        # For categorical variables, perform a chi-squared test of independence
        p <- chisq.test(table(y, g))$p.value
    }
    # Format the p-value, using an HTML entity for the less-than sign.
    # The initial empty string places the output on the line below the variable label.
    c("", sub("<", "&lt;", format.pval(p, digits=3, eps=0.001)))
}

table1(~ SEX + AGE  + AGE_classi + BMI + BMI_classi  + MMSE_classe + CCSCORE_classe  |
SOFAING_classe, data=data, overall=F, extra.col=list('P-value'=pvalue))
```

```
## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect
```

	sofa_alto (N=19)	sofa_basso (N=43)	P-value
SEX			
donna	14 (73.7%)	39 (90.7%)	0.173
uomo	5 (26.3%)	4 (9.3%)	
AGE			
Mean (SD)	86.6 (6.25)	84.7 (6.73)	0.305
Median [Min, Max]	87.0 [73.0, 102]	85.0 [70.0, 99.0]	
AGE_classi			
70-75	1 (5.3%)	4 (9.3%)	0.828
75-80	1 (5.3%)	7 (16.3%)	
80-85	7 (36.8%)	12 (27.9%)	

	sofa_alto (N=19)	sofa_basso (N=43)	P-value
85-90	6 (31.6%)	12 (27.9%)	
90-95	3 (15.8%)	5 (11.6%)	
95-100+	1 (5.3%)	3 (7.0%)	
BMI			
Mean (SD)	24.0 (5.17)	23.4 (4.68)	0.685
Median [Min, Max]	23.4 [17.3, 35.6]	23.2 [12.9, 34.5]	
Missing	0 (0%)	3 (7.0%)	
BMI_classi			
normopeso	11 (57.9%)	22 (51.2%)	0.868
sottopeso	3 (15.8%)	5 (11.6%)	
sovrappeso	5 (26.3%)	13 (30.2%)	
Missing	0 (0%)	3 (7.0%)	
MMSE_classe			
grave	4 (21.1%)	9 (20.9%)	0.687
moderato	3 (15.8%)	9 (20.9%)	
lieve	2 (10.5%)	8 (18.6%)	
normale	9 (47.4%)	14 (32.6%)	
Missing	1 (5.3%)	3 (7.0%)	
CCSCORE_classe			
cc_alto	5 (26.3%)	6 (14.0%)	0.0262
cc_basso	4 (21.1%)	25 (58.1%)	
cc_medio	10 (52.6%)	12 (27.9%)	

In questa tabella viene presentata una descrizione delle variabili principali classificate in base al punteggio SOFA, dove per le variabili categoriche viene confrontata la numerosità tra i gruppi e le loro percentuali, mentre per quelle numeriche vengono osservate la media con la deviazione standard e la mediana con il range di appartenenza.

Si può notare come ci sia una maggior presenza di pazienti con SOFA basso, che è prevalentemente di genere femminile con un'età molto alta, superiore agli 80 anni in entrambe le categorie di SOFA e con un BMI normale. Inoltre, osservando la variabile "CC-score", si nota come nella categoria SOFA alto si definiscono pazienti con un punteggio di comorbilità medio/alto, mentre nella categoria SOFA basso il 58% dei pazienti ha un CC-score basso. Il P-value rappresenta la probabilità che l'effetto osservato o la differenza tra i gruppi o la relazione tra le variabili non siano dovuti al caso, ma siano effettivamente presenti nella popolazione di riferimento, è importante notare che in questo caso le variabili risultano non significativi al 5% , tranne che per la classe CC-score", quindi possiamo affermare che esiste una relazione significativa tra i due punteggi.

```
pvalue <- function(x, ...) {
    # Construct vectors of data y, and groups (strata) g
    y <- unlist(x)
    g <- factor(rep(1:length(x), times=sapply(x, length)))
    if (is.numeric(y)) {
        # For numeric variables, perform a standard 2-sample t-test
        p <- t.test(y ~ g)$p.value
    } else {
        # For categorical variables, perform a chi-squared test of independence
        p <- chisq.test(table(y, g))$p.value
    }
    # Format the p-value, using an HTML entity for the less-than sign.
    # The initial empty string places the output on the line below the variable label.
    c("", sub("<", "&lt;", format.pval(p, digits=3, eps=0.001)))
}

# create a new function for the inverted grouping factors
pvalue_inverted <- function(x, ...) {
    # Construct vectors of data y, and groups (strata) g
    y <- unlist(x)
    g <- factor(rep(1:length(x), times=sapply(x, length)))
    if (is.numeric(y)) {
        # For numeric variables, perform an ANOVA test
        p <- anova(lm(y ~ g))$"Pr(>F)"[1]
    } else {
        # For categorical variables, perform a chi-squared test of independence
        p <- chisq.test(table(g, y))$p.value
    }
    # Format the p-value, using an HTML entity for the less-than sign.
    # The initial empty string places the output on the line below the variable label.
    c("", sub("<", "&lt;", format.pval(p, digits=3, eps=0.001)))
}

# use the new function for the inverted grouping factors
table1(~ SEX + AGE + AGE_classi + BMI + BMI_classi + MMSE_classe +SOFAING_classe | CCS
CORE_classe, data = data, overall = FALSE, extra.col = list('P-value' = pvalue_inverte
d))
```

```
## Warning in chisq.test(table(g, y)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(g, y)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(g, y)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(g, y)): Chi-squared approximation may be incorrect

