SupervisedML_Knn_and_Naive_Bayes

Szuyu,Kuo

2020/11/4

K-Nearest Neighbors:

1.read the file

```
cereal <- read.csv("Cereals(1).csv")
```

2.

a. Rating is numeric.

```
str(cereal$rating)

##  num [1:77] 68.4 34 59.4 93.7 34.4 ...
```

b.Convert rating to a category, lable the cereal with "Approved" with at or above median, and "Unapproved" with below median.

```
cereal$rating <-  cut(cereal$rating,breaks=c(-Inf,median(cereal$rating)
,Inf), labels=c("Unapproved","Approved"))
```

c. Make sure that rating is a factor now.

```
str(cereal$rating)

##  Factor w/ 2 levels "Unapproved","Approved": 2 1 2 2 1 1 1 1 2 2 ...
```

3.Replace NAs with the median value

```
which(is.na(cereal))

## [1] 674 751 775 791

colSums(is.na(cereal))

##     name      mfr     type calories  protein      fat   sodium    fi
ber
##        0        0        0        0        0        0        0
  0
##    carbo   sugars   potass vitamins    shelf   weight     cups   rat
ing
##        1        1        2        0        0        0        0
  0

cereal$sugars[is.na(cereal$sugars)] <- median(cereal$sugars, na.rm = TR
UE)
cereal$carbo[is.na(cereal$carbo)] <- median(cereal$carbo, na.rm = TRUE)
cereal$potass[is.na(cereal$potass)] <- median(cereal$potass, na.rm = TR
UE)
```

4.seed value (180), training (60%) and validation (40%) sets.

```
set.seed(180)
newcereal <- sample_n(cereal,nrow(cereal))

N <- nrow(cereal)*0.6
N2 <- nrow(cereal)
train <- slice(newcereal, 1:N)
valid <- slice(newcereal, N:N2)
```

5.Make a new cereal.

a.

The new cereal name “Super cereal.”

b.using runif to generate the new cereal variable.

```
supercereal <-data.frame(1)
for (i in c(4:15)) {
  supercereal[i-3]<-data.frame(i= runif(1, min(train[,i]), max(train[,i
])))
  }
name<- colnames(cereal[4:15])
colnames(supercereal)<- c(name)

supercereal

##   calories  protein      fat   sodium    fiber    carbo  sugars   po
tass
## 1  102.677 5.413279 2.872957 131.4084 1.790669 17.42395 2.53546 314.
4097
##   vitamins    shelf    weight     cups
## 1 83.92606 1.315833 0.8423837 1.024422
```

6. Normalize data (Normalize the data to get the distance)

```
train_norm<-train
valid_norm<-valid
normalize <- preProcess(train[,4:15],method=c("center", "scale"))

train_norm[,4:15] <-predict(normalize,train[,4:15])
valid_norm[,4:15] <-predict(normalize,valid[,4:15])
supercereal_norm <- predict(normalize,supercereal)
```

7.Using the knn() function and predicted the classification for supercereal.

```
nn <- knn(train = train_norm[, 4:15], test = supercereal_norm, cl = tra
in_norm[, 16], k = 7) #k =7 meaning the 7 nearest nighbors
r <- row.names(train)[attr(nn, "nn.index")]
nn[1]

## [1] Approved
## Levels: Approved
```

```
nearest7 <-train[c(r),]
nearest7
```

```
##                                 name mfr type calories protein fat s
odium
## 27                  Total_Whole_Grain   G    C      100       3   1
  200
## 36                           Wheaties   G    C      100       3   1
  200
## 33                              Maypo   A    H      100       4   1
    0
## 38 Muesli_Raisins,_Peaches,_&_Pecans   R    C      150       4   3
  150
## 29  Muesli_Raisins,_Dates,_&_Almonds   R    C      150       4   3
   95
## 26                          Special_K   K    C      110       6   0
  230
## 12                  Grape_Nuts_Flakes   P    C      100       3   1
  140
##    fiber carbo sugars potass vitamins shelf weight cups     rating
## 27     3    16      3    110      100     3      1 1.00   Approved
## 36     3    17      3    110       25     1      1 1.00   Approved
## 33     0    16      3     95       25     2      1 1.00   Approved
## 38     3    16     11    170       25     3      1 1.00 Unapproved
## 29     3    16     11    170       25     3      1 1.00 Unapproved
## 26     1    16      3     55       25     1      1 1.00   Approved
## 12     3    15      5     85       25     3      1 0.88   Approved
```

The outcome category of *supercereal* is predicted to be “Approved”, and the nearest 7 neighbors are showing above. "Unapproved" and "Approved" are 2 and 5.

7a. Use validation set to determine an optimal k-value.

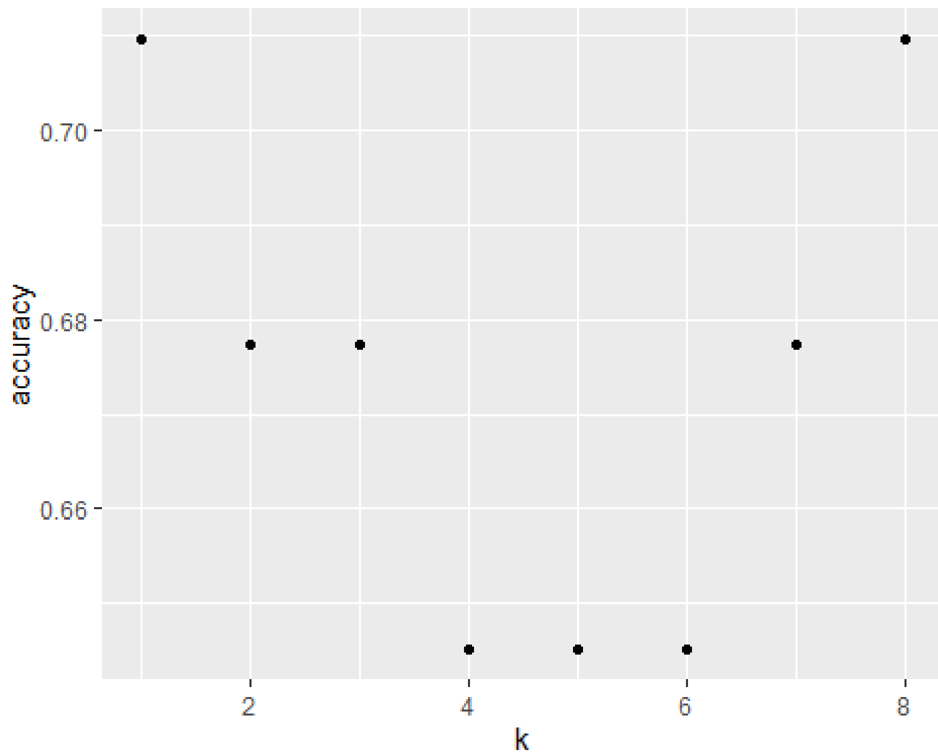
```
accuracy.df <- data.frame(k = seq(1, 8, 1), accuracy = rep(0, 8))
for(i in 1:8) {
  knn.pred <- knn(train_norm[, 4:15], valid_norm[, 4:15], cl = train_no
rm[, 16], k = i)
  accuracy.df[i, 2] <- confusionMatrix(knn.pred, valid_norm[, 16])$over
all[1]
}


accuracy.df
```

```
##   k  accuracy
## 1 1 0.7096774
## 2 2 0.6774194
## 3 3 0.6774194
## 4 4 0.6451613
## 5 5 0.6451613
## 6 6 0.6451613
```

```
## 7 7 0.6774194
## 8 8 0.7096774
```

