

R Notebook

Name: Gabriel Bentley

Date: 10/16/22

Dataset: Portuguese Elections

<https://archive.ics.uci.edu/ml/datasets/Real-time+Election+Results%3A+Portugal+2019>

Data set information

A data set describing the evolution of results in the Portuguese Parliamentary Elections of October 6th 2019. The data spans a time interval of 4 hours and 25 minutes, in intervals of 5 minutes, concerning the results of the 27 parties involved in the electoral event. The data set is tailored for predictive modelling tasks, mostly focused on numerical forecasting tasks. Regardless, it allows for other tasks such as ordinal regression or learn-to-rank. Provide a short description of your data set (less than 200 characters).

Attribute information

TimeElapsed (Numeric): Time (minutes) passed since the first data acquisition time (timestamp): Date and time of the data acquisition territoryName (string): Short name of the location (district or nation-wide) totalMandates (numeric): MP's elected at the moment availableMandates (numeric): MP's left to elect at the moment numParishes (numeric): Total number of parishes in this location numParishesApproved (numeric): Number of parishes approved in this location blankVotes (numeric): Number of blank votes blankVotesPercentage (numeric): Percentage of blank votes nullVotes (numeric): Number of null votes nullVotesPercentage (numeric): Percentage of null votes votersPercentage (numeric): Percentage of voters subscribedVoters (numeric): Number of subscribed voters in the location totalVoters (numeric): Percentage of blank votes (incorrect information) pre.blankVotes (numeric): Number of blank votes (previous election) pre.blankVotesPercentage (numeric): Percentage of blank votes (previous election) pre.nullVotes (numeric): Number of null votes (previous election) pre.nullVotesPercentage (numeric): Percentage of null votes (previous election) pre.votersPercentage (numeric): Percentage of voters (previous election) pre.subscribedVoters (numeric): Number of subscribed voters in the location (previous election) pre.totalVoters (numeric): Percentage of blank votes (previous election)(incorrect) Party (string): Political Party Mandates (numeric): MP's elected at the moment for the party in a given district Percentage (numeric): Percentage of votes in a party validVotesPercentage (numeric): Percentage of valid votes in a party Votes (numeric): Percentage of party votes (incorrect) Hondt (numeric): Number of MP's according to the distribution of votes now FinalMandates (numeric): Target: final number of elected MP's in a district/national-level

Read in the data set and remove unwanted attributes

```
library(e1071)
library(MASS)

df <- read.csv("ElectionData.csv")

df <- subset(df, select = -c(time, Mandates, Hondt, FinalMandates, nullVotesPercentage, votersPercentage))

df$territoryName <- as.factor(x = df$territoryName)
df$Party <- as.factor(x = df$Party)
```

split the data set into train, test, and valid

```
set.seed(6229)
spec <- c(train=.6, test=.2, validate=.2)
i <- sample(cut(1:nrow(df),
               nrow(df)*cumsum(c(0,spec))), labels=names(spec)))
train <- df[i=="train",]
test  <- df[i=="test",]
vald  <- df[i=="validate",]
```