## Warning in chisq.test(table(g, y)): Chi-squared approximation may be incorrect
```

	cc_alto (N=11)	cc_basso (N=29)	cc_medio (N=22)	P-value
SEX				
donna	9 (81.8%)	26 (89.7%)	18 (81.8%)	0.683
uomo	2 (18.2%)	3 (10.3%)	4 (18.2%)	
AGE				
Mean (SD)	84.9 (7.18)	85.1 (6.42)	85.8 (6.80)	0.918
Median [Min, Max]	89.0 [71.0, 92.0]	84.0 [70.0, 99.0]	85.5 [74.0, 102]	
AGE_classi				
70-75	2 (18.2%)	2 (6.9%)	1 (4.5%)	0.851
75-80	0 (0%)	4 (13.8%)	4 (18.2%)	
80-85	3 (27.3%)	10 (34.5%)	6 (27.3%)	
85-90	4 (36.4%)	8 (27.6%)	6 (27.3%)	
90-95	2 (18.2%)	3 (10.3%)	3 (13.6%)	
95-100+	0 (0%)	2 (6.9%)	2 (9.1%)	
BMI				
Mean (SD)	25.6 (5.51)	24.3 (4.76)	21.9 (4.15)	0.0805
Median [Min, Max]	25.0 [17.3, 35.6]	23.5 [14.7, 34.5]	22.2 [12.9, 31.7]	
Missing	1 (9.1%)	2 (6.9%)	0 (0%)	
BMI_classi				
normopeso	4 (36.4%)	14 (48.3%)	15 (68.2%)	0.252
sottopeso	1 (9.1%)	3 (10.3%)	4 (18.2%)	
sovrappeso	5 (45.5%)	10 (34.5%)	3 (13.6%)	
Missing	1 (9.1%)	2 (6.9%)	0 (0%)	
MMSE_classe				
grave	3 (27.3%)	0 (0%)	10 (45.5%)	0.00428
moderato	1 (9.1%)	7 (24.1%)	4 (18.2%)	
lieve	3 (27.3%)	6 (20.7%)	1 (4.5%)	
normale	4 (36.4%)	14 (48.3%)	5 (22.7%)	
Missing	0 (0%)	2 (6.9%)	2 (9.1%)	
SOFAING_classe				
sofa_alto	5 (45.5%)	4 (13.8%)	10 (45.5%)	0.0262
sofa_basso	6 (54.5%)	25 (86.2%)	12 (54.5%)	

In questa tabella viene presentata una descrizione delle variabili principali classificate in base al CC-score, dove per le variabili categoriche viene confrontata la numerosità tra i gruppi e le loro percentuali, mentre per quelle numeriche vengono osservate la media con la deviazione standard e la mediana con il range di appartenenza.

Per quanto riguarda il sesso, la maggioranza dei pazienti in tutti e tre i gruppi sono donne, ma non c'è una differenza significativa tra i gruppi (p = 0,683). Inoltre, non ci sono differenze significative tra i gruppi per l'età e il BMI.

Tuttavia, c'è una differenza significativa nella distribuzione dei punteggi del MMSE tra i gruppi (p = 0,0163), con la maggior parte dei pazienti del gruppo "sofaing_classe" con disfunzione degli organi moderata che hanno un deficit cognitivo grave (22,7%).

Infine, la distribuzione dei pazienti in base all'età e al BMI, suddivisi in classi, non mostra differenze significative tra i gruppi.

CCSCORE e SOFA

Andiamo ad anaizzare la relazione tra le variabili SOFA e CCSCORE in base al genere.

```
table1(~ CCSCORE_classe | SOFAING_classe*SEX, data=data,overall=F, extra.col=list('P-v
alue'=pvalue))
```

