Micro Project

XXX

09/02/2021

```
# clear workspace
rm(list = ls())

# load needed libraries
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.0.3
```

```
library(pscl)
```

```
## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis
```

```
library(car)
```

```
## Loading required package: carData
```

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 4.0.3
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
# read dataset
data <- read_csv("C:/Users/samue/Downloads/Studium/Economics (Master - Vienna)/1. Semester/Microeconome
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_character(),
##   COWCODE = col_double(),
##   ELITE_FEMALE = col_double(),
##   EC_GR_2Y = col_double()
## )
## i Use `spec()` for the full column specifications.
```

```r
# check if import worked
head(data)
```

```
## # A tibble: 6 x 22
##   COWCODE STATE REG_START REG_END REG_PARTY REG_REINST ELITE_NAME ELITE_PARTY
##     <dbl> <chr> <chr>     <chr>   <chr>     <chr>      <chr>      <chr>
## 1     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Backe, He~ NSDAP
## 2     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Blomberg,~ Independent
## 3     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Bormann, ~ NSDAP
## 4     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Darré, Ri~ NSDAP
## 5     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Dönitz, K~ NSDAP
## 6     255 Germ~ 30/1/1933 23/5/1~ National~ Cabinet    Dorpmülle~ NSDAP
## # ... with 14 more variables: ELITE_BIRTHDATE <chr>, ELITE_DEATHDATE <chr>,
## #   ELITE_FEMALE <dbl>, ELITE_REENTER <chr>, ELITE_REEXIT <chr>,
## #   ELITE_ENTERAGE <chr>, ELITE_EXITAGE <chr>, ELITE_RETENURE <chr>,
## #   ELITE_EXITTYPE <chr>, ELITE_EXITFATE <chr>, ELITE_EXITLEADER <chr>,
## #   ELITE_REPOSITION <chr>, ELITE_OCCUPATION <chr>, EC_GR_2Y <dbl>
```

```r
head(data$ELITE_NAME)
```

```
## [1] "Backe, Herbert"              "Blomberg, Werner von"
## [3] "Bormann, Martin"             "Darré, Richard Walther"
## [5] "Dönitz, Karl"                "Dorpmüller, Julius Heinrich"
```

```r
table(data$ELITE_EXITFATE)
```

```
##
##             .     Execution         Exile  Incarcerated Incarceration
##            19            13             4             1            23
##           N/A No punishment No Punishment            OK
##             6             2           480            10
```

```r
table(data$ELITE_EXITTYPE)
```

```
##
##               Assassination                       Death
##                           1                          67
##              Death- natural          Death - accidental
##                           1                           1
##       Death - Assassination Death - Automobile accident
##                           1                           1
##              Death - combat             Death - natural
##                           1                           2
##             Death - Natural             Death - suicide
##                           8                           4
##                    Demotion                   Execution
##                         253                           2
##                   Expulsion               Regime Change
##                          87                         178
##                Resignation   Ruling Institution Change
##                         185                          10
```

```r
# create dummy for death during regime or at end
dim(data); n <- dim(data)[1]
```

```
## [1] 827  22
```

```r
death1 <- rep(1,n)

# dummy for first type of death
for(i in 1:n){
  death1[i] <- ifelse(data$ELITE_EXITTYPE[i] != 'Demotion'
                      && data$ELITE_EXITTYPE[i] != 'Expulsion'
                      && data$ELITE_EXITTYPE[i] != 'Death- natural'
                      && data$ELITE_EXITTYPE[i] != 'Regime Change'
                      && data$ELITE_EXITTYPE[i] != 'Resignation',1,0)}
table(death1)
```

```
## death1
##   0   1
## 704  98
```

```r
# dummy for second type of death
death2 <- rep(1,n)
for(i in 1:n){
  death2[i] <- ifelse(data$ELITE_EXITFATE[i] == 'Execution',1,0)
}
table(death2)
```

```
## death2
##   0   1
## 545  13
```

```r
# merge dummies
a <- which(death2==1)
b <- which(death1==1)
c <- c(a,b)
death <- rep(0,n)
death[c] <- 1
table(death)
```

```
## death
##   0   1
## 716 111
```

```r
# dummy for country being in europe
EUROPE <- rep(0,n)
for(i in 1:n){
  EUROPE[i] <- ifelse(data$STATE[i] == 'Germany' || data$STATE[i] == 'Poland'
                    || data$STATE[i] == 'East Germany' || data$STATE[i] == 'Hungary'
                    || data$STATE[i] == 'Norway' || data$STATE[i] == 'Romania'
                    || data$STATE[i] == 'Soviet Union',1,0)
}
table(EUROPE)
```