Explore the data set

```
names(train)
```

```
## [1] "TimeElapsed"      "territoryName"
## [3] "totalMandates"    "availableMandates"
## [5] "numParishes"      "numParishesApproved"
## [7] "blankVotes"       "nullVotes"
## [9] "subscribedVoters" "totalVoters"
## [11] "pre.blankVotes"   "pre.blankVotesPercentage"
## [13] "pre.nullVotes"    "pre.nullVotesPercentage"
## [15] "pre.votersPercentage" "pre.subscribedVoters"
## [17] "pre.totalVoters"  "Party"
## [19] "Votes"
```

```
summary(train)
```

```
##   TimeElapsed      territoryName totalMandates  availableMandates
##   Min.   : 0.0    Porto           : 689    Min.   : 0.00    Min.   : 0.00
##   1st Qu.: 65.0   Leiria          : 684    1st Qu.: 1.00    1st Qu.: 0.00
##   Median :135.0   Território Nacional: 682    Median : 4.00    Median : 3.00
##   Mean   :133.1   Viseu           : 664    Mean   :11.79    Mean   :11.29
##   3rd Qu.:200.0   Braga           : 661    3rd Qu.: 9.00    3rd Qu.: 9.00
##   Max.   :265.0   Coimbra         : 661    Max.   :226.00    Max.   :226.00
##   (Other)          :8944
##   numParishes      numParishesApproved  blankVotes      nullVotes
##   Min.   : 54.0    Min.   : 3.0    Min.   : 19    Min.   : 39
##   1st Qu.: 75.0    1st Qu.: 67.0    1st Qu.:1188    1st Qu.:1094
##   Median :147.0    Median :120.0    Median :2998    Median :2232
##   Mean   :310.1    Mean   :263.8    Mean   :9027    Mean   :6251
##   3rd Qu.:242.0    3rd Qu.:208.0    3rd Qu.:6889    3rd Qu.:4073
##   Max.   :3092.0    Max.   :3092.0    Max.   :129599    Max.   :88539
##
##   subscribedVoters  totalVoters      pre.blankVotes  pre.blankVotesPercentage
##   Min.   : 5767    Min.   : 2833    Min.   : 32    Min.   :0.800
##   1st Qu.:122987    1st Qu.: 62671    1st Qu.:1130    1st Qu.:1.740
##   Median :228954    Median :106449    Median :2595    Median :2.030
##   Mean   :638360    Mean   :344882    Mean   :7740    Mean   :2.071
##   3rd Qu.:380489    3rd Qu.:206918    3rd Qu.:5929    3rd Qu.:2.290
##   Max.   :9343084    Max.   :5092424    Max.   :112666    Max.   :3.660
##
##   pre.nullVotes  pre.nullVotesPercentage pre.votersPercentage
##   Min.   : 40    Min.   :1.140    Min.   :40.87
##   1st Qu.:1124    1st Qu.:1.520    1st Qu.:51.21
```

```
## Median : 2141    Median :1.690          Median :56.26
## Mean   : 6016    Mean   :1.777          Mean   :54.57
## 3rd Qu.: 3967    3rd Qu.:1.970          3rd Qu.:58.21
## Max.   :86473    Max.   :3.120          Max.   :62.58
```

```
##
## pre.subscribedVoters pre.totalVoters      Party      Votes
## Min.   : 6383      Min.   : 3215    A       : 700    Min.   : 1
## 1st Qu.: 128926     1st Qu.: 69644    IL      : 700    1st Qu.: 237
## Median : 228497     Median : 111998    PAN     : 698    Median : 791
## Mean   : 648757     Mean   : 365673    CDS-PP  : 688    Mean   : 16308
## 3rd Qu.: 393314     3rd Qu.: 227620    PNR     : 686    3rd Qu.: 4679
## Max.   :9439701     Max.   :5380451    PCP-PEV : 684    Max.   :1846809
##                                     (Other):8829
```

```
str(train)
```

```
## 'data.frame': 12985 obs. of 19 variables:
## $ TimeElapsed : int 0 0 0 0 0 0 0 0 0 0 ...
## $ territoryName : Factor w/ 21 levels "Açores","Aveiro",...: 18 18 18 18 18 18 18 18 18 18 ...
## $ totalMandates : int 0 0 0 0 0 0 0 0 0 0 ...
## $ availableMandates : int 226 226 226 226 226 226 226 226 226 226 ...
## $ numParishes : int 3092 3092 3092 3092 3092 3092 3092 3092 3092 3092 ...
## $ numParishesApproved : int 1081 1081 1081 1081 1081 1081 1081 1081 1081 1081 ...
## $ blankVotes : int 9652 9652 9652 9652 9652 9652 9652 9652 9652 9652 ...
## $ nullVotes : int 8874 8874 8874 8874 8874 8874 8874 8874 8874 8874 ...
## $ subscribedVoters : int 752529 752529 752529 752529 752529 752529 752529 752529 752529 752529 ...
## $ totalVoters : int 386497 386497 386497 386497 386497 386497 386497 386497 386497 386497 ...
## $ pre.blankVotes : int 8317 8317 8317 8317 8317 8317 8317 8317 8317 8317 ...
## $ pre.blankVotesPercentage: num 1.94 1.94 1.94 1.94 1.94 1.94 1.94 1.94 1.94 1.94 ...
## $ pre.nullVotes : int 8171 8171 8171 8171 8171 8171 8171 8171 8171 8171 ...
## $ pre.nullVotesPercentage : num 1.91 1.91 1.91 1.91 1.91 1.91 1.91 1.91 1.91 1.91 ...
## $ pre.votersPercentage : num 52.7 52.7 52.7 52.7 52.7 ...
## $ pre.subscribedVoters : int 813743 813743 813743 813743 813743 813743 813743 813743 813743 813743 ...
## $ pre.totalVoters : int 428546 428546 428546 428546 428546 428546 428546 428546 428546 428546 ...
## $ Party : Factor w/ 21 levels "A","B.E.,"CDS-PP",...: 16 3 4 21 1 7 6 14 15 20 ...
## $ Votes : int 128624 18923 3659 3412 1870 1872 1236 795 803 740 ...
```

```
head(train, n = 20)
```

```
##      TimeElapsed      territoryName totalMandates availableMandates numParishes
## 2           0 Território Nacional              0              226          3092
## 4           0 Território Nacional              0              226          3092
## 7           0 Território Nacional              0              226          3092
## 8           0 Território Nacional              0              226          3092
## 10          0 Território Nacional              0              226          3092
## 11          0 Território Nacional              0              226          3092
## 13          0 Território Nacional              0              226          3092
## 15          0 Território Nacional              0              226          3092
## 16          0 Território Nacional              0              226          3092
## 17          0 Território Nacional              0              226          3092
## 19          0 Território Nacional              0              226          3092
## 21          0 Território Nacional              0              226          3092
## 22          0 Aveiro                        0              16           147
## 23          0 Aveiro                        0              16           147
## 25          0 Aveiro                        0              16           147
## 26          0 Aveiro                        0              16           147
```