```
## Warning in chisq.test(table(y, g)): Chi-squared approximation may be incorrect
```

	sofa_alto		sofa_basso		
	donna (N=14)	uomo (N=5)	donna (N=39)	uomo (N=4)	P-value
CCSCORE_classe					
cc_alto	4 (28.6%)	1 (20.0%)	5 (12.8%)	1 (25.0%)	0.244
cc_basso	3 (21.4%)	1 (20.0%)	23 (59.0%)	2 (50.0%)	
cc_medio	7 (50.0%)	3 (60.0%)	11 (28.2%)	1 (25.0%)	

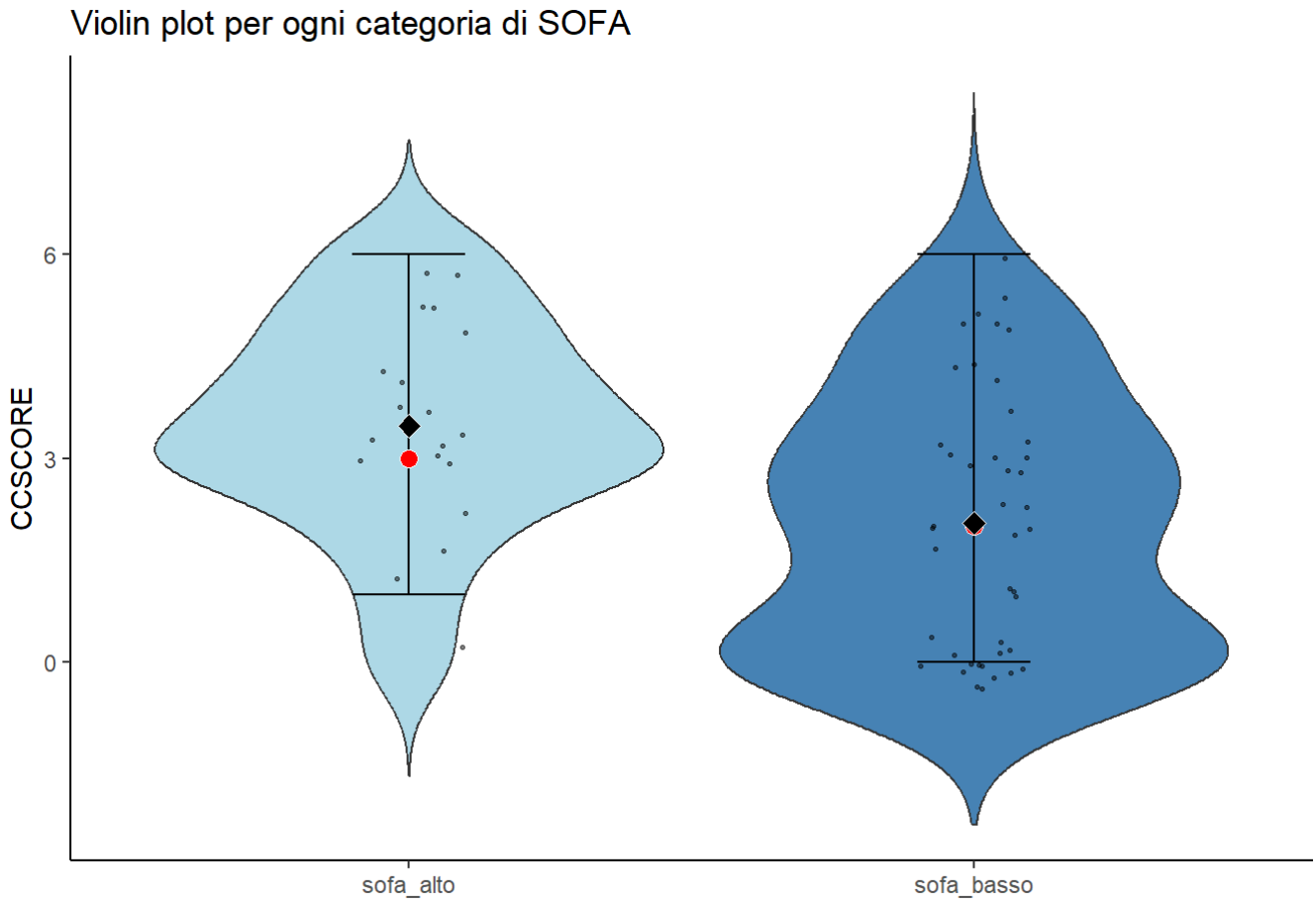
La tabella mostra che, tra le donne, il gruppo sofa_alto è presente in maggioranza tra coloro che appartengono alla classe di gravità cc_medio (50.0%), mentre tra gli uomini, il gruppo sofa_basso è più rappresentato tra coloro che appartengono alla classe di gravità cc_basso (60.0%).

Il test di significatività (P-value) indica che non c'è evidenza di una differenza significativa tra le distribuzioni dei gruppi in base alla classe di gravità tra donne e uomini.

```
my_colors <- c("lightblue", "steelblue")

ggplot(data, aes(x = SOFAING_classe, y = CCSCORE, fill = SOFAING_classe)) +
  geom_violin(trim = FALSE, scale = "width") +
  stat_boxplot(geom = 'errorbar', width = 0.2, notch = TRUE, varwidth = TRUE) +
  stat_summary(fun.y = median, geom = "point", shape = 21, size = 3, color = "white",
fill = "red") +
  stat_summary(fun.y = mean, geom = "point", shape = 23, size = 3, color = "white", fi
ll = "black") +
  geom_jitter(position = position_jitter(0.1), size = 0.5, alpha = 0.5) +
  scale_fill_manual(values = my_colors) +
  labs(x = " ", y = "CCSCORE") +
  ggtitle("Violin plot per ogni categoria di SOFA") +
  theme_bw() +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = "none")
```

```
## Warning in stat_boxplot(geom = "errorbar", width = 0.2, notch = TRUE, varwidth =
## TRUE): Ignoring unknown parameters: `notch` and `varwidth`
```

