Project Title

Comparison of Logistic Regression and SVM Method

Members

1. First member: name and e-mail adress Zeynep Göger and gogerz@itu.edu.tr

Contents

Description of the project

In this project, we compare two methods that are logistic regression method and SVM method by using 1984 United Stated Congressional Voting Records in R language. We analyse that which method is better to predict the congressman belongs to which party according to first column of the data to the rest of them.

The methods to be used

There are two different methods that are logistic regression method and SVM method in this project. Logistic regression is a type of a prediction method that is used if the data has binary dependent variables and this method determines the relationship between the set of dependent variables and the independent set.

Support Vector Machines method is another prediction method that is used if the data is binary too. However, in this method two kinds of accumulation of dependent variables are determined and divided down the middle of the distance that is called as margin.

The data

The data is 1984 United Stated Congressional Voting Records that has 435 rows and 17 columns. This data consists of 435 congressmen that are 267 democrats and 168 republicans. In this data has binary dependent variables such as yes or no and two class of congressmen that are republican and democrat. There are 16 numbers of attributes that are categorical.

Code

As proof of work, you must run this notebook. Upload an HTML output of this notebook on your github account.

In [64]:

```
X<- read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/voting-records/house-votes-84.data", header=FALSE)
```

In [65]:

```
head(X)
```

V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
republican	n	y	n	y	y	y	n	n	n	y	?	y	y	y	n	y
republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	?
democrat	?	y	y	?	y	y	n	n	n	n	y	n	y	y	n	n
democrat	n	y	y	n	?	y	n	n	n	n	y	n	y	n	n	y
democrat	y	y	y	n	y	y	n	n	n	n	y	?	y	y	y	y
democrat	n	y	y	n	y	y	n	n	n	n	n	n	y	y	y	y

In [ ]:

```

```

1.Logistic Regression

a.Setting up Logistic Regression Model

Logistic regression is a type of a prediction method that is used if the data has binary dependent variables and this method determines the relationship between the set of dependent variables and the independent set. Odds is the ratio of the probability of occurrence with probability of event that does not occur in this formula stated in [1]. \begin{equation} odds= p/(1-p)\qquad\qquad\qquad \qquad\qquad\qquad (1.1)\end{equation}

There is link between the Binomial distribution and the linear combination of independent variables and this function is called logit function presented in [1] .
\begin{equation}logit p = log(odds) = log(p/(1-p)) = \beta_0 + \beta_1x_1 +...+ \beta_k x_k \qquad\qquad\qquad(1.2)\end{equation}

This formula resembles a linear regression but logistic regression gets a best fitting by using maximum likelihood function that maximizes the probability.
\begin{equation}p=\frac{e^{\beta_0 + \beta_1x_1 +...+ \beta_k x_k}}{1+e^{\beta_0 + \beta_1x_1 +...+ \beta_k x_k}}\qquad \qquad\qquad\qquad \qquad\qquad\qquad (1.3)\end{equation}

where β_0 is constant,$\beta_1,...,\beta_k$ are the coeffients of predictor variables, e is a natural logarithm and p is the probability of a specific case.

Also we can write this equation which is emphasized in [1] as the following \begin{equation} e(logit p)= e^{(\beta_0) + (\beta_1x_1) +...+ (\beta_k x_k)} \qquad\qquad \qquad\qquad\qquad \qquad (1.4)\end{equation} then \begin{equation}o= p/(1-p)=e^{(\beta_0)}e^{(\beta_1x_1)}e^{(\beta_2x_2)}....e^{(\beta_kx_k)} \qquad\qquad\qquad \qquad (1.5) \end{equation}

where β_k's are constant coefficients that determines the changes of logistic regression model when x_i is added. Also, adding or subtracting a unit alters the odds with constant amount.

b.The Main Goals of Logistictic Regression Model

There are two primary purposes of logistic regression models. First one of these purposes is to predict of group membership.This model uses the odds ratio;therefore, the conclusion of using this model has the the same form with the odds ratio. The second aim is suplying informations about connections and strongness between the variables.

c.Setting up a Model in R

Initially, we divide the data to constitute our model and we divide 30% of the data randomly. To do this we use this code.