## 27	0	Aveiro	0	16	147
## 28	0	Aveiro	0	16	147
## 30	0	Aveiro	0	16	147
## 33	0	Aveiro	0	16	147
##	numParishesApproved	blankVotes	nullVotes	subscribedVoters	totalVoters
## 2	1081	9652	8874	752529	386497
## 4	1081	9652	8874	752529	386497
## 7	1081	9652	8874	752529	386497
## 8	1081	9652	8874	752529	386497
## 10	1081	9652	8874	752529	386497
## 11	1081	9652	8874	752529	386497
## 13	1081	9652	8874	752529	386497
## 15	1081	9652	8874	752529	386497
## 16	1081	9652	8874	752529	386497
## 17	1081	9652	8874	752529	386497
## 19	1081	9652	8874	752529	386497
## 21	1081	9652	8874	752529	386497
## 22	12	186	101	11953	6371
## 23	12	186	101	11953	6371
## 25	12	186	101	11953	6371
## 26	12	186	101	11953	6371
## 27	12	186	101	11953	6371
## 28	12	186	101	11953	6371
## 30	12	186	101	11953	6371
## 33	12	186	101	11953	6371
##	pre.blankVotes	pre.blankVotesPercentage	pre.nullVotes		
## 2	8317		1.94	8171	
## 4	8317		1.94	8171	
## 7	8317		1.94	8171	
## 8	8317		1.94	8171	
## 10	8317		1.94	8171	
## 11	8317		1.94	8171	
## 13	8317		1.94	8171	
## 15	8317		1.94	8171	
## 16	8317		1.94	8171	
## 17	8317		1.94	8171	
## 19	8317		1.94	8171	
## 21	8317		1.94	8171	
## 22	124		1.82	111	
## 23	124		1.82	111	
## 25	124		1.82	111	
## 26	124		1.82	111	
## 27	124		1.82	111	
## 28	124		1.82	111	
## 30	124		1.82	111	
## 33	124		1.82	111	
##	pre.nullVotesPercentage	pre.votersPercentage	pre.subscribedVoters		
## 2	1.91	52.66	813743		
## 4	1.91	52.66	813743		
## 7	1.91	52.66	813743		
## 8	1.91	52.66	813743		
## 10	1.91	52.66	813743		
## 11	1.91	52.66	813743		
## 13	1.91	52.66	813743		

```

## 15          1.91          52.66          813743
## 16          1.91          52.66          813743
## 17          1.91          52.66          813743
## 19          1.91          52.66          813743
## 21          1.91          52.66          813743
## 22          1.63          54.56          12511
## 23          1.63          54.56          12511
## 25          1.63          54.56          12511
## 26          1.63          54.56          12511
## 27          1.63          54.56          12511
## 28          1.63          54.56          12511
## 30          1.63          54.56          12511
## 33          1.63          54.56          12511
##      pre.totalVoters  Party  Votes
## 2          428546 PPD/PSD 128624
## 4          428546 CDS-PP 18923
## 7          428546 CH 3659
## 8          428546 R.I.R. 3412
## 10         428546 A 1870
## 11         428546 L 1872
## 13         428546 JPP 1236
## 15         428546 PDR 795
## 16         428546 PNR 803
## 17         428546 PURP 740
## 19         428546 MPT 642
## 21         428546 MAS 96
## 22         6826 PPD/PSD 2405
## 23         6826 PS 2187
## 25         6826 CDS-PP 391
## 26         6826 PCP-PEV 152
## 27         6826 PAN 130
## 28         6826 R.I.R. 103
## 30         6826 CH 36
## 33         6826 L 25