```
## Warning: The `fun.y` argument of `stat_summary()` is deprecated as of ggplot2 3.3.
0.
## i Please use the `fun` argument instead.
```



Osservando il grafico, confermiamo quanto detto precedentemente.

Analisi correlazione

La matrice di correlazione può essere utilizzata per identificare le relazioni tra le variabili e per selezionare le variabili per un ulteriori analisi.

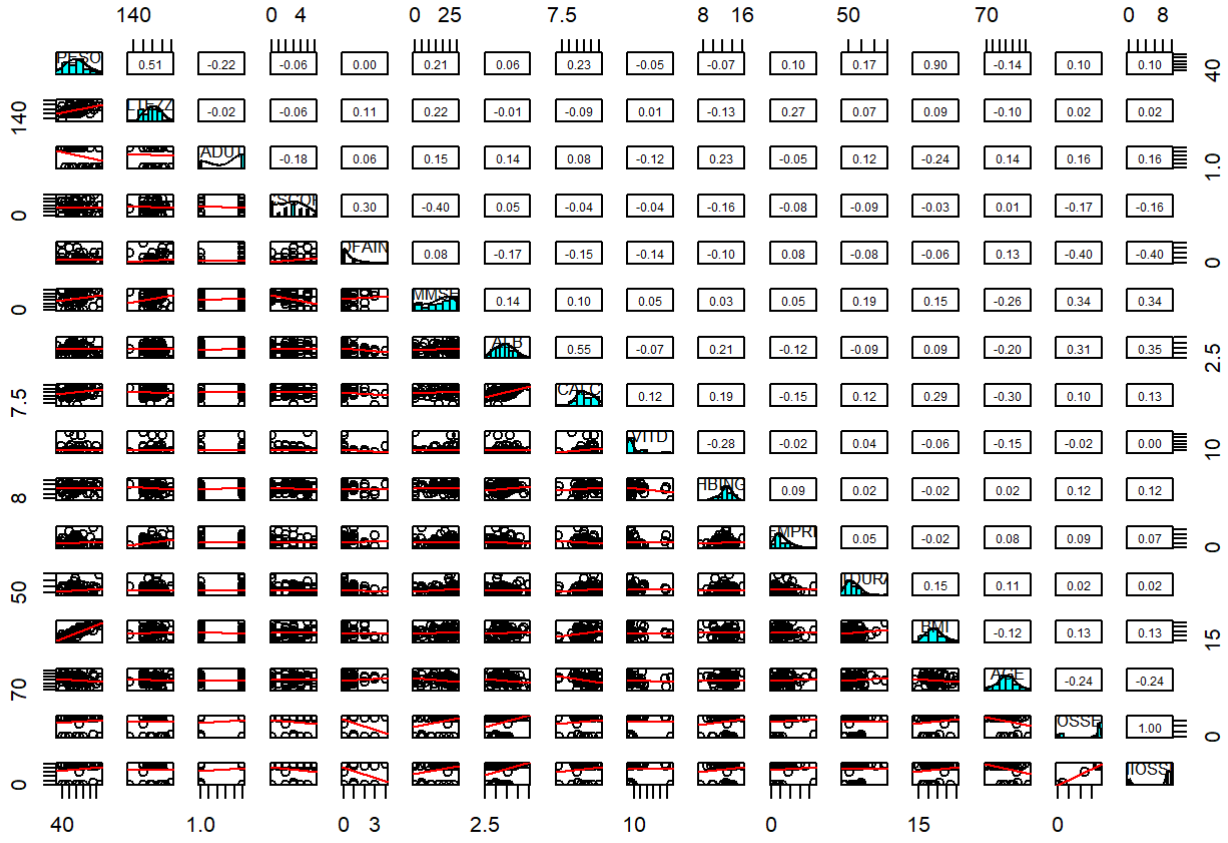
```
data_num <- select_if(data, is.numeric)#Prendo solo variabili numeriche
data_num <- subset(data_num, select = -PAZIENTE) # Elimino la variabile PAziente
corr<-cor(na.omit(data_num))# Elimino NA
corr
```