7b. scatterplot with the various k values

```
ggplot(accuracy.df, aes(k,accuracy))+geom_point()
```



8. Re-run knn() function with the new k-value.

```
knn.pred.new <- knn(train_norm[, 4:15], supercereal_norm, cl = train_no
rm[, 16], k = 8)
r2 <- row.names(train)[attr(knn.pred.new, "nn.index")]
knn.pred.new

## [1] Approved
## attr(,"nn.index")
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
## [1,]   27   36   33   38   29   26   12   41
## attr(,"nn.dist")
##          [,1]     [,2]     [,3]     [,4]     [,5]     [,6]     [,7]
## [1,] 4.646788 4.987376 5.016792 5.232463 5.244299 5.462923 5.536603
##          [,8]
## [1,] 5.592264
## Levels: Approved
```

The result is not different from the result with the k-value of 7. Also, the outcome class for the 8 nearest neighbor were most approved.

```
nearest <-train[c(r2),]
c(nearest[1],nearest[16])

## $name
## [1] Total_Whole_Grain                  Wheaties

## [3] Maypo                              Muesli_Raisins,_Peaches,_&_Pec
ans
## [5] Muesli_Raisins,_Dates,_&_Almonds  Special_K

## [7] Grape_Nuts_Flakes                  Quaker_Oatmeal

## 77 Levels: 100%_Bran 100%_Natural_Bran ... Wheaties_Honey_Gold
##
## $rating
## [1] Approved   Approved   Approved   Unapproved Unapproved Approved

## [7] Approved   Approved
## Levels: Unapproved Approved
```

Naive Bayes:

1.

```
Con_V <- read.csv("congressional_votes(1).csv")
head(Con_V)

##   handicapped.infants water.project.cost.sharing
## 1                   n                          y
## 2                   n                          y
## 3                   ?                          y
## 4                   n                          y
## 5                   y                          y
## 6                   n                          y
##   adoption.of.the.budget.resolution physician.fee.freeze el.salvador
.aid
## 1                                 n                    y
   y
## 2                                 n                    y
   y
## 3                                 y                    ?
   y
## 4                                 y                    n
   ?
## 5                                 y                    n
   y
## 6                                 y                    n
   y
##   religious.groups.in.schools anti.satellite.test.ban
## 1                           y                       n
## 2                           y                       n
```

```
## 3                           y                        n
## 4                           y                        n
## 5                           y                        n
## 6                           y                        n
##   aid.to.nicaraguan.contras mx.missile immigration
## 1                         n          n           y
## 2                         n          n           n
## 3                         n          n           n
## 4                         n          n           n
## 5                         n          n           n
## 6                         n          n           n
##   synfuels.corporation.cutback education.spending superfund.right.to
.sue
## 1                            ?                  y
   y
## 2                            n                  y
   y
## 3                            y                  n
   y
## 4                            y                  n
   y
## 5                            y                  ?
   y
## 6                            n                  n
   y
##   crime duty.free.exports export.administration.act.south.africa
## 1     y                 n                                      y
## 2     y                 n                                      ?
## 3     y                 n                                      n
## 4     n                 n                                      y
## 5     y                 y                                      y
## 6     y                 y                                      y
##   Political_Party
## 1      republican
## 2      republican
## 3        democrat
## 4        democrat
## 5        democrat
## 6        democrat
```

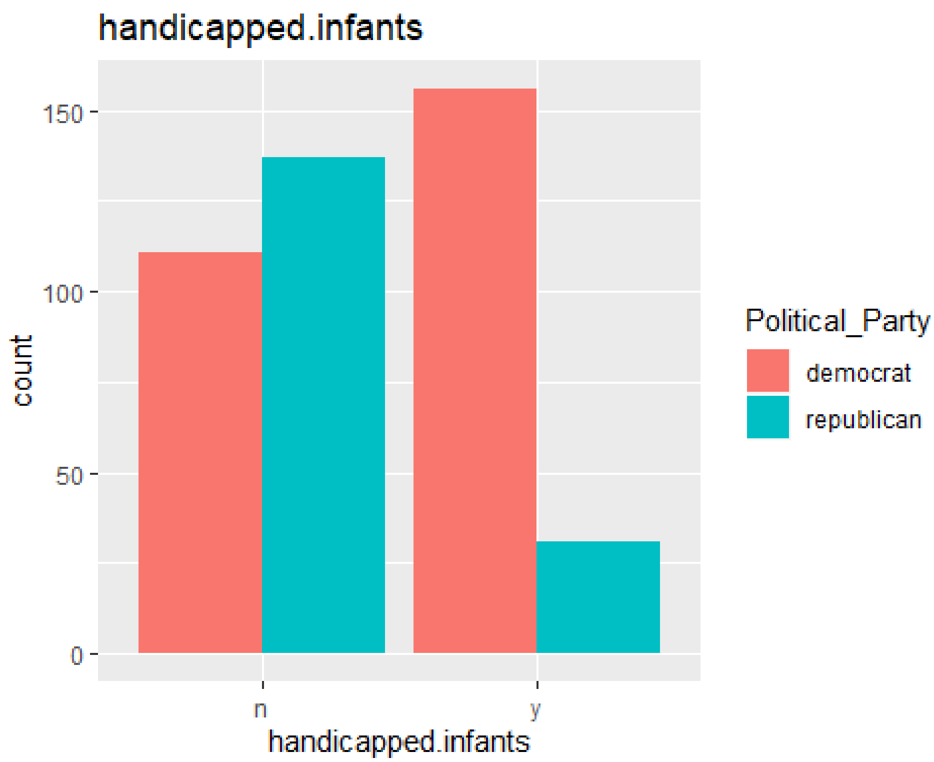
2.Change “?” to “n”

```
for (i in 1:nrow(Con_V)) {
  for (a in 1:ncol(Con_V)) {
    if (Con_V[i,a]=='?')
      Con_V[i,a]<-"n"
  }
}

summary(Con_V) #check it
```