```

```
colSums(is.na(train))
```

```

##      TimeElapsed      territoryName      totalMandates
##           0           0           0
##      availableMandates      numParishes      numParishesApproved
##           0           0           0
##           blankVotes      nullVotes      subscribedVoters
##           0           0           0
##           totalVoters      pre.blankVotes      pre.blankVotesPercentage
##           0           0           0
##           pre.nullVotes      pre.nullVotesPercentage      pre.votersPercentage
##           0           0           0
##           pre.subscribedVoters      pre.totalVoters      Party
##           0           0           0
##           Votes
##           0

```

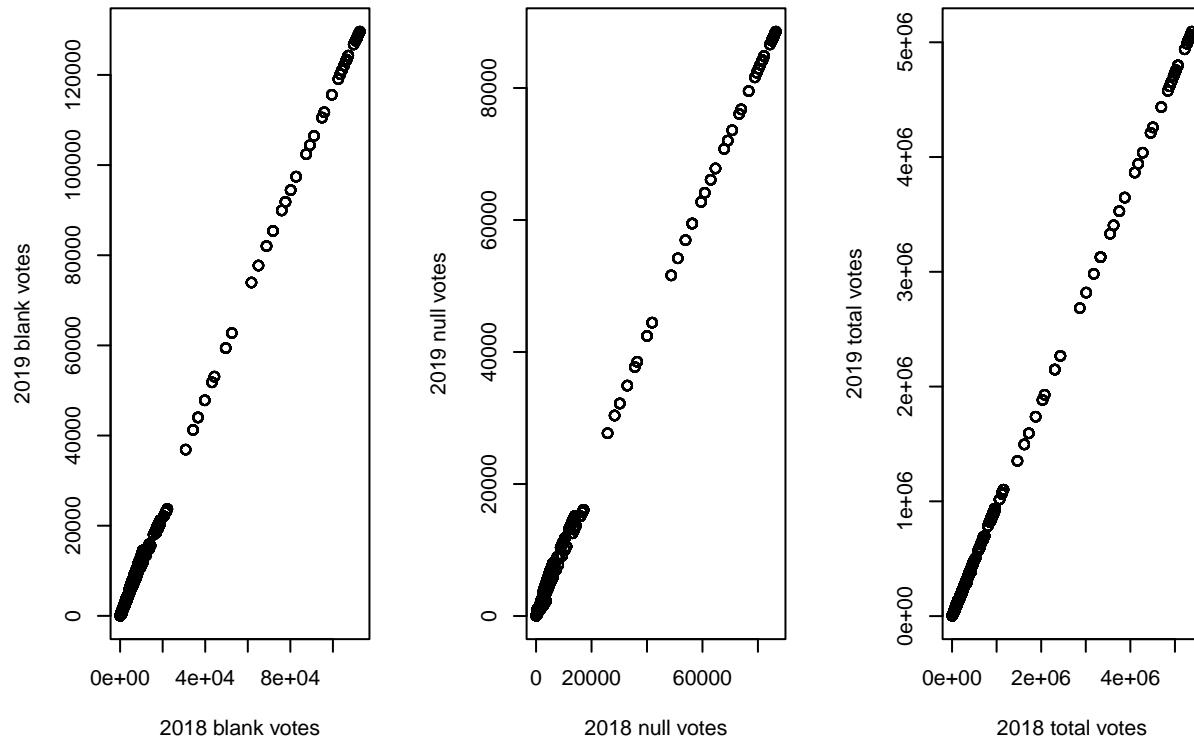
Graphically explore the data set

```
par(mfrow=c(1,3))
```

```
plot(train$pre.blankVotes, train$blankVotes, xlab="2018 blank votes", ylab="2019 blank votes")
```

```
plot(train$pre.nullVotes, train$nullVotes, xlab="2018 null votes", ylab="2019 null votes")
```

```
plot(train$pre.totalVoters, train$totalVoters, xlab="2018 total votes", ylab="2019 total votes")
```