```
## EUROPE
##   0   1
## 209 618
```

```r
# dummy for military as occupation outside of regime
MIL <- rep(0,n)
for(i in 1:n){
  MIL[i] <- ifelse(data$ELITE_OCCUPATION[i] == 'Soldier'
                    || data$ELITE_OCCUPATION[i] == 'State Security'
```

```
                  || data$ELITE_OCCUPATION[i] == 'Army officer'
                  || data$ELITE_OCCUPATION[i] == 'Naval officer'
                  || data$ELITE_OCCUPATION[i] == 'Military Police officer'
                  || data$ELITE_OCCUPATION[i] == 'Police officer'
                  || data$ELITE_OCCUPATION[i] == 'Air Force Officer'
                  || data$ELITE_OCCUPATION[i] == 'Air Force officer',1,0)
}
table(MIL)
```

```
## MIL
##   0   1
## 714  95
```

```
# dummy for economists
ECON <- rep(0,n)
for(i in 1:n){
  ECON[i] <- ifelse(data$ELITE_OCCUPATION[i] == 'Economist'
                    || data$ELITE_OCCUPATION[i] == 'economist',1,0)
}
table(ECON)
```

```
## ECON
##   0   1
## 784  25
```

```
# create function to extract date from string
substrRight <- function(x, n){
  substr(x, nchar(x)-n+1, nchar(x))
}

# get regime end year
END <- data$REG_END
END <- substrRight(END, 4)
END <- as.numeric(END)
table(END)
```

```
## END
## 1945 1949 1958 1966 1968 1973 1977 1979 1983 1989 1990 1991 2011 2019
##   71   10    5    4    9    5   12   10   22  223  176  160    4    6
```

```
# get regime start year
START <- as.numeric(substrRight(data$REG_START, 4))
table(START)
```

```
## START
## 1922 1933 1942 1944 1945 1947 1948 1949 1955 1957 1963 1966 1969 1971 1973 1975
##  160   51   20  106  117   95   10  129    9    5    4    5   12   16   12   44
## 1976 1979 1992 2010
##   13    9    6    4
```

```
# take care of regimes that have not ended
END <- ifelse(is.na(END),2020,END)
table(END)
```

```
## END
## 1945 1949 1958 1966 1968 1973 1977 1979 1983 1989 1990 1991 2011 2019 2020
##   71   10    5    4    9    5   12   10   22  223  176  160    4    6  110
```

```r
# variable for regime duration
DURATION <- END-START
table(DURATION)
```

```
## DURATION
##   1   3   4   7   8  12  13  17  27  41  43  44  45  49  69  71
##  19  24  19  18  12  51   9  12   6  69  95 117 140  16 160  60
```

```r
# standardize regime start year
mean(START)
```

```
## [1] 1945.261
```

```r
START <- START-mean(START)

mean(END)
```

```
## [1] 1988.707
```

```r
END <- END - mean(END)

#View(data)



# make data numeric for the model
death <- as.numeric(death)
ELITE_FEMALE <- as.numeric(data$ELITE_FEMALE)
ELITE_ENTERAGE <- as.numeric(data$ELITE_ENTERAGE)
```

```
## Warning: NAs introduced by coercion
```

```r
ELITE_RETENURE <- as.numeric(data$ELITE_RETENURE)
```

```
## Warning: NAs introduced by coercion
```

```r
EC_GR_2Y <- as.numeric(data$EC_GR_2Y)

# create dataset only based on relevant variables
data0 <- cbind(data$ELITE_NAME,death,EUROPE,START,DURATION,ELITE_FEMALE,ELITE_ENTERAGE,ELITE_RETENURE,
               EC_GR_2Y,ECON,MIL,END)
data <- cbind(death,EUROPE,START,DURATION,ELITE_FEMALE,ELITE_ENTERAGE,ELITE_RETENURE,
               EC_GR_2Y,ECON,MIL,END)

# get vectors for specific people:
hitler <- c(1,data[21,-c(1,3)])
# check if person is correct
data0[21,8] == data[21,7]
```

```
## ELITE_RETENURE
##           TRUE
```

```r
goebbles <- c(1,data[13,-c(1,3)])