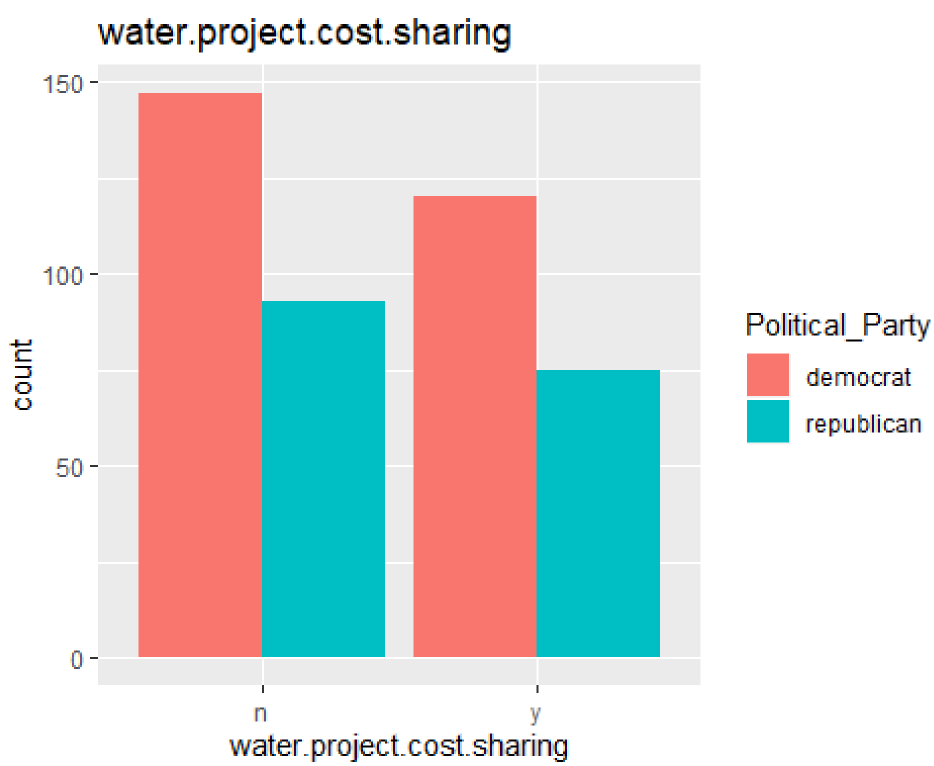
```
##  handicapped.infants water.project.cost.sharing
##  ?:  0               ?:  0
##  n:248               n:240
##  y:187               y:195
##  adoption.of.the.budget.resolution physician.fee.freeze el.salvador.
aid
##  ?:  0                             ?:  0                ?:  0

##  n:182                             n:258                n:223

##  y:253                             y:177                y:212

##  religious.groups.in.schools anti.satellite.test.ban
##  ?:  0                       ?:  0
##  n:163                       n:196
##  y:272                       y:239
##  aid.to.nicaraguan.contras mx.missile immigration
##  ?:  0                     ?:  0      ?:  0
##  n:193                     n:228      n:219
##  y:242                     y:207      y:216
##  synfuels.corporation.cutback education.spending superfund.right.to.
sue
##  ?:  0                        ?:  0              ?:  0

##  n:285                        n:264              n:226

##  y:150                        y:171              y:209

##  crime   duty.free.exports export.administration.act.south.africa
##  ?:  0   ?:  0             ?:  0
##  n:187   n:261             n:166
##  y:248   y:174             y:269
##    Political_Party
##  democrat  :267
##  republican:168
##
```

3. Preparatory data analysis

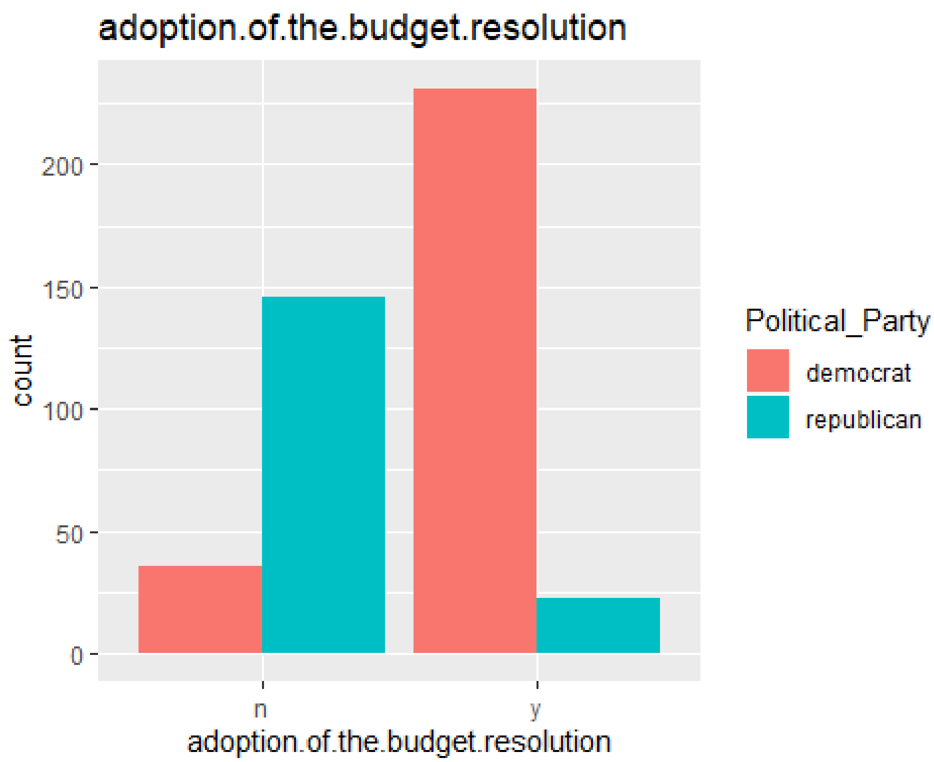
```
colNames <- names(Con_V)[1:16]

#using aes_string instead of aes for i to apply
for(n in colNames){
  p <- ggplot(Con_V,aes_string(x=n,fill= "Political_Party"))+ geom_bar(
position = position_dodge())+ggtitle(n)
  print(p)
  Sys.sleep(2) # rest for every 2 second
}
```