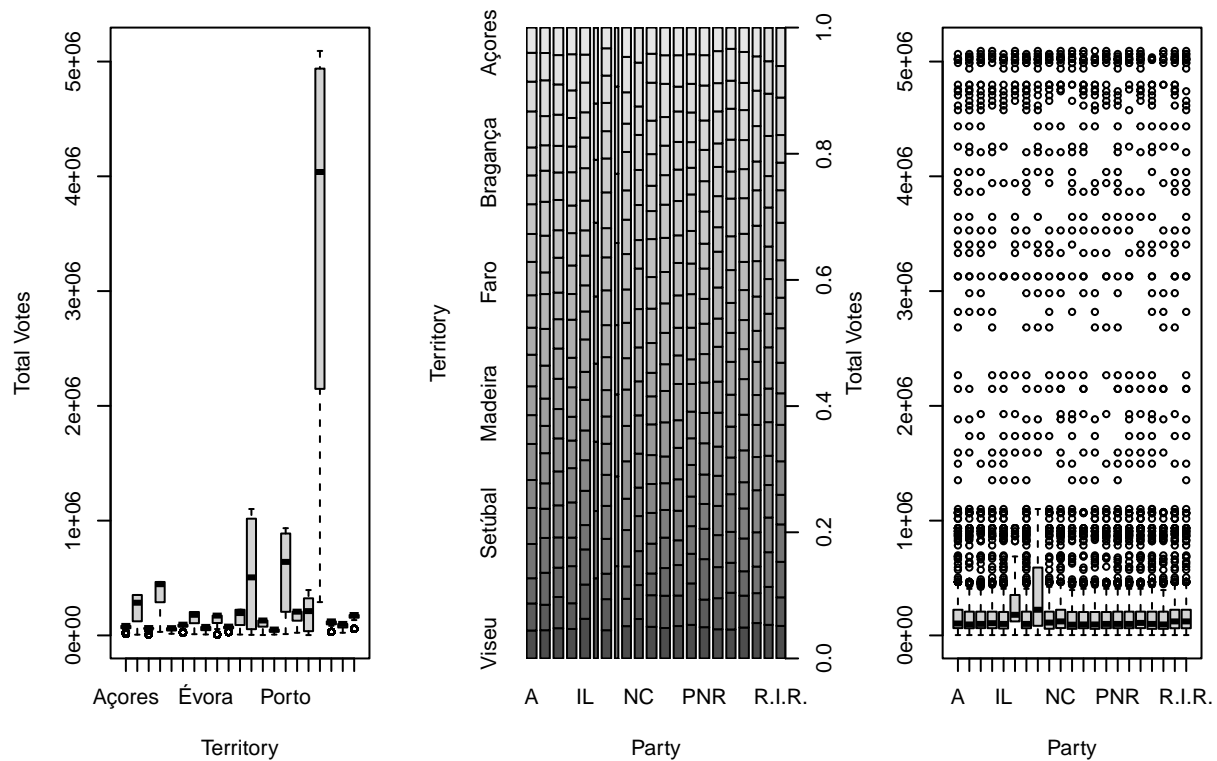


```
par(mfrow=c(1,3))
```

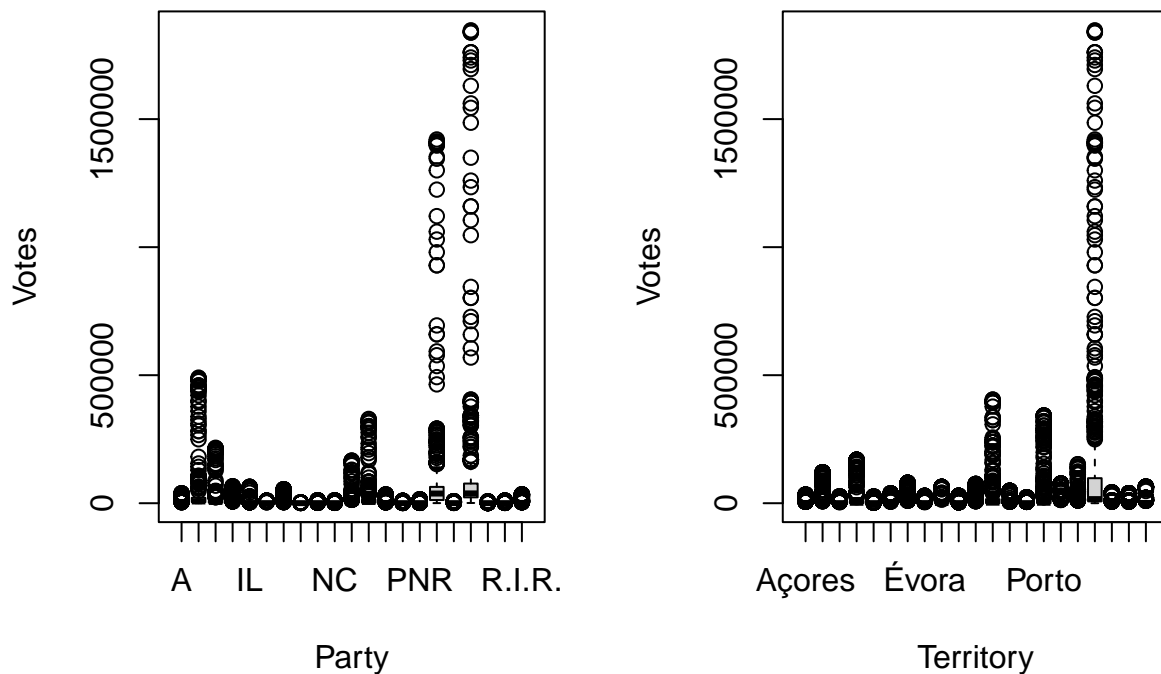
```
plot(train$territoryName, train$totalVoters, xlab="Territory", ylab="Total Votes")
```

```
plot(train$Party, train$territoryName, xlab="Party", ylab="Territory")
```

```
plot(train$Party, train$totalVoters, xlab="Party", ylab="Total Votes")
```



```
par(mfrow=c(1,2))
plot(train$Party, train$Votes, xlab="Party", ylab = "Votes")
plot(train$territoryName, train$Votes, xlab="Territory", ylab = "Votes")
```



```
partyVotes <- aggregate(x = train$totalVoters,
                        by = list(train$Party),
                        FUN = sum)
# Specify data column
# Specify group indicator
```

```
partyVotes
```

```
##      Group.1      x
## 1      A 241449578
## 2     B.E. 237656392
## 3    CDS-PP 228479379
## 4      CH 202382439
## 5      IL 210221858
## 6     JPP 167513031
## 7      L 235653448
## 8     MAS 184131415
## 9     MPT 216096413
## 10     NC 211777901
## 11     PAN 210435385
## 12    PCP-PEV 221491568
## 13 PCTP/MRPP 198621823
## 14     PDR 197636425
## 15     PNR 231616150
## 16    PPD/PSD 214152954
## 17     PPM 218199857
## 18     PS 224167849
## 19     PTP 164704622
## 20     PURP 221703091
## 21    R.I.R. 240198080
```

```
sort(partyVotes$x)
```

```
## [1] 164704622 167513031 184131415 197636425 198621823 202382439 210221858
## [8] 210435385 211777901 214152954 216096413 218199857 221491568 221703091
## [15] 224167849 228479379 231616150 235653448 237656392 240198080 241449578
```

Regular Linear Regression

```
lm1 <- lm(TimeElapsed~. -Party -territoryName, data=train)
pred <- predict(lm1, newdata=test)
cor_lm1 <- cor(pred, test$TimeElapsed)
mse_lm1 <- mean((pred-test$TimeElapsed)^2)

summary(lm1)
```

```