```
##                    PESO      ALTEZZA       CADUTE     CCSCORE     SOFAING
## PESO        1.000000000  0.513966114 -0.173039594 -0.02625805  0.27520298
## ALTEZZA     0.513966114  1.000000000 -0.005667324 -0.06504524  0.23199266
## CADUTE     -0.173039594 -0.005667324  1.000000000 -0.12557702  0.04288779
## CCSCORE    -0.026258046 -0.065045242 -0.125577016  1.00000000  0.23196332
## SOFAING     0.275202983  0.231992662  0.042887788  0.23196332  1.00000000
## MMSE        0.096489853  0.072993825  0.345170146 -0.41656138  0.05792937
## ALB        -0.107100282  0.033535735  0.301305734 -0.03734987 -0.19347232
## CALC        0.176624422 -0.136835939  0.061061994 -0.21748803 -0.21335445
## VITD       -0.008634677  0.050012755 -0.125274952 -0.06423237 -0.32217526
## HBING      -0.109474641 -0.094747211  0.270686485 -0.33043355 -0.14585254
## TEMPRIC     0.194242289  0.379132628 -0.087549100 -0.04544848  0.19277033
## INTDURAT    0.157645685  0.094160913  0.120216351  0.05471227 -0.12833942
## BMI         0.872541797  0.038302982 -0.168335192  0.01138605  0.18091373
## AGE         0.126030931  0.194954284 -0.006146420  0.10973040  0.09401960
## GGOSSERV   -0.014189552  0.030250453  0.333114520 -0.15783565 -0.45319767
## ANNIOSSERV -0.018346611  0.044981287  0.337503568 -0.16593261 -0.47146898
##                   MMSE         ALB        CALC         VITD       HBING
## PESO        0.09648985 -0.10710028  0.17662442 -0.008634677 -0.10947464
## ALTEZZA     0.07299383  0.03353573 -0.13683594  0.050012755 -0.09474721
## CADUTE      0.34517015  0.30130573  0.06106199 -0.125274952  0.27068648
## CCSCORE    -0.41656138 -0.03734987 -0.21748803 -0.064232370 -0.33043355
## SOFAING     0.05792937 -0.19347232 -0.21335445 -0.322175259 -0.14585254
## MMSE        1.00000000  0.28811564  0.29638288  0.036212054  0.16593613
## ALB         0.28811564  1.00000000  0.63312717 -0.083737624  0.31571353
## CALC        0.29638288  0.63312717  1.00000000  0.136035511  0.20837738
## VITD        0.03621205 -0.08373762  0.13603551  1.000000000 -0.30536799
## HBING       0.16593613  0.31571353  0.20837738 -0.305367989  1.00000000
## TEMPRIC     0.04913439 -0.06736174 -0.10340166 -0.007065364  0.02476724
## INTDURAT    0.30509863  0.11452577  0.14803735  0.019298593 -0.01918765
## BMI         0.09997724 -0.12538239  0.27274931 -0.037176002 -0.06682710
## AGE        -0.14908719 -0.40105696 -0.39538964 -0.176706898 -0.20563914
## GGOSSERV    0.41701127  0.43685811  0.24605806  0.017822038  0.06889697
## ANNIOSSERV  0.42766369  0.47804483  0.27237354  0.033515960  0.06589768
##                TEMPRIC    INTDURAT          BMI         AGE    GGOSSERV
## PESO        0.194242289  0.15764569  0.872541797  0.12603093 -0.01418955
## ALTEZZA     0.379132628  0.09416091  0.038302982  0.19495428  0.03025045
## CADUTE     -0.087549100  0.12021635 -0.168335192 -0.00614642  0.33311452
## CCSCORE    -0.045448477  0.05471227  0.011386053  0.10973040 -0.15783565
## SOFAING     0.192770328 -0.12833942  0.180913725  0.09401960 -0.45319767
## MMSE        0.049134395  0.30509863  0.099977240 -0.14908719  0.41701127
## ALB        -0.067361742  0.11452577 -0.125382385 -0.40105696  0.43685811
## CALC       -0.103401660  0.14803735  0.272749306 -0.39538964  0.24605806
## VITD       -0.007065364  0.01929859 -0.037176002 -0.17670690  0.01782204
## HBING       0.024767245 -0.01918765 -0.066827100 -0.20563914  0.06889697
## TEMPRIC     1.000000000  0.10918686  0.019270126  0.26519504  0.01762601
## INTDURAT    0.109186856  1.00000000  0.136444967  0.11514971  0.11701691
## BMI         0.019270126  0.13644497  1.000000000  0.02128455  0.01851376
## AGE         0.265195044  0.11514971  0.021284551  1.00000000 -0.13250621
## GGOSSERV    0.017626008  0.11701691  0.018513762 -0.13250621  1.00000000
## ANNIOSSERV  0.001682151  0.13401290  0.004805872 -0.13936478  0.99622089
```

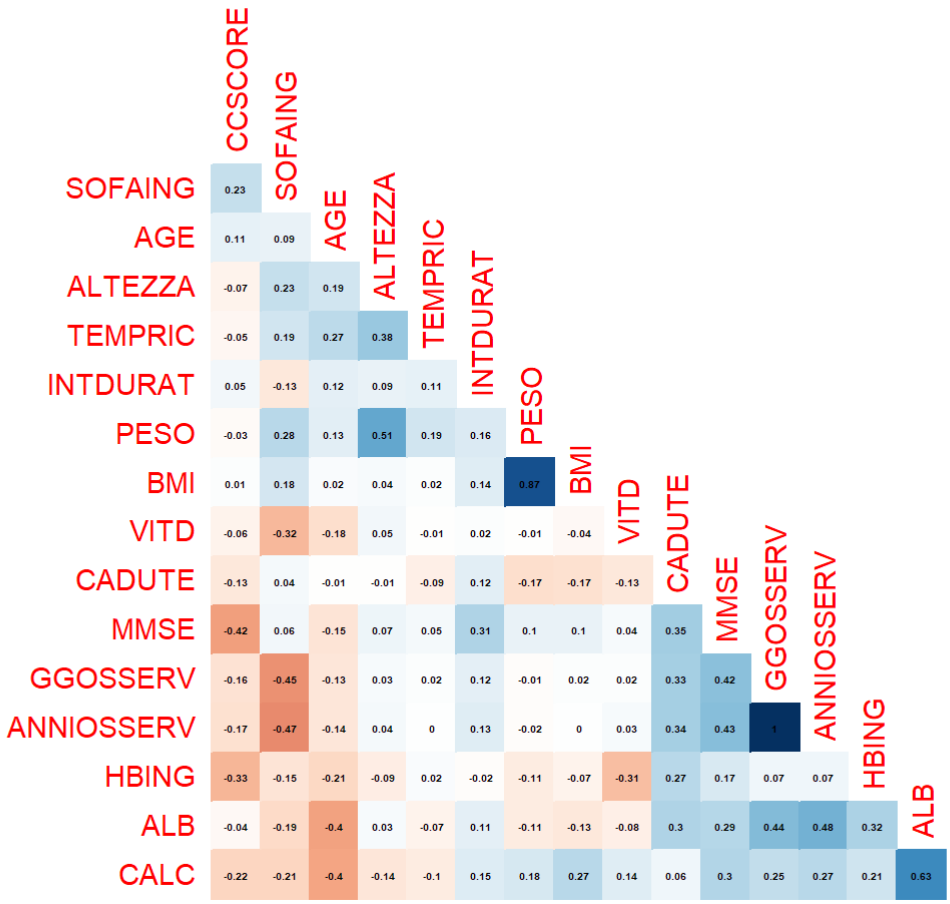
```
##                ANNIOSSERV
## PESO       -0.018346611
## ALTEZZA     0.044981287
## CADUTE      0.337503568
## CCSCORE    -0.165932605
## SOFAING    -0.471468983
## MMSE        0.427663689
## ALB         0.478044830
## CALC        0.272373538
## VITD        0.033515960
## HBING       0.065897679
## TEMPRIC     0.001682151
## INTDURAT    0.134012904
## BMI         0.004805872
## AGE        -0.139364780
## GGOSSERV    0.996220892
## ANNIOSSERV  1.000000000
```

La tabella mostra la matrice di correlazione tra diverse variabili. Ogni cella mostra il coefficiente di correlazione tra due variabili. Il coefficiente di correlazione varia da -1 a 1 e misura la forza e la direzione della relazione lineare tra le due variabili. Un coefficiente di 1 indica una perfetta correlazione positiva, mentre un coefficiente di -1 indica una perfetta correlazione negativa. Un coefficiente di 0 indica l'assenza di correlazione tra le due variabili.

