```r
# setup####
setwd("/Users/niccolofantone/Desktop/Rstudio")
library(readxl)
library(tidyr)
library(dplyr)
library(ggplot2)
library(randomForest)
library(shapviz)
library(kernelshap)
library(ranger)
library(corrplot)
library(car)
library(bestNormalize)
library(stargazer)
library(sandwich)
library(lmtest)
dati=read_xlsx("dataset.xlsx", na=c("n.d.", "n.s.", "n.a."))
dati=dati[1:733,]


# data cleaning####

dati_long <- dati %>%
  pivot_longer(
    cols = matches("^(EBITDA|DE|ROA|Oneri)[0-9]+$"),
    names_to = c("Variabile", "Anno"),
    names_pattern = "([A-Za-z]+)([0-9]+)",
    values_to = "Valore"
  ) %>%
  arrange(`Ragione sociale`, Variabile, Anno)

dati_large <- dati_long %>%
  pivot_wider(
    names_from = Variabile,
    values_from = Valore
  )




# aggiunta nuove feature####

dati_large$anno_concordato = as.numeric(substr(dati_large$`Data di inizio procedura/cessazione`,7,10))
dati_large$anno_ultimo_bilancio = as.numeric(substr(dati_large$`Chiusura bilancio`,1,4))
dati_large$Anno = 2024-as.numeric(dati_large$Anno)
dati_large$Area = ifelse(dati_large$`Indirizzo sede legale - Regione` %in% c("Lombardia", "Liguria", "Piemonte",
"Trentino-Alto Adige/Südtirol", "Friuli-Venezia Giulia", "Veneto", "Emilia-Romagna")
                         , "Nord", ifelse(dati_large$`Indirizzo sede legale - Regione` %in% c("Toscana", "Lazio",
"Umbria", "Marche", "Abruzzo"),"Centro", ifelse(is.na(dati_large$`Indirizzo sede legale - Regione`), NA, "Sud")) )

dati_large_filtered = na.omit(dati_large%>%dplyr::select(`Ragione sociale`,DE, Anno, EBITDA, ROA, Oneri, Area,
anno_concordato, anno_ultimo_bilancio))



dati_large_filtered= dati_large_filtered[dati_large_filtered$Anno>=dati_large_filtered$anno_concordato &
dati_large_filtered$Anno <= dati_large_filtered$anno_ultimo_bilancio,]
# filtro i casi con DE=0
dati_large_filtered = dati_large_filtered[dati_large_filtered$DE!=0,]

# analisi descrittive####
# numero aziende
length(unique(dati_large$`Ragione sociale`))
length(unique(dati_large_filtered$`Ragione sociale`))
# numero righe per anno
nrow(dati_large_filtered )
numeric_data <- dati_large_filtered[, sapply(dati_large_filtered, is.numeric)]

# Calcola media, mediana e deviazione standard
summary_stats <- data.frame(
  Media   = colMeans(numeric_data, na.rm = TRUE),
  Mediana = apply(numeric_data, 2, median, na.rm = TRUE),
  SD      = apply(numeric_data, 2, sd, na.rm = TRUE),
  max      = apply(numeric_data, 2, max, na.rm = TRUE),
  min      = apply(numeric_data, 2, min, na.rm = TRUE)

)

# numero osservazione con de>0
nrow(dati_large_filtered[dati_large_filtered$DE>0,])
# numero osservazione con de<0
nrow(dati_large_filtered[dati_large_filtered$DE<0,])

boxplot(dati_large_filtered$DE)
```

```
dati_large_filtered$EBITDA_t <- bestNormalize(dati_large_filtered$EBITDA)$x.t
dati_large_filtered$ROA_t    <- bestNormalize(dati_large_filtered$ROA)$x.t
dati_large_filtered$Oneri_t  <- bestNormalize(dati_large_filtered$Oneri)$x.t
dati_large_filtered$Anno_t   <- scale(dati_large_filtered$Anno)



dati_large_filtered$NegEq = ifelse(dati_large_filtered$DE> 0,0,1)
dati_large_filtered$absDe=  abs(dati_large_filtered$DE)
dati_large_filtered$absDet = log(dati_large_filtered$absDe+1)
# matriche di correlazione
round(cor(dati_large_filtered%>%select(absDet, EBITDA_t, ROA_t, Oneri_t, Anno_t)),2)

boxplot(dati_large_filtered$absDet)
plot(dati_large_filtered$absDet[dati_large_filtered$DE<=0]~dati_large_filtered$EBITDA_t[dati_large_filtered$DE<=0])



# double stage model####
step1 <- glm(NegEq ~ Anno_t + EBITDA_t  + Oneri_t + Area , data = dati_large_filtered, family="binomial")
summary(step1)


# eq positivi
step2 <- lm(absDet ~ Anno_t + EBITDA_t  + Oneri_t + Area , data =
dati_large_filtered[dati_large_filtered$NegEq==0,] )
summary(step2)