##
## Call:
## lm(formula = TimeElapsed ~ . - Party - territoryName, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -158.215  -59.308   -1.486    55.914   133.744
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.616e+02  2.122e+01   7.618 2.76e-14 ***
## totalMandates    3.142e+00  2.477e-01  12.684 < 2e-16 ***
## availableMandates -4.435e-01  9.717e-02  -4.565 5.05e-06 ***
## numParishes     -7.179e-02  1.007e-02  -7.131 1.05e-12 ***
```



```
## numParishesApproved      1.958e-01  2.199e-02   8.905 < 2e-16 ***
## blankVotes               1.904e-02  2.803e-03   6.791 1.16e-11 ***
## nullVotes                -7.763e-03  2.766e-03  -2.807 0.00501 **
## subscribedVoters         9.943e-04  1.900e-04   5.233 1.69e-07 ***
## totalVoters              6.778e-04  1.541e-04   4.398 1.10e-05 ***
## pre.blankVotes           -5.101e-02  4.172e-03 -12.227 < 2e-16 ***
## pre.blankVotesPercentage  4.547e+01  2.060e+00  22.076 < 2e-16 ***
## pre.nullVotes            7.133e-02  3.981e-03  17.918 < 2e-16 ***
## pre.nullVotesPercentage -6.561e+01  3.294e+00 -19.916 < 2e-16 ***
## pre.votersPercentage     -9.434e-02  3.040e-01  -0.310 0.75629
## pre.subscribedVoters     -1.586e-03  2.164e-04  -7.331 2.43e-13 ***
## pre.totalVoters          -1.634e-04  2.057e-04  -0.794 0.42693
## Votes                    7.110e-06  7.024e-06   1.012 0.31141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 68.34 on 12968 degrees of freedom
## Multiple R-squared:  0.2297, Adjusted R-squared:  0.2287
## F-statistic: 241.6 on 16 and 12968 DF,  p-value: < 2.2e-16
```

```
cat("\nCor for linear regression is", cor_lm1, "\n")
```

```
##
```

```
## Cor for linear regression is 0.4729907
```

```
cat("MSE for linear regression is", mse_lm1, "\n")
```

```
## MSE for linear regression is 4696.85
```