```
pairs.panels(data_num, ellipses = F, lm=T,bg = c('blue','pink')[data$SEX],pch= 21, sta
rs=FALSE, cex = 1)
```



```
# Visualizzazione della matrice di correlazione
par(mar=c(0,0,2,0))
corrplot(corr, method = "color", type = "lower", order = "hclust",
addCoef.col = "black", number.cex = 0.3,
diag = FALSE, tl.cex = 0.9, tl.srt = 90,
cl.cex = 0.5, cl.pos = "n")
```



Considerando solo le variabili più importanti per questa analisi, è possibile osservare sia correlazioni positive che negative. Tuttavia, il confronto più rilevante è quello tra il punteggio “SOFA” e il “CC-score”. Sebbene il coefficiente di correlazione tra queste due variabili sia debolmente positivo (forse a causa della bassa numerosità del campione), è comunque significativo. Ciò significa che ad ogni aumento di una unità del “CC-score”, il “SOFA” aumenta di circa il 26%, in altre parole, se un paziente ha più malattie contemporaneamente, il rischio di disfunzione degli organi aumenta del 26%.

Survival analysis

Per finire, un’ulteriore analisi che si può effettuare con i dati a nostra disposizione è l’analisi di sopravvivenza. Come prima cosa creiamo una variabile ’EVENTO’che ha i seguenti valori:

1 = paziente deceduto

0 = paziente vivo

Ciò che si vuole dimostrare è che ad un elevato punteggio SOFA corrisponde un aumento del rischio di mortalità, la prima verifica viene fatta tramite la curva Kaplan-Meier.

```
data$EVENTO<-ifelse(is.na(data$`DATA DECESSO`), 0,1 )
data$EVENTO
```

```
##  [1] 0 1 1 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
## [39] 1 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 0 0 0 1
```

```
data_1 <- mutate(data, all = EVENTO != "0")# La funzione mutate() del pacchetto dplyr serve per creare o modificare colonne di un dataframe. In questo caso, la funzione mutate() viene utilizzata per creare una nuova colonna chiamata "all", che assume valore TRUE se l'evento ("EVENTO") del paziente non è "0" e FALSE altrimenti.
```

```
View(data)
```

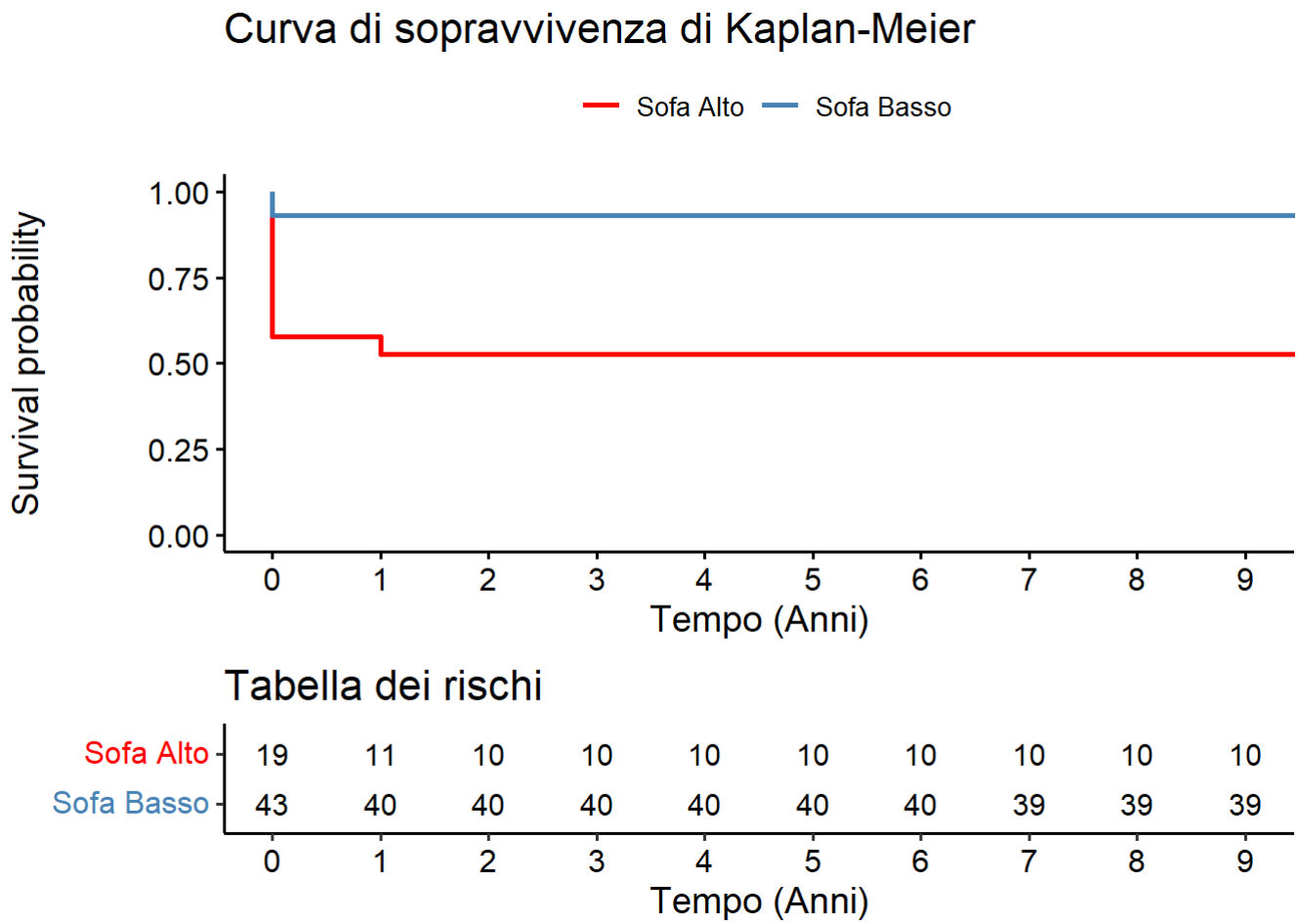
Modello di Kaplan-Meier

Costruiamo la curva di sopravvivenza