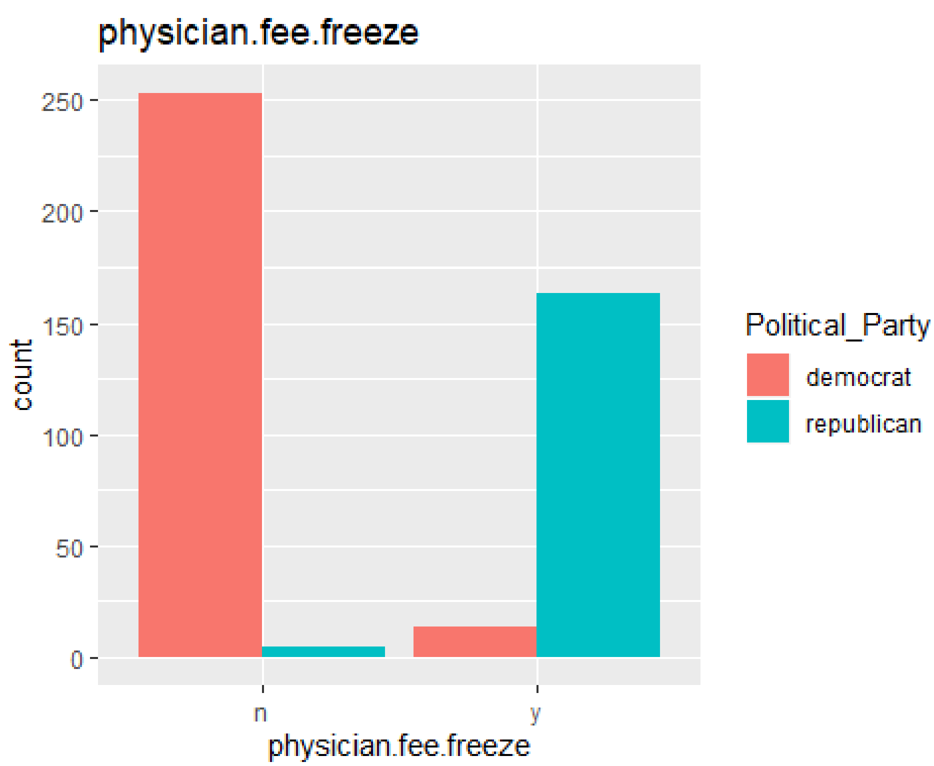


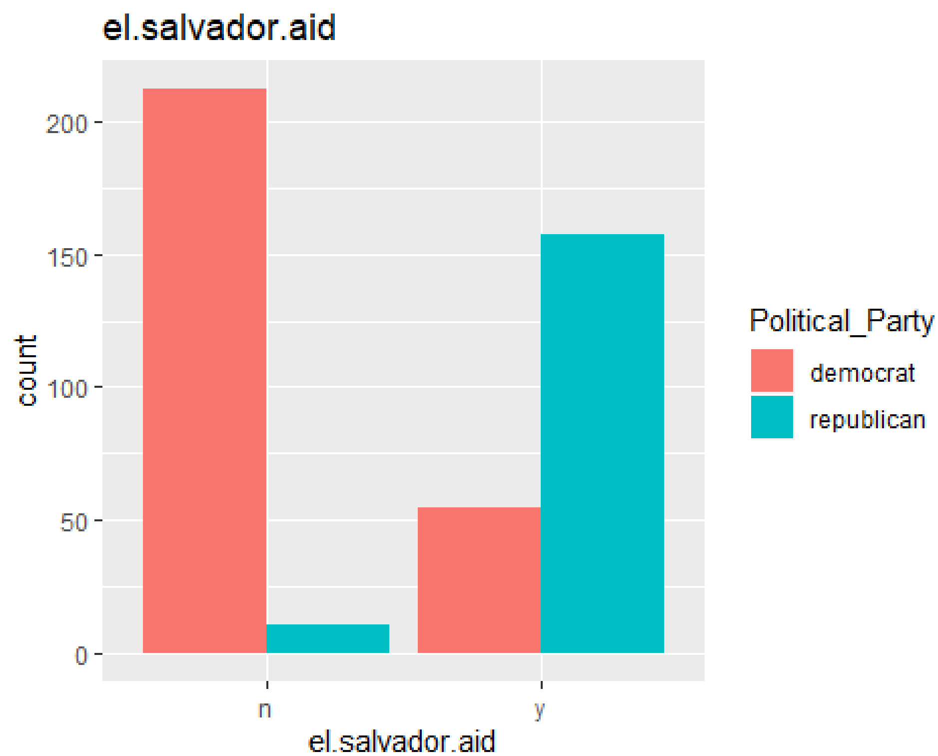
handicapped.infants
150
100
50
0
count
n
y
handicapped.infants
Political_Party
democrat
republican

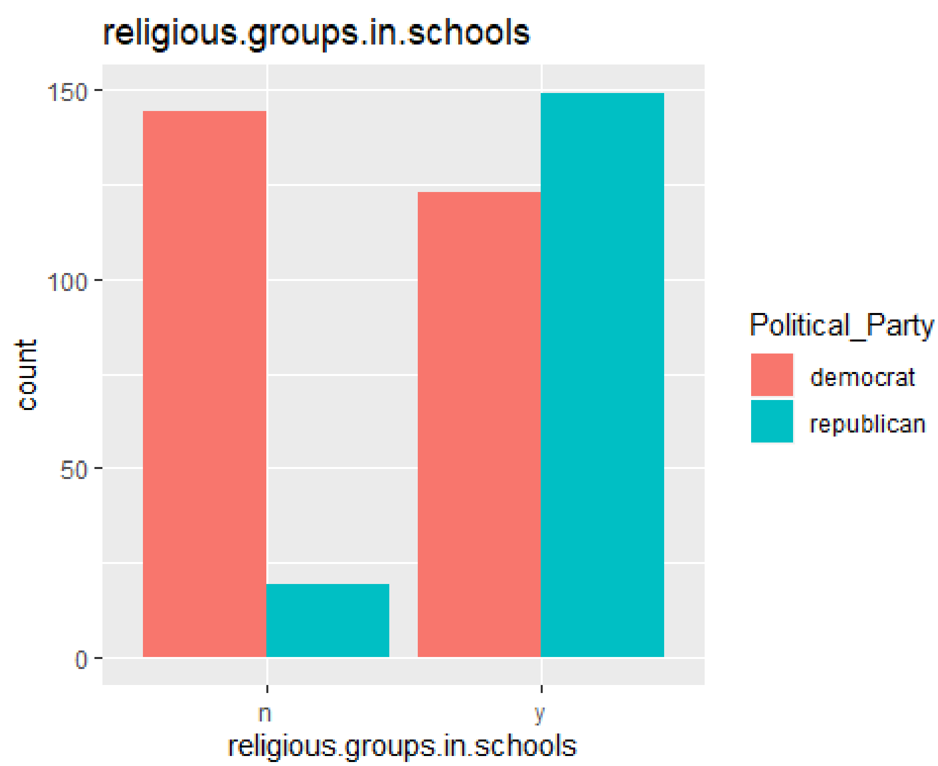


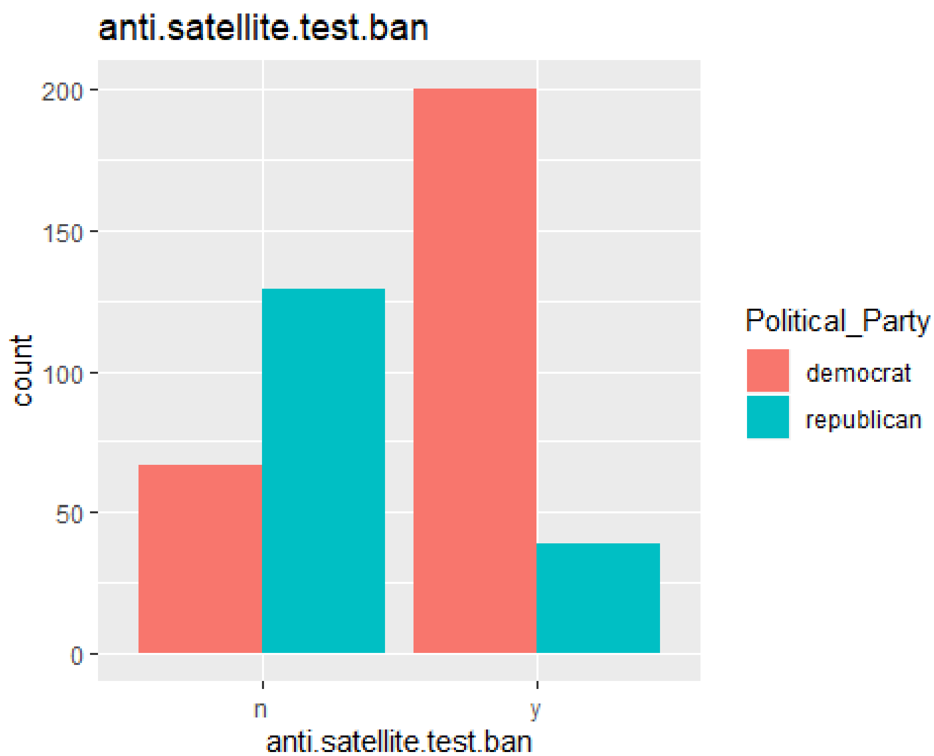
water.project.cost.sharing
150
100
50
0
count
n
y
water.project.cost.sharing
Political_Party
democrat
republican