SVM Regression with linear kernels

```
svm1 <- svm(TimeElapsed~. -Party -territoryName, data=train, kernel="linear", cost=1, scale=TRUE)
summary(svm1)
```

```
##
```

```
## Call:
```

```
## svm(formula = TimeElapsed ~ . - Party - territoryName, data = train,
##      kernel = "linear", cost = 1, scale = TRUE)
```

```
##
```

```
##
```

```
## Parameters:
```

```
##      SVM-Type:  eps-regression
```

```
##      SVM-Kernel: linear
```

```
##           cost:  1
```

```
##           gamma: 0.0625
```

```
##           epsilon: 0.1
```

```
##
```

```
##
```

```
## Number of Support Vectors: 11912
```

```
pred <- predict(svm1, newdata=test)
```

```
cor_svm1 <- cor(pred, test$TimeElapsed)
```

```
mse_svm1 <- mean((pred - test$TimeElapsed)^2)
```

```
cat("\nCost = 1\n")
```

```
##
## Cost = 1
cat("\nCor for SVM linear kernels is", cor_svm1, "\n")
```

```
##
## Cor for SVM linear kernels is 0.3840566
cat("MSE for SVM linear kernels is", mse_svm1, "\n\n")
```

```
## MSE for SVM linear kernels is 7249.985
```

Tune SVM for linear

```
tune_svm1 <- tune(svm, TimeElapsed~., data=vald, kernel="linear",
                 ranges=list(cost=c(0.001, 0.01, 0.1, 1, 5, 10, 100)))
summary(tune_svm1)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##   cost
##     1
##
## - best performance: 4162.836
##
## - Detailed performance results:
##   cost   error dispersion
## 1 1e-03 4966.960    346.6701
## 2 1e-02 4865.199    278.1731
## 3 1e-01 4355.461    360.7025
## 4 1e+00 4162.836    378.0949
## 5 5e+00 4250.058    430.8164
## 6 1e+01 4247.997    444.1664
## 7 1e+02 4381.242    522.9834
```

SVM Regression with polynomial kernels

```
svm1 <- svm(TimeElapsed~. -Party -territoryName, data=train, kernel="polynomial", cost=100, scale=TRUE)
summary(svm1)
```

```
##
## Call:
## svm(formula = TimeElapsed ~ . - Party - territoryName, data = train,
##     kernel = "polynomial", cost = 100, scale = TRUE)
##
##
## Parameters:
##   SVM-Type:  eps-regression
##   SVM-Kernel: polynomial
##     cost:   100
##   degree:    3
```

```
##      gamma: 0.0625
##      coef.0: 0
##      epsilon: 0.1
##
##
## Number of Support Vectors: 10064

pred <- predict(svm1, newdata=test)
cor_svm1 <- cor(pred, test$TimeElapsed)
mse_svm1 <- mean((pred - test$TimeElapsed)^2)

cat("\nCost = 100\n")

##
## Cost = 100

cat("\nCor for SVM polynomial kernels is", cor_svm1, "\n")

##
## Cor for SVM polynomial kernels is 0.8413185

cat("MSE for SVM polynomial kernels is", mse_svm1, "\n\n")

## MSE for SVM polynomial kernels is 1827.722
```