In [66]:

```
sample(1:nrow(X), 0.3*nrow(X))
```

```
test <-sample(1:nrow(X), 0.3*nrow(X))
```

143 101 134 121 6 265 71 261 393 25 345 36 105 139 396 119 408 55 273 155 197 22 107 234 280 59 216 162 4 188 117 62 214 38 238 252 2 249 112 343 200 269 405 399 21 315 177 433 317 368 171 88 1 394 272 354 218 226 282 284 360 406 160 84 87 192 231 8 148 243 232 66 195 213 295 321 259 189 248 411 424 426 95 335 228 291 389 74 208 358 237 353 147 222 417 283 142 99 130 296 185 247 274 352 370 80 179 97 69 198 348 287 138 34 279 377 290 30 194 310 133 68 423 255 90 425 210 235 168 165

In [67]:

```
length(test)
```

130

130 members of this data is selected randomly by using this code in R but these numbers are not useful for testing and forming our model. Therefore, we convert them to a matrix by using the code below.

In [68]:

```
X[test,]
```

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
214	democrat	n	y	y	n	n	y	n	y	y	n	y	n	y	n	y	y
33	democrat	y	y	y	n	n	n	y	y	y	y	n	n	y	n	y	y
273	democrat	y	n	y	n	n	n	y	y	y	y	n	n	n	n	y	?
355	democrat	n	y	y	n	n	y	y	y	y	y	n	?	n	n	y	y
227	democrat	n	n	y	n	n	y	y	y	y	n	y	n	n	y	y	y
105	democrat	?	?	?	?	n	y	y	y	y	y	?	n	y	y	n	?
2	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	?
369	democrat	n	y	y	n	n	y	y	y	n	y	n	n	n	n	y	y
3	democrat	?	y	y	?	y	y	n	n	n	n	y	n	y	y	n	n
243	republican	n	n	n	n	y	y	y	n	n	n	n	?	n	y	y	y
427	democrat	y	n	y	n	n	n	y	y	y	y	n	n	n	n	y	y
242	democrat	y	n	y	n	n	n	y	y	y	y	y	n	n	y	y	y
86	democrat	n	n	y	n	y	y	n	n	n	y	y	y	y	y	n	y
235	democrat	n	n	y	n	n	y	y	y	y	y	n	y	n	y	y	?
9	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	y
223	democrat	y	n	y	n	n	n	y	y	y	n	y	n	n	n	y	?
375	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	y
20	democrat	y	y	y	n	n	n	y	y	y	n	y	n	n	n	y	y
176	democrat	n	y	y	n	n	n	y	y	y	y	n	n	n	n	y	y
150	democrat	n	n	y	n	n	n	y	y	y	y	n	n	y	n	y	y
216	democrat	n	y	y	y	y	y	n	n	n	y	y	y	y	y	y	?
316	republican	n	y	y	y	y	y	y	?	n	n	n	n	?	?	y	?
409	democrat	y	n	y	n	n	y	y	y	y	n	n	y	?	y	y	y
292	democrat	y	n	y	n	n	y	y	y	y	y	n	?	n	y	n	y
29	republican	y	n	n	y	y	n	y	y	y	n	n	y	y	y	n	y
425	democrat	n	y	y	n	n	?	y	y	y	y	y	n	?	y	y	y
71	democrat	y	n	y	n	n	n	y	y	y	n	n	n	y	n	y	?
264	democrat	y	n	y	n	n	n	y	y	y	n	n	n	n	n	y	?
98	democrat	y	n	n	n	y	y	y	n	n	y	y	n	n	y	n	y
435	republican	n	y	n	y	y	y	n	n	n	y	n	y	y	y	?	n