# eq negativi
step3 <- lm(absDet ~ Anno_t + EBITDA_t  + Oneri_t + Area , data =
dati_large_filtered[dati_large_filtered$NegEq==1,] )
summary(step3)


stargazer(step1,step2, step3 )


# Robustness check####

# step1####
step1 <- glm(NegEq ~ Anno_t + EBITDA_t  + Oneri_t + Area , data = dati_large_filtered, family="binomial")
summary(step1)

# test roa
step11 <- glm(NegEq ~ Anno_t + ROA_t  + Oneri_t + Area , data = dati_large_filtered, family="binomial")
summary(step11)

# test outliers
res_std <- rstandard(step1)

outliers <- abs(res_std) > 3
dati_no_out <- dati_large_filtered[!outliers, ]
step1_robust <- glm(NegEq ~ Anno_t + EBITDA_t  + Oneri_t + Area , data = dati_no_out, family="binomial")
summary(step1_robust)

# VIF
vif(step1)

# Omoscedasticity
bptest(step1)

vc_rob <- vcovHC(step1, type = "HC1")
coeftest(step1, vc_rob)# standard error robusti

# step 2a#####
step2 <- lm(absDet ~ Anno_t + EBITDA_t    + Oneri_t + Area , data =
dati_large_filtered[dati_large_filtered$NegEq==0,] )
summary(step2)

# test roa
step21 <- lm(absDet ~ Anno_t + ROA_t  + Oneri_t + Area , data = dati_large_filtered[dati_large_filtered$NegEq==0,]
)
summary(step21)

# test outliers
res_std <- rstandard(step2)

outliers <- abs(res_std) > 3
dati_no_out <- dati_large_filtered[dati_large_filtered$NegEq == 0, ][!outliers, ]
step2_robust <- lm(absDet ~ Anno_t + EBITDA_t + Oneri_t + Area, data = dati_no_out)
summary(step2_robust)
# VIF
vif(step2)

# Omoscedasticity
bptest(step2)

# Matrice di varianza-covarianza robusta (HC)
```

```
vc_rob <- vcovHC(step2, type = "HC1")
coeftest(step2, vc_rob)# standard error robusti

# step 2b#####
step3 <- lm(absDet ~ Anno_t + EBITDA_t    + Oneri_t + Area , data =
dati_large_filtered[dati_large_filtered$NegEq==1,] )
summary(step3)

# test roa
step31 <- lm(absDet ~ Anno_t + ROA_t  + Oneri_t + Area , data = dati_large_filtered[dati_large_filtered$NegEq==1,]
)
summary(step31)

# test outliers
res_std <- rstandard(step3)

outliers <- abs(res_std) > 3
dati_no_out <- dati_large_filtered[dati_large_filtered$NegEq == 1, ][!outliers, ]
step3_robust <- lm(absDet ~ Anno_t + EBITDA_t + Oneri_t + Area, data = dati_no_out)
summary(step3_robust)
# VIF
vif(step3)

# Omoscedasticity
bptest(step3)

# Matrice di varianza-covarianza robusta (HC)
vc_rob <- vcovHC(step3, type = "HC1")
coeftest(step3, vc_rob)# standard error robusti



# Random Forest####
dati_large_filtered1=dati_large_filtered%>%dplyr::select(absDet, Anno_t, EBITDA_t , Oneri_t, Area, NegEq)

set.seed(123)
rf1 = ranger(absDet ~ Anno_t + EBITDA_t  + Oneri_t+ Area, data=dati_large_filtered1, num.trees=500)

rows = sample(1:nrow(dati_large_filtered1), 0.1*nrow(dati_large_filtered1))
dati_small = dati_large_filtered1[rows,]
#ps <- permshap(rf1, X = dati_small ,bg_X = dati_small)
ps <- permshap(rf1, X = dati_large_filtered1 ,bg_X = dati_small)

shp <- shapviz(ps)
help(sv_dependence)
shp %>%
  sv_dependence(v = "EBITDA_t", color_var = "NegEq")
shp %>%
  sv_dependence(v = "Oneri_t", color_var = "NegEq")
shp %>%
  sv_dependence(v = "Anno_t", color_var = "NegEq")
shp %>%
  sv_dependence(v = "Area", color_var = "NegEq")


rf2 <- ranger(absDet ~ Anno_t + EBITDA_t + Oneri_t + Area,
              data = dati_large_filtered1,
              num.trees = 500,
              importance = "impurity")  # oppure "permutation" per maggiore robustezza

# Estraggo l'importanza delle variabili
imp <- data.frame(
  Variabile = names(rf2$variable.importance),
  Importanza = rf2$variable.importance
)

# Ordino le variabili per importanza
imp <- imp[order(imp$Importanza, decreasing = TRUE), ]

# Plot
ggplot(imp, aes(x = reorder(Variabile, Importanza), y = Importanza)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(title = "Importanza delle variabili (Random Forest)",
       x = "Variabile",
       y = "Importanza") +
  theme_minimal(base_size = 14)
```