Tune for SVM Polynomial

```
tune_svm1 <- tune(svm, TimeElapsed~., data=vald, kernel="polynomial",
                 ranges=list(cost=c(0.001, 0.01, 0.1, 1, 5, 10, 100)))
summary(tune_svm1)

##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##   cost
##   100
##
## - best performance: 2314.628
##
## - Detailed performance results:
##   cost      error dispersion
## 1 1e-03 5767.875   161.7382
## 2 1e-02 5673.955   176.0275
## 3 1e-01 5577.235   187.7613
## 4 1e+00 5113.061   218.6836
## 5 5e+00 4380.746   255.9669
## 6 1e+01 4000.048   255.7050
## 7 1e+02 2314.628   192.8751
```

SVM Regression with radial kernels

```
svm1 <- svm(TimeElapsed~. -Party -territoryName, data=train, kernel="radial", cost=100, gamma = 0.5, scale=1)
summary(svm1)
```

```
##
## Call:
## svm(formula = TimeElapsed ~ . - Party - territoryName, data = train,
##      kernel = "radial", cost = 100, gamma = 0.5, scale = TRUE)
##
##
## Parameters:
##      SVM-Type:  eps-regression
##      SVM-Kernel: radial
##           cost:  100
##           gamma: 0.5
##      epsilon:  0.1
##
##
## Number of Support Vectors: 7225
pred <- predict(svm1, newdata=test)
cor_svm1 <- cor(pred, test$TimeElapsed)
mse_svm1 <- mean((pred - test$TimeElapsed)^2)

cat("\nCost = 100, gamma = 0.5\n")

##
## Cost = 100, gamma = 0.5
cat("\nCor for SVM radial kernels is", cor_svm1, "\n")

##
## Cor for SVM radial kernels is 0.9236012
cat("MSE for SVM radial kernels is", mse_svm1, "\n\n")

## MSE for SVM radial kernels is 887.0489
```

Tune

```
set.seed(6229)
tune.out <- tune(svm, TimeElapsed~. -Party - territoryName, data=valid, kernel="radial",
                ranges=list(cost=c(0.1,1,10,100,1000),
                           gamma=c(0.5,1,2,3,4)))
summary(tune.out)

##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##      cost gamma
##      100   0.5
##
## - best performance: 942.3251
##
## - Detailed performance results:
##      cost gamma      error dispersion
## 1  1e-01   0.5 1623.3262 114.18671
```

## 2	1e+00	0.5	1054.8274	63.34937
## 3	1e+01	0.5	954.2966	62.53321
## 4	1e+02	0.5	942.3251	67.80555
## 5	1e+03	0.5	972.5289	75.92220
## 6	1e-01	1.0	1453.8438	93.84430
## 7	1e+00	1.0	1001.9913	65.97580
## 8	1e+01	1.0	946.7260	69.67114
## 9	1e+02	1.0	946.2801	72.24012
## 10	1e+03	1.0	977.3096	73.57489
## 11	1e-01	2.0	1374.7241	84.83065
## 12	1e+00	2.0	984.2186	69.31926
## 13	1e+01	2.0	952.0866	75.91539
## 14	1e+02	2.0	960.2135	78.43904
## 15	1e+03	2.0	1005.3989	113.65541
## 16	1e-01	3.0	1345.6399	83.38485
## 17	1e+00	3.0	982.1561	74.67856
## 18	1e+01	3.0	963.4774	84.86017
## 19	1e+02	3.0	972.2602	83.72258
## 20	1e+03	3.0	995.3337	75.73564
## 21	1e-01	4.0	1341.1773	79.07939
## 22	1e+00	4.0	985.5521	76.41773
## 23	1e+01	4.0	972.0528	88.36131
## 24	1e+02	4.0	981.8230	83.83309
## 25	1e+03	4.0	1006.0832	80.17504

Analysis of the results

Linear regression Regular linear regression for the TimeElapsed during the election preformed poorly having a low correlation of 0.47 and a high mse of over 4000.

SVM Linear Kernel SVM using a linear kernel with the best tuned cost preformed much worse than regular linear regression with a correlation of 0.38 and a mse of over 7000.

SVM Polynomial Kernel SVM using a polynomial kernel with the best tuned cost had a much higher correlation of 0.84 with a relatively lower mse of 1827.

SVM Radial Kernel SVM using a radial kernel preformed the best with a high correlation of 0.92 and a lower mse of 887.

Conclusion Of the three SVM kernels types polynomial and radial had a higher performance than regular linear regression for the TimeElapsed during the election, while the linear kernel had a worse performance when compared to regular linear regression. The data for the elections does not appear to be linear which explains why the SVM linear kernel and regular linear regression preformed poorly on the data set. Polynomial kernel and radial kernel had better results because they work well with non-linear data sets, with radial being slightly better at non-linear data sets.