# check dimenstion for later calculations
length(hitler)
```

```
## [1] 10
```

```r
t(rep(1,length(hitler)))%*%hitler
```

```
##          [,1]
## [1,] 12.99564
```

```r
data <- as.data.frame(data)
data0 <- as.data.frame(data0)
data0 <- na.exclude(data0)
#View(data0)




# remove NAs
data <- na.exclude(data)



#View(data)


# logit model
model1 <- glm(death ~ EUROPE+START+DURATION+ELITE_FEMALE+ELITE_ENTERAGE+ELITE_RETENURE
              +EC_GR_2Y+ECON+MIL,
    family = binomial(link = 'logit'), data=data)
summary(model1)
```

```
##
## Call:
## glm(formula = death ~ EUROPE + START + DURATION + ELITE_FEMALE +
##     ELITE_ENTERAGE + ELITE_RETENURE + EC_GR_2Y + ECON + MIL,
##     family = binomial(link = "logit"), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1440  -0.5687  -0.4116  -0.3107   2.9320
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -4.54890    1.12984  -4.026 5.67e-05 ***
## EUROPE          -1.06026    0.57492  -1.844 0.065157 .
## START            0.05698    0.02149   2.652 0.007998 **
## DURATION         0.05515    0.01421   3.882 0.000104 ***
## ELITE_FEMALE   -15.55979  723.00312  -0.022 0.982830
## ELITE_ENTERAGE   0.01017    0.01389   0.732 0.464133
## ELITE_RETENURE   0.03543    0.01424   2.488 0.012839 *
## EC_GR_2Y        -0.20591    0.05481  -3.756 0.000172 ***
## ECON            -0.43318    0.75572  -0.573 0.566505
## MIL             -0.15738    0.46691  -0.337 0.736062
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 538.91  on 720  degrees of freedom
## Residual deviance: 490.04  on 711  degrees of freedom
## AIC: 510.04
##
## Number of Fisher Scoring iterations: 16
```

```
# pseudo R2s
pR2(model1)
```

```
## fitting null model for pseudo-r2
```

```
##           llh       llhNull            G2      McFadden          r2ML
## -245.01964860 -269.45408398   48.86887076    0.09068126    0.06553331
##          r2CU
##    0.12448740
```

```
# variance inflation factors
vif(model1)
```

```
##         EUROPE          START       DURATION   ELITE_FEMALE ELITE_ENTERAGE
##       3.543817      11.733995       5.790912       1.000000       1.219254
## ELITE_RETENURE       EC_GR_2Y           ECON            MIL
##       1.151339       4.189975       1.011020       1.394974
```

```
# cor(data) shows correlations across all variables
cor(EUROPE,START)
```

```
## [1] -0.7091193
```

```
# model without START
model2 <- glm(death ~ EUROPE+DURATION+ELITE_FEMALE+ELITE_ENTERAGE+ELITE_RETENURE
              +EC_GR_2Y+ECON+MIL+END,
    family = binomial(link = 'logit'), data = data)
summary(model2)
```

```
##
## Call:
## glm(formula = death ~ EUROPE + DURATION + ELITE_FEMALE + ELITE_ENTERAGE +
##     ELITE_RETENURE + EC_GR_2Y + ECON + MIL + END, family = binomial(link = "logit"),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1440  -0.5687  -0.4116  -0.3107   2.9320
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.073215   0.841672  -2.463 0.013770 *
## EUROPE          -1.060259   0.574921  -1.844 0.065157 .
## DURATION        -0.001833   0.011757  -0.156 0.876084
## ELITE_FEMALE   -15.559791 723.003116  -0.022 0.982830
## ELITE_ENTERAGE   0.010169   0.013891   0.732 0.464133
## ELITE_RETENURE   0.035427   0.014238   2.488 0.012839 *
## EC_GR_2Y        -0.205906   0.054814  -3.756 0.000172 ***
```

```
## ECON            -0.433185   0.755722  -0.573 0.566505
## MIL             -0.157384   0.466913  -0.337 0.736062
## END              0.056983   0.021486   2.652 0.007998 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 538.91  on 720  degrees of freedom
## Residual deviance: 490.04  on 711  degrees of freedom
## AIC: 510.04
##
## Number of Fisher Scoring iterations: 16
```