...	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
182	democrat	n	n	y	n	n	n	y	y	y	y	y	n	n	n	y	y
117	democrat	y	n	y	n	n	n	y	y	y	n	y	n	n	n	y	y
172	republican	n	?	n	y	y	y	n	n	n	y	n	y	y	y	n	y
145	democrat	n	?	n	n	n	y	y	y	y	y	n	n	n	y	n	?
127	republican	n	?	n	y	y	y	n	n	n	n	n	y	y	y	n	n
43	democrat	y	n	y	n	n	n	y	y	y	n	n	n	n	n	n	y
215	republican	y	y	n	y	y	y	n	n	n	y	n	y	y	y	n	y
130	democrat	?	?	y	n	n	n	y	y	?	n	?	?	?	?	?	?
32	democrat	y	y	y	n	n	n	y	y	y	n	y	n	n	n	y	?
350	republican	n	y	y	y	y	y	y	y	y	n	n	y	y	y	n	y
141	republican	n	n	n	y	n	n	y	y	y	y	n	n	y	y	n	y
200	democrat	y	y	n	n	n	n	y	y	?	n	y	n	n	n	y	?
231	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	y
59	republican	n	y	n	y	y	y	n	n	n	y	n	y	y	y	n	y
160	democrat	n	y	y	n	?	y	y	y	y	y	y	n	n	?	n	?
259	democrat	n	n	y	n	n	n	y	y	y	n	y	n	n	n	y	y
139	democrat	n	n	y	n	n	y	y	y	y	y	n	n	n	y	n	y
153	democrat	n	y	y	n	n	y	n	y	y	y	y	n	y	n	y	y
149	republican	n	y	n	y	y	y	n	n	n	y	y	y	y	y	n	y
95	democrat	y	n	y	n	y	y	n	n	n	n	n	n	n	n	n	y
93	democrat	y	y	y	n	n	n	y	y	n	y	y	n	n	?	y	y
129	democrat	n	?	y	n	n	y	n	y	n	y	y	n	n	n	y	y
189	republican	y	?	n	y	y	y	y	y	n	n	n	y	?	y	?	?
96	democrat	y	n	y	n	y	y	n	?	?	n	y	?	?	?	y	y
14	democrat	y	y	y	n	n	y	y	y	?	y	y	?	n	n	y	?
165	democrat	y	y	n	n	y	y	n	n	n	y	y	y	y	y	n	?
384	democrat	y	y	y	n	y	y	n	y	y	y	y	n	n	n	n	y
194	democrat	n	n	y	n	n	n	y	y	y	n	n	n	n	n	y	y
417	republican	y	y	n	y	y	y	n	n	n	y	n	n	y	y	n	y
186	democrat	y	n	y	n	n	n	y	y	y	y	n	?	n	n	y	y

In [69]:

```
nrow(X[test,])
modeldata<-X[test,]
```

130

In [70]:

```
X[-test,]
```

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
4	democrat	n	y	y	n	?	y	n	n	n	n	y	n	y	n	n	y
5	democrat	y	y	y	n	y	y	n	n	n	n	y	?	y	y	y	y
7	democrat	n	y	n	y	y	y	n	n	n	n	n	n	?	y	y	y
8	republican	n	y	n	y	y	y	n	n	n	n	n	n	y	y	?	y
10	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	?	?
11	republican	n	y	n	y	y	n	n	n	n	n	?	?	y	y	n	n