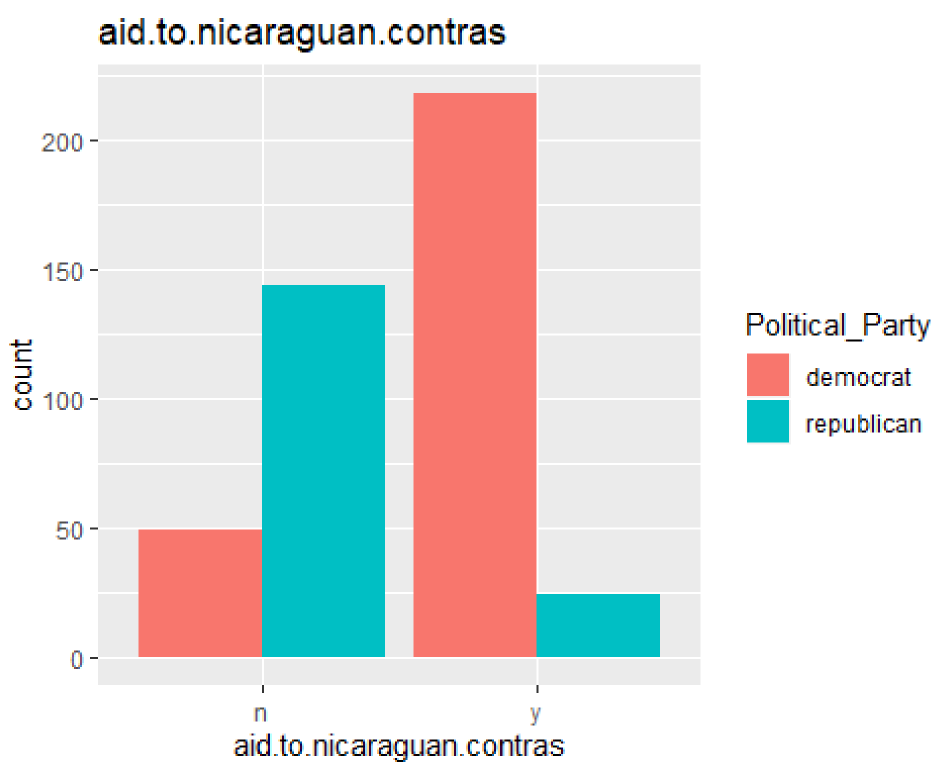

adoption.of.the.budget.resolution
count
200
150
100
50
0
n
y
adoption.of.the.budget.resolution
Political_Party
democrat
republican

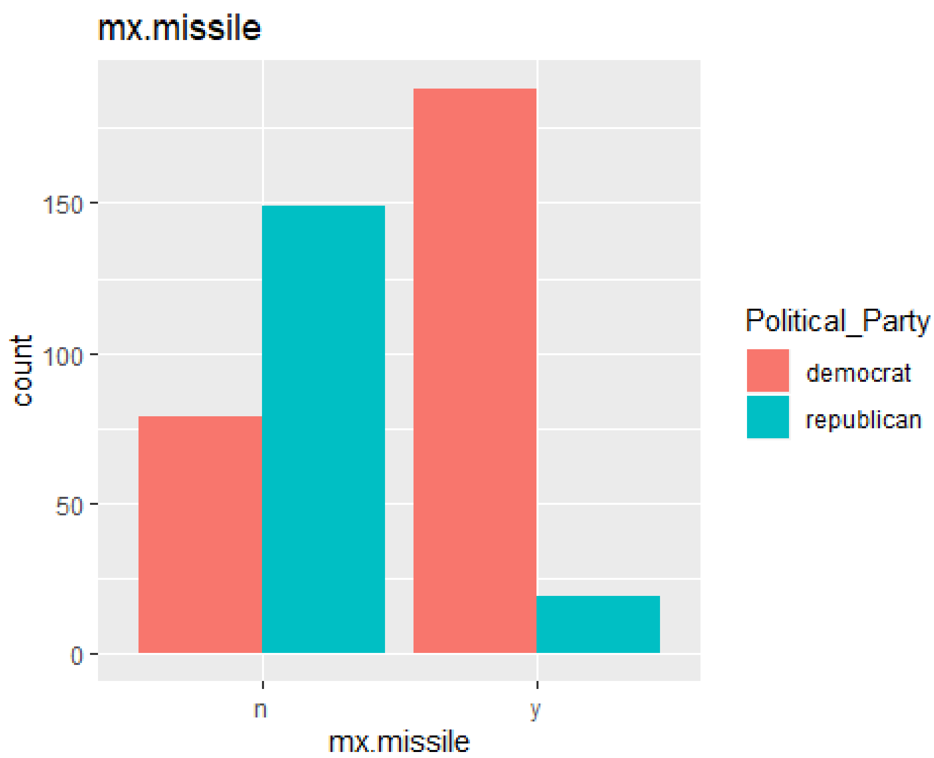

physician.fee.freeze
count
250
200
150
100
50
0
n
y
physician.fee.freeze
Political_Party
democrat
republican

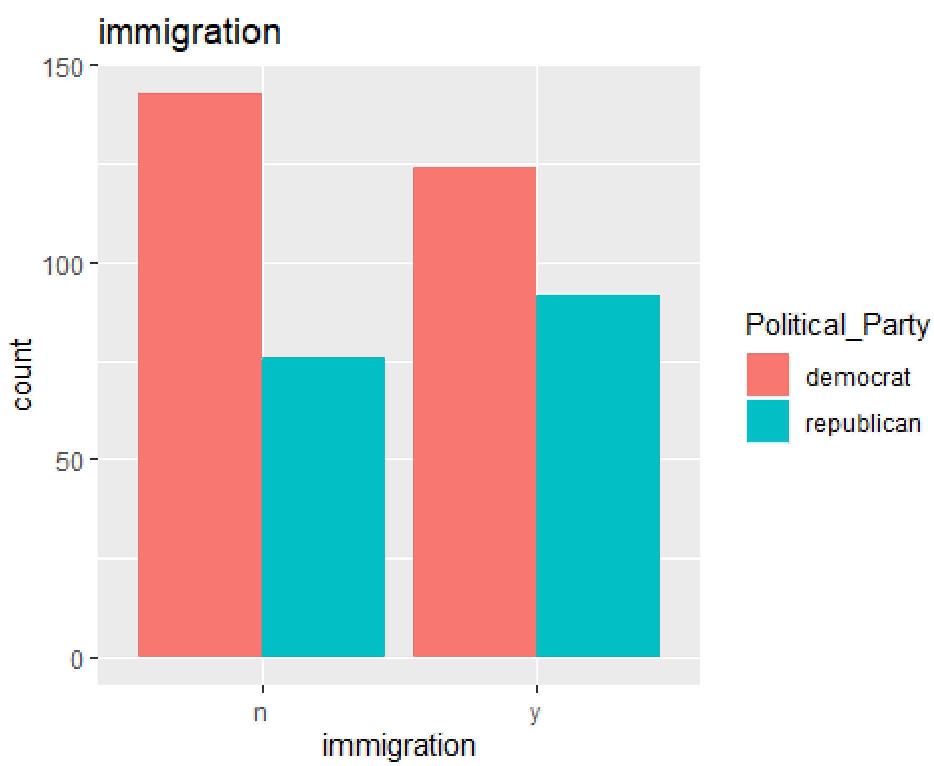

el.salvador.aid
count
200
150
100
50
0
n
y
el.salvador.aid
Political_Party
democrat
republican