```
#Curva di sopravvivenza
curva_soprav <- survfit(Surv(ANNIOSSERV, all) ~ SOFAING_classe, data = data_1)
table_curva <- fortify(curva_soprav)
table_curva
```

```
##   time n.risk n.event n.censor      surv    std.err     upper     lower
## 1    0     19       8        0 0.5789474 0.19564640 0.8495200 0.3945523
## 2    1     11       1        0 0.5263158 0.21764288 0.8063143 0.3435488
## 3    9     10       0        5 0.5263158 0.21764288 0.8063143 0.3435488
## 4   10      5       0        5 0.5263158 0.21764288 0.8063143 0.3435488
## 5    0     43       3        0 0.9302326 0.04176345 1.0000000 0.8571216
## 6    6     40       0        1 0.9302326 0.04176345 1.0000000 0.8571216
## 7    9     39       0       20 0.9302326 0.04176345 1.0000000 0.8571216
## 8   10     19       0       19 0.9302326 0.04176345 1.0000000 0.8571216
##       strata
## 1  sofa_alto
## 2  sofa_alto
## 3  sofa_alto
## 4  sofa_alto
## 5 sofa_basso
## 6 sofa_basso
## 7 sofa_basso
## 8 sofa_basso
```

```
ggsurvplot(curva_soprav, title='Curva di sopravvivenza di Kaplan-Meier',
           risk.table = TRUE, xlab = "Tempo (Anni)", censor = F,
           xlim = c(0,9), break.x.by = 1,
           palette = c("red", "steelblue"), # Imposta la palette di colori
           legend.title = "", # Imposta il titolo della legenda
           legend.labs = c('Sofa Alto','Sofa Basso'), # Imposta le etichette della leg
enda
           legend.labs.size = 14, # Imposta la dimensione delle etichette della legend
a
           legend.position = "bottom", # Imposta la posizione della legenda
           legend.fontsize = 10, # Imposta la dimensione del font della legenda
           risk.table.col = "black", # Imposta il colore del testo della tabella
           risk.table.fontsize = 4, # Imposta la dimensione del font del testo della t
abella
           risk.table.title = "Tabella dei rischi", # Imposta il titolo della tabella
           risk.table.height = 0.3, # Imposta l'altezza della tabella
           risk.table.width = 0.5) # Imposta la larghezza della tabella
```



Come si può facilmente osservare dal grafico è presente una riduzione della curva che fa riferimento a coloro che hanno un punteggio SOFA alto. Ad esempio si nota come il 50% dei pazienti che avevano un punteggio SOFA alto all'inizio sono deceduti entro un anno, invece i pazienti con un punteggio relativamente basso si riducono del 10% circa. Bisogna tener conto che l'affidabilità dei risultati deve tenere in considerazione la bassa numerosità del campione di riferimento e del periodo temporale preso in considerazione, è deducibile però che in linea di massima il risultato con un maggior numero di osservazioni sarebbe simile.

Modello di Cox

L'obiettivo dell'analisi è valutare l'associazione tra la variabile indipendente SOFA e la sopravvivenza dei pazienti.

Il modello di Cox è un modello di regressione utilizzato per analizzare la sopravvivenza in base a un insieme di variabili esplicative, è un modello semiparametrico in quanto assume solo una distribuzione per il tempo di sopravvivenza, ma non fa alcuna ipotesi sulla forma funzionale della relazione. Dapprima utilizziamo il modello tenendo in considerazione la variabile di classificazione SOFA.

```
# Esegue il modello di Cox
cox_SOFA <- coxph(Surv(ANNIOSSERV, all) ~ SOFAING_classe, data = data_1)

# Stampa i risultati
summary(cox_SOFA)
```

```
## Call:
## coxph(formula = Surv(ANNIOSSERV, all) ~ SOFAING_classe, data = data_1)
##
##   n= 62, number of events= 12
##
##                             coef exp(coef) se(coef)      z Pr(>|z|)
## SOFAING_classesofa_basso -2.1358    0.1181   0.6681 -3.197  0.00139 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                          exp(coef) exp(-coef) lower .95 upper .95
## SOFAING_classesofa_basso    0.1181      8.464    0.0319    0.4376
##
## Concordance= 0.768  (se = 0.068 )
## Likelihood ratio test= 12.39  on 1 df,   p=4e-04
## Wald test            = 10.22  on 1 df,   p=0.001
## Score (logrank) test = 14.67  on 1 df,   p=1e-04
```

Il risultato più importante da notare è il p-value del test di Wald, pari a 0.001, che indica che la variabile indipendente SOFA è significativamente associata alla sopravvivenza dei pazienti. Inoltre, il coefficiente di regressione per la classe "sofa_basso" è -2.1358, il che significa che i pazienti con un punteggio SOFA basso hanno un tasso di rischio di mortalità significativamente inferiore rispetto a quelli con un punteggio SOFA alto.

Il valore di concordanza è pari a 0.768, il che indica un buon adattamento del modello ai dati. In generale, questi risultati suggeriscono che il punteggio SOFA può essere un fattore importante nella prognosi dei pazienti e può essere utilizzato come indicatore di gravità della malattia.

Ora utilzziamo il modello tenendo in considerazione anche le altre variabili.