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
12	republican	n	y	n	y	y	y	n	n	n	n	y	?	y	y	?	?
13	democrat	n	y	y	n	n	n	y	y	y	n	n	n	y	n	?	?
15	republican	n	y	n	y	y	y	n	n	n	n	n	y	?	?	n	?
17	democrat	y	n	y	n	n	y	n	y	?	y	y	y	?	n	n	y
18	democrat	y	?	y	n	n	n	y	y	y	n	n	n	y	n	y	y
21	democrat	y	y	y	n	n	?	y	y	n	n	y	n	n	n	y	y
22	democrat	y	y	y	n	n	n	y	y	y	n	n	n	?	?	y	y
24	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	y	y
25	democrat	y	n	y	n	n	n	y	y	y	n	n	n	n	n	y	?
30	democrat	y	y	y	n	n	n	y	y	y	n	y	n	n	n	y	y
31	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	n
35	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	y	y
36	republican	n	y	n	y	y	y	n	n	n	n	n	y	y	y	n	n
38	republican	y	y	n	y	y	y	n	n	n	n	n	n	y	y	n	y
39	republican	n	y	n	y	y	y	n	n	n	y	n	y	y	y	n	n
40	democrat	y	n	y	n	n	n	y	y	y	y	y	n	y	n	y	y
41	democrat	y	y	y	n	n	n	y	y	y	n	?	n	n	n	n	?
44	democrat	y	n	y	n	n	n	y	y	y	n	n	n	n	n	y	y
45	democrat	y	y	y	n	n	n	y	y	y	n	y	n	n	n	n	?
46	democrat	y	y	y	n	n	n	y	y	?	n	y	n	n	n	y	?
47	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	n	y
48	democrat	y	n	y	n	n	n	y	y	?	n	n	n	n	n	n	?
49	democrat	y	y	y	n	n	n	y	y	n	n	n	n	n	y	n	y
50	republican	n	?	n	y	y	y	n	n	n	n	n	y	y	y	n	n
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
393	republican	y	y	n	y	y	y	n	n	n	n	y	y	y	y	n	y
394	republican	?	?	?	?	n	y	n	y	y	n	n	y	y	n	n	?
395	democrat	y	y	?	?	?	y	n	n	n	n	y	n	y	n	n	y
397	democrat	y	y	y	n	y	y	n	y	n	n	y	n	y	n	y	y
398	democrat	y	y	n	n	y	?	n	n	n	n	y	n	y	y	n	y
400	republican	n	y	n	y	?	y	n	n	n	y	n	y	y	y	n	n
401	republican	n	y	n	y	y	y	n	?	n	n	?	?	?	y	n	?
403	republican	?	n	y	y	n	y	y	y	y	y	n	y	n	y	n	y
404	republican	n	y	n	y	y	y	n	n	n	y	n	y	?	y	n	n
405	republican	y	y	n	y	y	y	n	n	n	y	n	y	y	y	n	y
406	republican	n	n	n	y	y	y	n	n	n	n	n	y	y	y	n	y
407	democrat	y	n	y	n	y	y	n	n	y	y	n	n	y	y	n	y
408	democrat	n	n	n	y	y	y	n	n	n	n	y	y	y	y	n	n
410	republican	n	n	n	y	y	y	n	n	n	n	n	y	y	y	n	n
411	republican	n	n	n	y	y	y	n	n	n	n	y	y	y	y	n	y
412	democrat	y	n	y	n	n	y	y	y	y	y	y	n	n	n	n	y
414	republican	y	y	y	y	y	y	y	y	n	y	?	?	?	y	n	y
415	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	n	y
416	democrat	n	y	y	n	n	y	y	y	?	y	n	n	n	n	n	y
418	democrat	y	y	y	n	n	n	y	y	y	y	y	n	y	n	n	y

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17
419	democrat																
420	democrat	y	y	y	n	n	n	y	y	y	n	n	n	n	n	n	y
421	republican	y	y	y	y	y	y	y	y	n	y	n	n	y	y	n	y
422	democrat	n	y	y	n	y	y	y	y	n	n	y	n	y	n	y	y
423	democrat	n	n	y	n	n	y	y	y	y	n	y	n	n	n	y	y
424	democrat	n	y	y	n	n	y	y	y	y	n	y	n	n	y	y	y
429	democrat	?	?	?	n	n	n	y	y	y	y	n	n	y	n	y	y
430	democrat	y	n	y	n	?	n	y	y	y	y	n	y	n	?	y	y
431	republican	n	n	y	y	y	y	n	n	y	y	n	y	y	y	n	y
433	republican	n	?	n	y	y	y	n	n	n	n	y	y	y	y	n	y

We use the rest of the model data to test our model;consequently, it is assigned to testdata variable.

In [71]:

```
testdata<-X[-test,]
nrow(testdata)
```

305

Logistic regression model is formed by using glm function in R language. The meaning of glm is fitting generalized linear models.