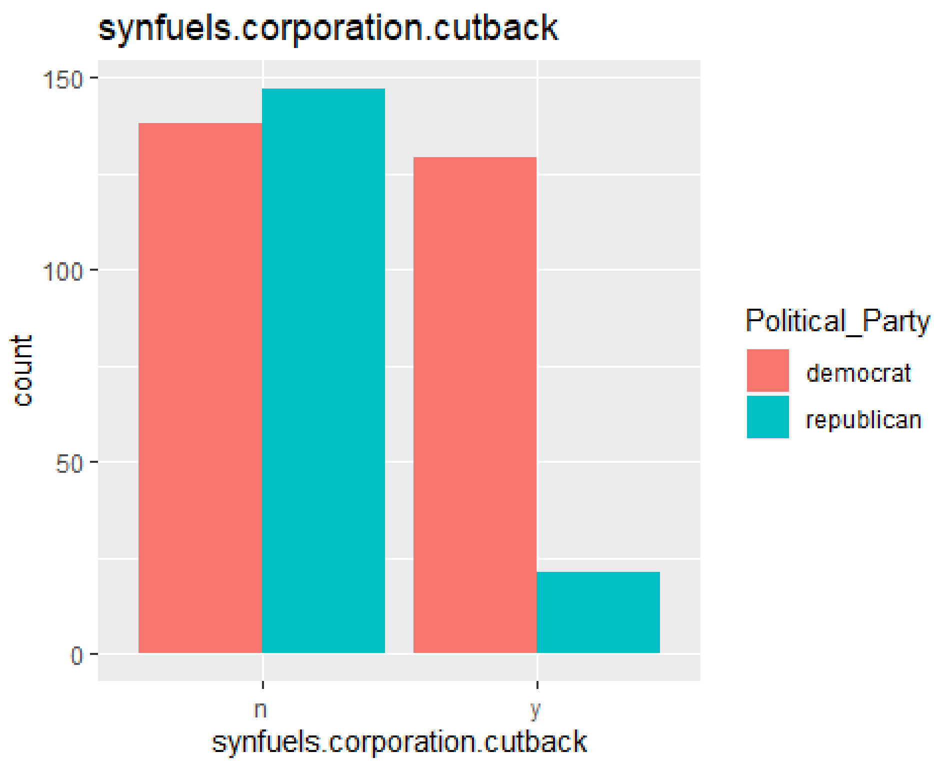

religious.groups.in.schools
count
150
100
50
0
n
y
religious.groups.in.schools
Political_Party
democrat
republican


anti.satellite.test.ban
count
200
150
100
50
0
n
y
anti.satellite.test.ban
Political_Party
democrat
republican

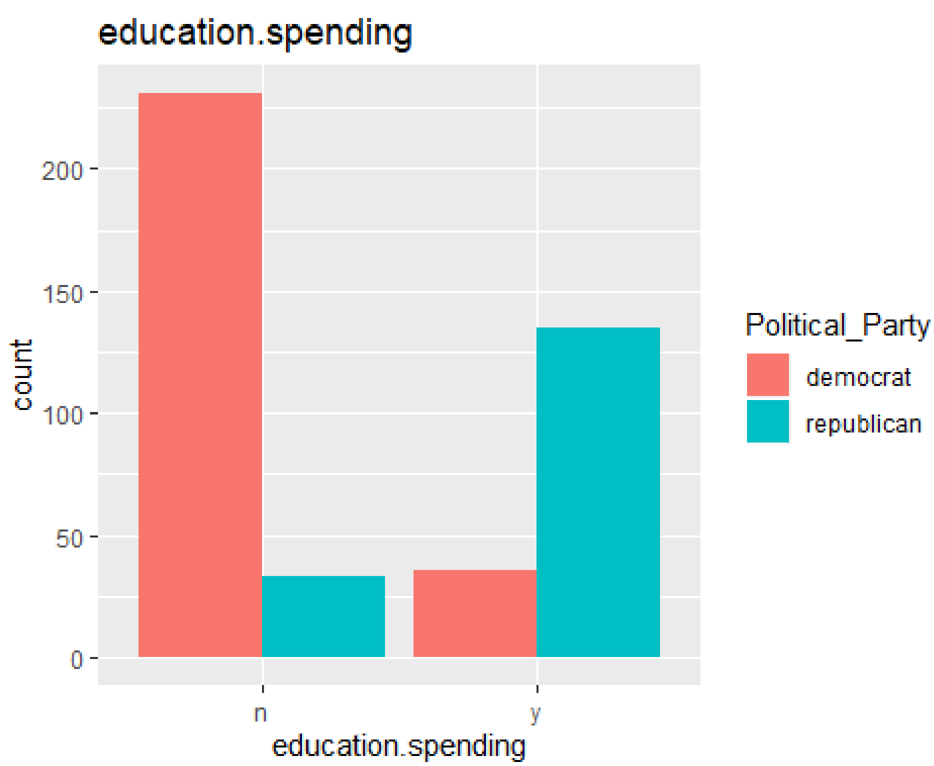

aid.to.nicaraguan.contras
count
200
150
100
50
0
n
y
aid.to.nicaraguan.contras
Political_Party
democrat
republican