```
pR2(model2)
```

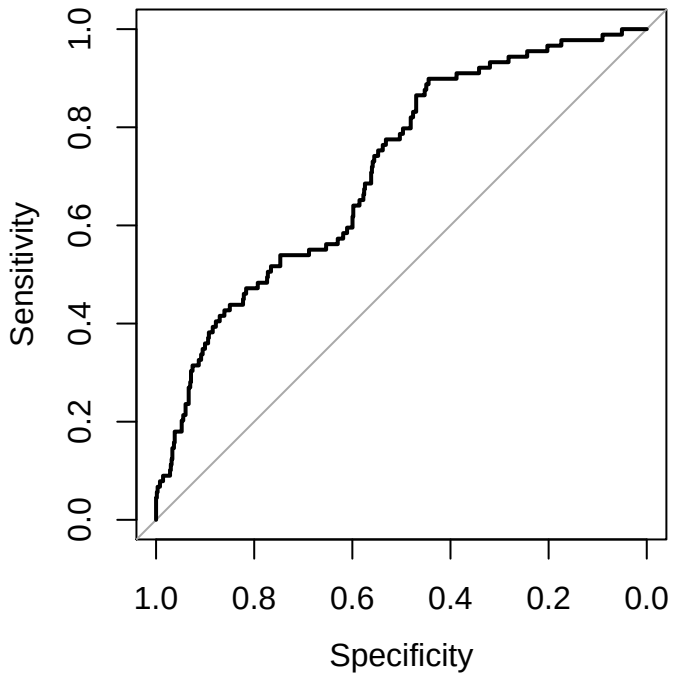
```
## fitting null model for pseudo-r2
```

```
##           llh       llhNull            G2      McFadden          r2ML
## -245.01964860 -269.45408398   48.86887076    0.09068126    0.06553331
##          r2CU
##    0.12448740
```

```
roc(data$death,predict.glm(model2,type='response'),plot=TRUE)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```




```
##
## Call:
## roc.default(response = data$death, predictor = predict.glm(model2,     type = "response"), plot = TRU
```

```
##
## Data: predict.glm(model2, type = "response") in 632 controls (data$death 0) < 89 cases (data$death 1)
## Area under the curve: 0.712
```

```
vif(model2)
```

```
##         EUROPE       DURATION   ELITE_FEMALE ELITE_ENTERAGE ELITE_RETENURE
##       3.543817       3.966153       1.000000       1.219254       1.151339
##       EC_GR_2Y           ECON            MIL            END
##       4.189975       1.011020       1.394974       8.011559
```

```
# get hit rate of the model with a 0.5 cutoff
cutoff <- 0.5
# get predicted probabilities
latent_pred <- predict.glm(model2,type = 'response')
# get binary result from the cutoff
latent_bin <- ifelse(latent_pred >= cutoff,1,0)
# hitrate
mean(latent_bin==data$death)
```

```
## [1] 0.8807212
```

```
# percentage by just guessing more likely outcome
1-mean(data$death)
```

```
## [1] 0.8765603
```

```
# check coefficients
length(model2$coefficients);length(hitler)
```

```
## [1] 10
```

```
## [1] 10
```

```
# check vectors for matching variables
hitler
```

```
##                       EUROPE       DURATION   ELITE_FEMALE ELITE_ENTERAGE
##        1.00000        1.00000       12.00000        0.00000       43.80822
## ELITE_RETENURE       EC_GR_2Y           ECON            MIL            END
##       12.25479      -13.36000        0.00000        0.00000      -43.70738
```

```
model2$coefficients
```

```
##    (Intercept)         EUROPE       DURATION   ELITE_FEMALE ELITE_ENTERAGE
##   -2.073214543   -1.060258788   -0.001833297  -15.559790990    0.010168763
## ELITE_RETENURE       EC_GR_2Y           ECON            MIL            END
##    0.035427232   -0.205906013   -0.433184552   -0.157384010    0.056982796
```

```
# get probabilities
Pr_hitler <- 1/(1+exp(-t(model2$coefficients)%*%(hitler)))
Pr_hitler
```

```
##           [,1]
## [1,] 0.1175843
```

```
Pr_goebbles <- 1/(1+exp(-t(model2$coefficients)%*%(goebbles)))
Pr_goebbles
```

```
##           [,1]
## [1,] 0.1086048
```

```r
# verify that the model would predict wrongly
latent_bin[21]==data$death[21]
```

```
##    21
## FALSE
```

```r
latent_bin[13]==data$death[13]
```

```
##    13
## FALSE
```

Get the predicted probabilities for Ghadaffi, Hitler, Mussolini, Goebbels, someone from SE Asia (no external war), Stalin (communist).