In [72]:

```
logitmodel<- glm(V1 ~ V4+V7+V9+V12+V15+V17, family=binomial(link="logit"), data= modeldata)
logitmodel
```

```
Call:  glm(formula = V1 ~ V4 + V7 + V9 + V12 + V15 + V17, family = binomial(link = "logit"),
    data = modeldata)

Coefficients:
(Intercept)          V4n          V4y          V7n          V7y          V9n
     3.2776       3.3063      -0.2505      -1.3029      -2.8804      -2.2424
        V9y         V12n         V12y         V15n         V15y         V17n
    -4.9442       1.2610      -1.8200     -18.1931      -0.0705       0.1346
       V17y
     0.5301

Degrees of Freedom: 129 Total (i.e. Null);  117 Residual
Null Deviance:      163.6
Residual Deviance: 42.88   AIC: 68.88
```

After the model is generated, we use summary code to get a short information about the model that we generate.

In [73]:

```
summary(logitmodel)
anova(logitmodel, test="Chisq")
```

```
Call:
glm(formula = V1 ~ V4 + V7 + V9 + V12 + V15 + V17, family = binomial(link = "logit"),
    data = modeldata)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.53865  -0.06169  -0.00002   0.28523   2.49880

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    3.2776     4.5160   0.726    0.468
V4n            3.3064     4.5938   0.720    0.472
V4y           -0.2505     4.5355  -0.055    0.956
V7n           -1.3029     4.5629  -0.286    0.775
V7y           -2.8804     4.4843  -0.642    0.521
V9n           -2.2424     3.4889  -0.643    0.520
V9y           -4.9442     3.5722  -1.384    0.166
V12n           1.2610     2.0928   0.603    0.547
V12y          -1.8200     2.1078  -0.863    0.388
V15n         -18.1931  2167.5919  -0.008    0.993
```

```
V15y          -0.0705     3.1309  -0.023    0.982
V17n           0.1346     1.4207   0.095    0.925
V17y           0.5301     1.1543   0.459    0.646

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 163.583  on 129  degrees of freedom
Residual deviance:  42.883  on 117  degrees of freedom
AIC: 68.883

Number of Fisher Scoring iterations: 19
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL	NA	NA	129	163.58343	NA
V4	2	79.347659	127	84.23577	5.886740e-18
V7	2	3.490466	125	80.74531	1.746043e-01
V9	2	15.116268	123	65.62904	5.218482e-04
V12	2	17.154162	121	48.47488	1.883740e-04
V15	2	5.312743	119	43.16213	7.020250e-02
V17	2	0.279193	117	42.88294	8.697091e-01

After the model is formed, we use predict function to estimate the model for testdata variable which is the rest of the modeldata.

In [74]:

```
prelogitmodel<-predict(logitmodel,testdata, type = "response")
```

Now, `prelogitmodel` contains the probability that given MP is a republican or a democrat. We need to set a threshold for the probability. If it is greater than 0.6then we will predict that MP is a republican

In [75]:

```
a<-ifelse(prelogitmodel>0.6,"republican","democrat")
```

In [76]:

```
table(a)
```

```
a
  democrat republican
       206         99
```

To compare logitmodel that we constitute and prelogitmodel that predict our model, we use table function. We need to compare the number of democrat congressmen and republican congressmen.For this reason, we take V1 column of the testdata that consists of the class of the congressmen which are democrat and republican. Also, the table(a) that is formed by using prelogitmodel consists of the number of the congressmen that belongs to two class.

In [77]:

```
logistic_table1<-table(real=testdata$V1,predicted=a)
```

In [78]:

```
logistic_table1
```

```
            predicted
real         democrat republican
  democrat        173          6
  republican       33         93
```

When we look at the anova function that gives informations about prelogitmodel, we take out laws that their residual deviance has a small differences. Therefore, V9 and V17 are taken out from the code to change our model.

In [95]:

```
logitmodel<- glm(V1 ~ V4+V9+V12, family=binomial(link="logit"), data= modeldata)
logitmodel
```

```
Call:  glm(formula = V1 ~ V4 + V9 + V12, family = binomial(link = "logit"),
```