mx.missile
count
150
100
50
0
n
y
mx.missile
Political_Party
democrat
republican

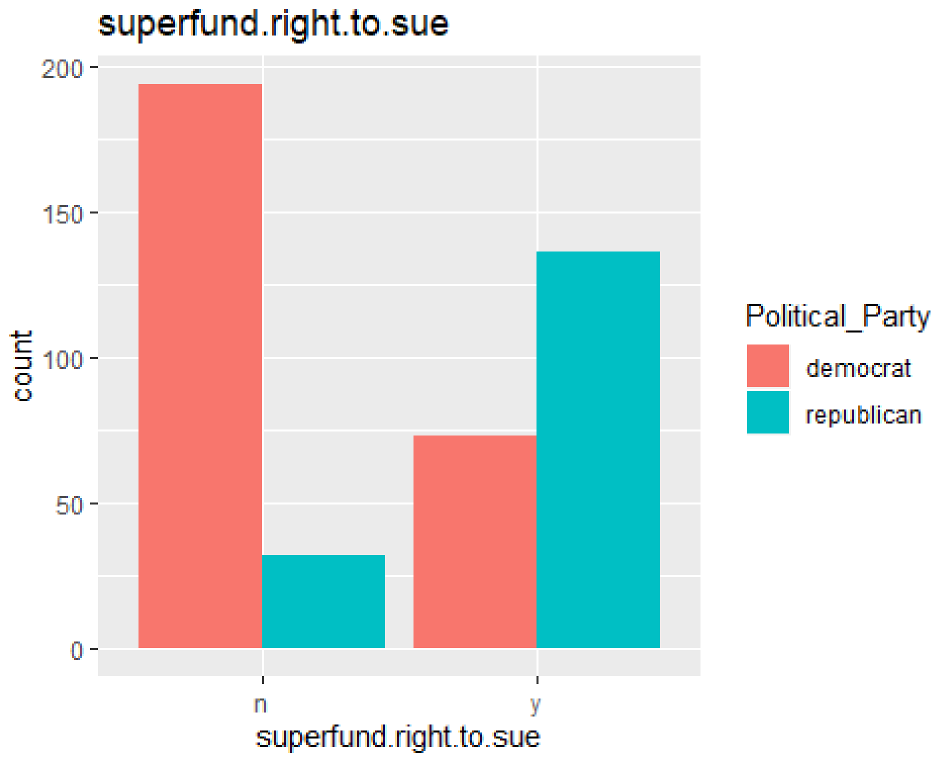

immigration
count
150
100
50
0
n
y
immigration
Political_Party
democrat
republican

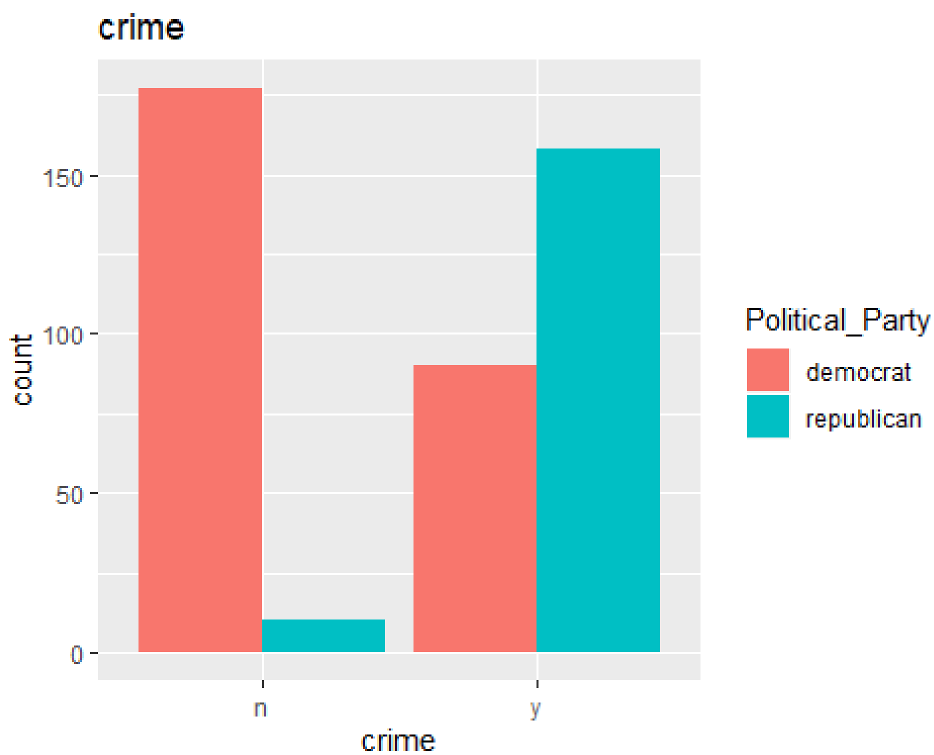


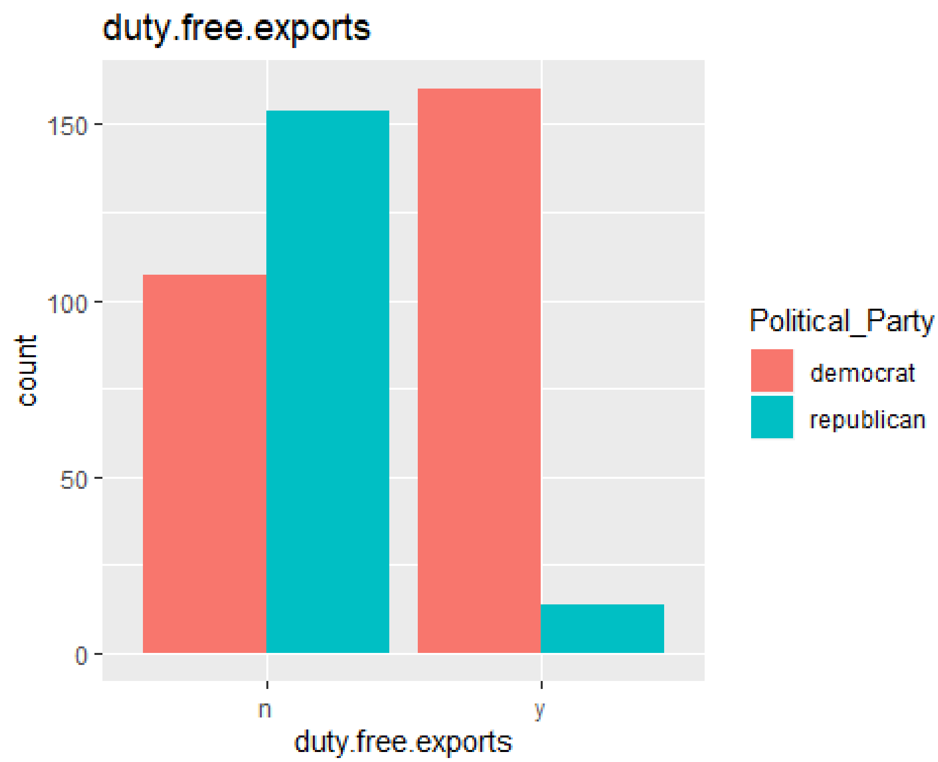
synfuels.corporation.cutback
150
100
50
0
count
n
y
synfuels.corporation.cutback
Political_Party
democrat
republican