```
cox_GEN <- coxph(Surv(ANNIOSSERV,all)~SOFAING_classe+AGE_classi+BMI_classi+CCSCORE_cla
sse+MMSE_classe+SEX,data = data_1)
```

```
## Warning in coxph.fit(X, Y, istrat, offset, init, control, weights = weights, :
## Loglik converged before variable 3,4,5,6 ; coefficient may be infinite.
```

```
summary(cox_GEN)
```

```
## Call:
## coxph(formula = Surv(ANNIOSSERV, all) ~ SOFAING_classe + AGE_classi +
##     BMI_classi + CCSCORE_classe + MMSE_classe + SEX, data = data_1)
##
##   n= 55, number of events= 12
##    (7 osservazioni eliminate a causa di valori mancanti)
##
##                               coef  exp(coef)   se(coef)      z Pr(>|z|)
## SOFAING_classesofa_basso -3.307e+00  3.662e-02  1.033e+00 -3.202  0.00136 **
## AGE_classi75-80           2.083e+00  8.026e+00  1.566e+04  0.000  0.99989
## AGE_classi80-85           1.968e+01  3.515e+08  1.115e+04  0.002  0.99859
## AGE_classi85-90           2.018e+01  5.797e+08  1.115e+04  0.002  0.99856
## AGE_classi90-95           1.706e+01  2.567e+07  1.115e+04  0.002  0.99878
## AGE_classi95-100+         1.916e+01  2.090e+08  1.115e+04  0.002  0.99863
## BMI_classisottopeso       4.055e+00  5.769e+01  2.085e+00  1.945  0.05177 .
## BMI_classisovrappeso      3.622e-01  1.436e+00  1.211e+00  0.299  0.76482
## CCSCORE_classecc_basso    9.819e-01  2.670e+00  1.469e+00  0.668  0.50386
## CCSCORE_classecc_medio   -3.349e-01  7.154e-01  1.245e+00 -0.269  0.78798
## MMSE_classemoderato      -1.839e+00  1.589e-01  1.738e+00 -1.058  0.29007
## MMSE_classelieve         -5.395e+00  4.537e-03  2.617e+00 -2.062  0.03925 *
## MMSE_classenormale       -3.657e+00  2.580e-02  1.487e+00 -2.459  0.01393 *
## SEXuomo                   3.235e-01  1.382e+00  8.775e-01  0.369  0.71240
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                           exp(coef) exp(-coef) lower .95 upper .95
## SOFAING_classesofa_basso 3.662e-02  2.731e+01 4.837e-03    0.2772
## AGE_classi75-80          8.026e+00  1.246e-01 0.000e+00       Inf
## AGE_classi80-85          3.515e+08  2.845e-09 0.000e+00       Inf
## AGE_classi85-90          5.797e+08  1.725e-09 0.000e+00       Inf
## AGE_classi90-95          2.567e+07  3.896e-08 0.000e+00       Inf
## AGE_classi95-100+        2.090e+08  4.785e-09 0.000e+00       Inf
## BMI_classisottopeso      5.769e+01  1.733e-02 9.693e-01 3433.5453
## BMI_classisovrappeso     1.436e+00  6.961e-01 1.339e-01   15.4132
## CCSCORE_classecc_basso   2.670e+00  3.746e-01 1.500e-01   47.5216
## CCSCORE_classecc_medio   7.154e-01  1.398e+00 6.231e-02    8.2137
## MMSE_classemoderato      1.589e-01  6.291e+00 5.267e-03    4.7971
## MMSE_classelieve         4.537e-03  2.204e+02 2.686e-05    0.7664
## MMSE_classenormale       2.580e-02  3.876e+01 1.398e-03    0.4761
## SEXuomo                  1.382e+00  7.236e-01 2.475e-01    7.7167
##
## Concordance= 0.967  (se = 0.021 )
## Likelihood ratio test= 35.62  on 14 df,   p=0.001
## Wald test            = 18.89  on 14 df,   p=0.2
## Score (logrank) test = 32.39  on 14 df,   p=0.004
```

L’output mostra che diverse variabili predittive sono statisticamente significative nella previsione della variabile di outcome. Ad esempio le variabili SOFA e MMSE, rispettivamente, con un punteggio basso e lieve/moderato hanno valori p inferiori a 0,05, indicando che sono predittori significativi dell’outcome al livello 0,05.

Conclusions

In sintesi, il presente progetto ha riguardato un’analisi di dati complessi, caratterizzati da una serie di problemi quali la presenza di dati mancanti, errori di digitazione, valori anomali e una grande quantità di variabili. Attraverso un’accurata fase di pulizia dei dati e di pre-processing, siamo riusciti a rimuovere tali problemi e ad ottenere un dataset pulito e ben strutturato, pronto per l’analisi.

Successivamente, abbiamo condotto un’analisi esplorativa dei dati per comprendere meglio la distribuzione e la relazione tra le variabili. Questa fase ha permesso di individuare alcune variabili che sembrano influenzare la sopravvivenza dei soggetti in esame.

Infine, abbiamo applicato un’analisi di sopravvivenza per valutare l’effetto di queste variabili sulla sopravvivenza dei soggetti. I risultati hanno mostrato che alcune variabili sono fortemente correlate con la sopravvivenza e possono essere utilizzate come predittori per identificare i soggetti a maggior rischio.