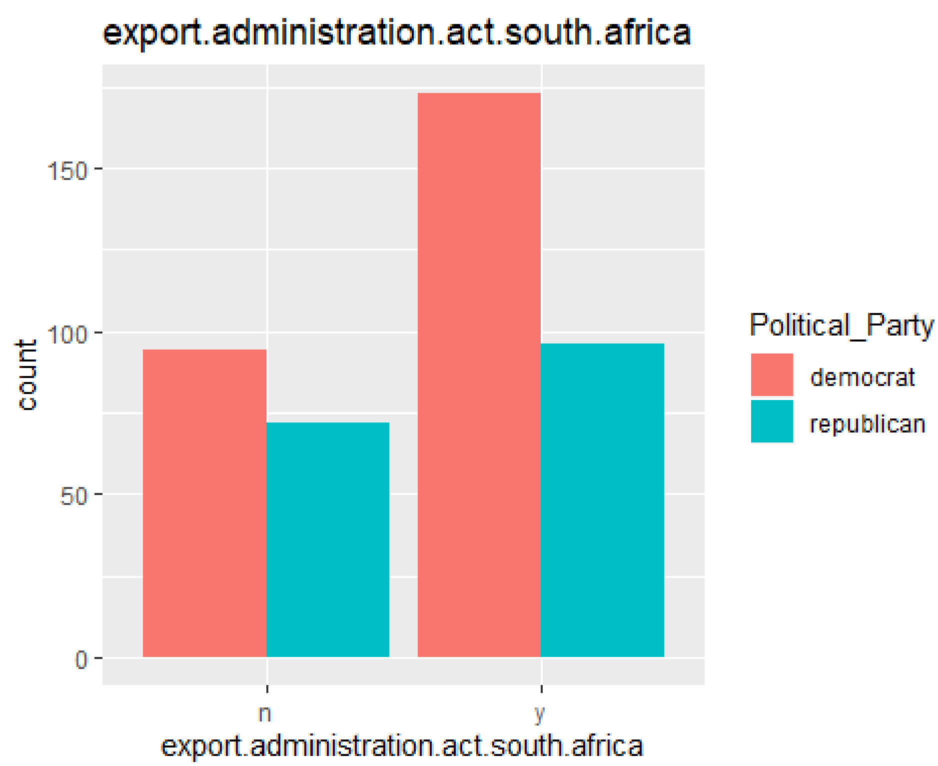


education.spending
200
150
100
50
0
count
n
y
education.spending
Political_Party
democrat
republican


superfund.right.to.sue
200
150
100
50
0
count
n
y
superfund.right.to.sue
Political_Party
democrat
republican


crime
150
100
50
0
count
n
y
crime
Political_Party
democrat
republican





Based on the barplots, Congresspersons usually vote based on thier party identification. Among 16 topics, only “Water Project Cost Sharing”, “Export Administration Act South Afica” and “Immigration” topics were voted close to even on both party.

4.seed value (180), training (60%) and validation (40%) sets.

```
set.seed(180)
newCon_V <- sample_n(Con_V,nrow(Con_V))

N <- nrow(Con_V)*0.6
N2 <- nrow(Con_V)
train.v <- slice(newCon_V, 1:N)
valid.v <- slice(newCon_V, N:N2)
```

5.Build a naive bayes model

```
nb.model <-naiveBayes(Political_Party ~ ., data = train.v)
nb.model

## 
## Naive Bayes Classifier for Discrete Predictors
## 
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
## 
## A-priori probabilities:
## Y
##   democrat republican 
##  0.6091954  0.3908046 
## 
## Conditional probabilities:
##             handicapped.infants
## Y                    ?         n         y
##   democrat   0.0000000 0.3773585 0.6226415
##   republican 0.0000000 0.8235294 0.1764706
## 
##             water.project.cost.sharing
## Y                    ?         n         y
##   democrat   0.0000000 0.5974843 0.4025157
##   republican 0.0000000 0.5588235 0.4411765
## 
##             adoption.of.the.budget.resolution
## Y                    ?         n         y
##   democrat   0.0000000 0.1069182 0.8930818
##   republican 0.0000000 0.8725490 0.1274510
## 
##             physician.fee.freeze
## Y                     ?          n          y
##   democrat   0.00000000 0.94968553 0.05031447
##   republican 0.00000000 0.01960784 0.98039216
## 
##             el.salvador.aid
## Y                     ?          n          y
##   democrat   0.00000000 0.82389937 0.17610063
##   republican 0.00000000 0.05882353 0.94117647
## 
```

```
##             religious.groups.in.schools
## Y                   ?         n         y
##   democrat   0.0000000 0.6037736 0.3962264
##   republican 0.0000000 0.1176471 0.8823529
##
##             anti.satellite.test.ban
## Y                   ?         n         y
##   democrat   0.0000000 0.2327044 0.7672956
##   republican 0.0000000 0.7941176 0.2058824
##
##             aid.to.nicaraguan.contras
## Y                   ?         n         y
##   democrat   0.0000000 0.1635220 0.8364780
##   republican 0.0000000 0.8627451 0.1372549
##
##             mx.missile
## Y                   ?         n         y
##   democrat   0.0000000 0.2578616 0.7421384
##   republican 0.0000000 0.8921569 0.1078431
##
##             immigration
## Y                   ?         n         y
##   democrat   0.0000000 0.5345912 0.4654088
##   republican 0.0000000 0.4215686 0.5784314
##
##             synfuels.corporation.cutback
## Y                   ?         n         y
##   democrat   0.0000000 0.5345912 0.4654088
##   republican 0.0000000 0.8725490 0.1274510
##
##             education.spending
## Y                   ?         n         y
##   democrat   0.0000000 0.8993711 0.1006289
##   republican 0.0000000 0.1960784 0.8039216
##
##             superfund.right.to.sue
## Y                   ?         n         y
##   democrat   0.0000000 0.7295597 0.2704403
##   republican 0.0000000 0.1862745 0.8137255
##
##             crime
## Y                    ?          n          y
##   democrat   0.00000000 0.71698113 0.28301887
##   republican 0.00000000 0.06862745 0.93137255
##
##             duty.free.exports
## Y                    ?          n          y
##   democrat   0.00000000 0.38364780 0.61635220
##   republican 0.00000000 0.92156863 0.07843137
##
```

```
##             export.administration.act.south.africa
## Y                     ?         n         y
##   democrat   0.0000000 0.3018868 0.6981132
##   republican 0.0000000 0.4607843 0.5392157
```

6.Confusion matrix compares the performance of model against the training data, and the validation data

```
pred.train.v<-predict(nb.model, newdata = train.v)
confusionMatrix(pred.train.v, train.v$Political_Party)$overall[1]

##  Accuracy
## 0.9118774

pred.valid.v<- predict(nb.model, newdata = valid.v)
confusionMatrix(pred.valid.v, valid.v$Political_Party)$overall[1]

## Accuracy
##     0.88
```

Both performance of Confusion Matrix show high accuracy (around 90%) of the prediction.

7.a new Congressperson

a.

SuperMan Congressperson

b.

```
SuperMan <- data.frame(1)
for(y in c(1:16)){
  SuperMan[y] <- data.frame(y = sample_n(Con_V[y], 1))
}

colnames(SuperMan)<- c(colNames)
SuperMan

##   handicapped.infants water.project.cost.sharing
## 1                   y                          y
##   adoption.of.the.budget.resolution physician.fee.freeze el.salvador
.aid
## 1                                 n                    y
   n
##   religious.groups.in.schools anti.satellite.test.ban
## 1                           n                       y
##   aid.to.nicaraguan.contras mx.missile immigration
## 1                         y          y           n
##   synfuels.corporation.cutback education.spending superfund.right.to
.sue
## 1                            y                  y
   n
```

```
##   crime duty.free.exports export.administration.act.south.africa
## 1     n                 n                                      n
```

c.SuperMan Congressperson is democrat party

```
predict(nb.model, newdata = SuperMan)

## [1] democrat
## Levels: democrat republican
```

d.

```
predict(nb.model, newdata = SuperMan, type = "raw")

##       democrat   republican
## [1,] 0.9995642 0.0004358338
```