```
    data = modeldata)

Coefficients:
(Intercept)          V4n          V4y          V9n          V9y         V12n
      2.178        2.601       -1.217       -2.634       -5.433        1.087
       V12y
     -2.217

Degrees of Freedom: 129 Total (i.e. Null);  123 Residual
Null Deviance:     163.6
Residual Deviance: 49.99  AIC: 63.99
```

In [96]:

```
summary(logitmodel)
anova(logitmodel, test="Chisq")
```

```
Call:
glm(formula = V1 ~ V4 + V9 + V12, family = binomial(link = "logit"),
    data = modeldata)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.55772  -0.25809  -0.04986   0.27824   2.61483

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.178      2.580   0.844  0.39861
V4n            2.601      2.807   0.926  0.35419
V4y           -1.217      2.769  -0.440  0.66026
V9n           -2.634      1.880  -1.401  0.16121
V9y           -5.433      1.965  -2.765  0.00569 **
V12n           1.087      1.995   0.545  0.58590
V12y          -2.217      2.000  -1.109  0.26761
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 163.58  on 129  degrees of freedom
Residual deviance:  49.99  on 123  degrees of freedom
AIC: 63.99

Number of Fisher Scoring iterations: 7
```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)
NULL	NA	NA	129	163.58343	NA
V4	2	79.34766	127	84.23577	5.886740e-18
V9	2	17.44960	125	66.78617	1.625051e-04
V12	2	16.79647	123	49.98970	2.252650e-04

When we look at the table above, the differences between the residual deviances are close to each other. Consequently, we constitute our logitmodel again without V7, V9 and V17 columns.

In [97]:

```
prelogitmodel<-predict(logitmodel,testdata, type = "response")
```

In [98]:

```
a<-ifelse(prelogitmodel>0.6,"republican","democrat")
```

In [99]:

```
table(a)
```

```
a
  democrat republican 
       205        100 
```

In [100]:

```
logistic_table2<-table(real=testdata$V1,predicted=a)
```

In [101]:

```
logistic_table2
```

```
            predicted
real          democrat republican
  democrat         173          6
  republican        32         94
```

According to table functions, the total number of congressmen that is known wrongly is 49; therefore, these two table functions are compared by using Pearson's Chi-squared test. To use Pearson's Chi-squared test, chisq.test(matrix()) is used and compared it X-squared values. The logitmodel is statisticaly good when it has a big X-squared value.

In [102]:

```
chisq.test(logistic_table1)
chisq.test(logistic_table2)
```

```
 Pearson's Chi-squared test with Yates' continuity correction

data:  logistic_table1
X-squared = 164.25, df = 1, p-value < 2.2e-16

 Pearson's Chi-squared test with Yates' continuity correction

data:  logistic_table2
X-squared = 167.14, df = 1, p-value < 2.2e-16
```

The second modal that is used has bigger X-squared value and it has less independent variables;consequently, using this model is statistically good. However, we will choose the second logistic model if we have small X-squared number because the second logistic model has a less independent variables than the first logistic model.

2.SVM Model

a.Definition

SVM is the contraction of Support Vector Machines.Cortes & Vapnik developed SVM's for binary classification. In this model, support vectors are determined by using the closest points that are the elements of the classes. Support vectors pass on the closest points and the distance between the two of support vectors are called margin.Also margin is maximized in this model. There is a hyperplane that seperates the support vectors and margin in the middle. In this model, each of classes shows a tendency to -1 or 1. Therefore, there are two hyperplanes that are $w.x_i>= 1$ when $y_i=1$ and $w.x_i <= -1$ when $y_i=-1$ for x_i's are the set of input, y_i is set of output corresponding to x_i and w is the weight vector that predicts the yi value. H_1 and H_2 are planes such that \begin{equation} H_1:w.x_i=1 \qquad\qquad\qquad \qquad(2.1)\end{equation} and \begin{equation}H_2:w.x_i= -1.\qquad \qquad\qquad\qquad (2.2)\end{equation} Also there is a plane in the middle of this planes such that

\begin{equation}H_0: w.x_i=0.\qquad \qquad\qquad\qquad (2.3)\end{equation} These equations of hyperplanes are stated in [3]. The shortest distance between H_0 and H_1 is called $+d$ and the shortest distance between H_0 and H_2 is called $-d$.

b.Equations of SVM Model

Considering the movement of the support vector changes the boundary; concequently, the form of equation of hyperplanes occurs such that $w^T.x+b=0$ where w is a weigth vector, x is an input vector and b is bias. The distance between H_0 and H_1 is $(w.x+b)/||w||= 1/||w||$, then the margin is $2/||w||$. To maximize margin,w can be minimized. When hyperplanes that we mentioned are gathered up, the equation $y_i(w.x_i)>=1$ is formed. Suppose that we have a problem and we need to minimize w, \begin{equation}minf(x)=(1/2)||w||^2 \qquad\qquad\qquad \qquad (2.4)\end{equation} and $g(x)=y_i(w.x_i)+b=1$ or $g(x)$ can be written as $g(x)=(y_i(w.x_i)+b)-1=0$. It can be solved by Langrange Multiplier and we have two constraints that one of them is $g(x)=0$ and the solution maximum or minimum according to the gradient of f or g. When these two constraints are considered the Langrange Equation which is analysed in [3] must be like this $L(x,a)= f(x)- ag(x)$ and the derivative of $L(x,a)$ is zero. In general \begin{equation}L(x,a)= f(x)- \sum_i a_i g_i \qquad\qquad\qquad \qquad(2.5)\end{equation} In this case given f(x) and g(x) we have Langrangian \begin{equation}min Langrangian= (1/2)||w||^2 - \sum_i a_i((y_i(w.x_i)+b)-1) \qquad\qquad\qquad \qquad(2.6)\end{equation} and the last form of the equation is \begin{equation}min Langrangian= (1/2)||w||^2 - \sum_i a_i((y_i(w.x_i+b)+\sum_i a_i \qquad\qquad\qquad \qquad(2.7)\end{equation}

In SVM model the Langrangian stated in is \begin{equation}(1/2)||w||^2 - \sum_{i=1}^l a_i((y_i(w.x*i+b)+\sum*{i=1}^l a_i \qquad\qquad\qquad \qquad(2.8)\end{equation} where $i,a_i>0$ and l is the number of training points. The derivatives of Langrangian with respect to w and b are zero therefore, we get \begin{equation}w=\sum_{i=1}^l a_iy_ix*i \qquad\qquad\qquad*

\qquad and \qquad \sum{i=1}^l a_iy*i=0 \qquad\qquad\qquad \qquad(2.9)\end{equation} When we rewrite the minimum Langrangian by putting w and $b*=\sum{i=1}^l a_iy*i=0$ then we have maximum Langrangian \begin{equation}max L*=\sum{i=1}^l a*i- (1/2)\sum*{i=1}^l a_ia_jy_iy_j(x_i.x_j) \qquad\qquad\qquad \qquad(2.10)\end{equation} where a_i's are support vectors and positive. The above-stated equations about SVM model emphasized in [3] and these equations informed about the occurence of this model.

In this equation that is emphasized in [4] we learned the margin is maximized or not by using the inner product of x_i and x_j and owing to Kernel function, this computation is easier because Kernel function \begin{equation}K(x_i,x_j)=\phi(x_i).\phi(x_j) \qquad\qquad\qquad \qquad(2.11)\end{equation} describes the inner product or resembels in transformed space.

c. SVM Kernels

There are kernels that are used in SVM model. When the SVM model increases the dimensions of transformed space by using the data, the model determines the proper kernel. There are three type of kernels that are most known and the equations of kernels are determined in [3].

1.Polinomial Kernel
\begin{equation}K(x,y)=(x.y+1)^p \qquad\qquad\qquad\qquad (2.12)\end{equation}

2.Radial Basis Kernel
\begin{equation}K(x,y)=exp{-(|x-y|^2)/2}\sigma^2\qquad\qquad\qquad\qquad(2.13)\end{equation}

3.Sigmoid Kernel
\begin{equation}K(x,y)=tanh(\kappa x.y-\delta) \qquad\qquad\qquad \qquad (2.14)\end{equation}

First of all, we need to download the library of SVM for using the model in R. Therefore, we use this code below

In [87]:

```
library (e1071)
```

After we downloaded the library, we construct our SVM model by using modeldata and classes that are the first column of this data.

In [88]:

```
svmmodel<- svm(V1 ~ . , data=modeldata)
svmmodel
```

```
Call:
svm(formula = V1 ~ ., data = modeldata)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.03030303

Number of Support Vectors:  43
```

After the svmmodel is constructed, to take a short information about svmmodel, summary() function can be used as the following.

In [89]:

```
summary(svmmodel)
```

```
Call:
svm(formula = V1 ~ ., data = modeldata)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1
      gamma:  0.03030303

Number of Support Vectors:  43

 ( 21 22 )


Number of Classes:  2
```

```
Levels:
 democrat republican
```

As indicated above the C-classification is one of the dual representation of SVM and in this type of classification the error function minimized. The minimized error function is $(1/2)w w^T + C\sum{i=1}^n\xi_i$ subject to constraints: $y_i(w^T\phi(x_i)+b)>=1-\xi_i$ where $\xi_i>0$ and $i=1,2,...,N$. In error function C is the capasity constant, w is weight vector or vector of coefficient, b is constant, i is the number of training, y_i is the sign of class and x_i is the set of independent variables. Besides the kernel $\phi(x_i)$ converts from the input data(independent variables) to feature space.

We need to predict that the svmmodel after we set up model by using testdata. Also predict() function is used to predict svmmodel and it assigns svmpredict variable.

In [90]:

```
svmpredict<- predict(svmmodel,testdata)
svmpredict
```

4
democrat
5
democrat
7
republican
8
republican
10
democrat
11
republican
12
republican
13
democrat
15
republican
17
democrat
18
democrat
21
democrat
22
democrat
24
democrat
25
democrat
30
democrat
31
republican
35
democrat
36
republican
38
republican
39
republican
40
democrat
41
democrat
44

44
democrat
45
democrat
46
democrat
47
democrat
48
democrat
49
democrat
50
republican
51
democrat
52
republican
53
democrat
54
republican
55
democrat
56
republican
58
republican
60
republican
62
republican
63
democrat
64
democrat
65
democrat
67
republican
68
republican
69
democrat
70
democrat
72
democrat
73
democrat
75
democrat
76
democrat
77
democrat
78
democrat
79
democrat
80
republican
81
democrat
82
democrat

83
republican
84
republican
87
republican
89
democrat
91
democrat
94
democrat
97
democrat
99
democrat
100
republican
101
democrat
103
democrat
106
democrat
107
republican
108
democrat
109
democrat
110
democrat
111
democrat
112
republican
113
democrat
114
republican
116
democrat
120
republican
121
republican
122
republican
123
republican
124
republican
125
democrat
126
republican
128
democrat
131
democrat
132
democrat
134
republican
135

republican
136
republican
137
republican
138
democrat
140
democrat
142
republican
143
republican
144
democrat
146
democrat
147
republican
148
democrat
151
republican
154
democrat
155
republican
156
republican
157
republican
158
democrat
159
republican
161
republican
163
democrat
166
democrat
167
republican
168
democrat
169
democrat
170
democrat
171
democrat
173
democrat
174
democrat
177
democrat
178
democrat
180
democrat
181
democrat
183
democrat
184

democrat
187
democrat
188
democrat
190
democrat
191
republican
192
republican
193
democrat
195
democrat
196
republican
197
democrat
198
republican
202
democrat
203
democrat
204
democrat
205
republican
206
democrat
208
republican
209
democrat
210
democrat
211
democrat
212
republican
213
democrat
217
democrat
218
republican
219
democrat
220
democrat
224
republican
225
republican
226
republican
228
democrat
229
republican
230
republican
233
democrat

234
republican
236
republican
237
democrat
238
democrat
240
republican
244
democrat
245
democrat
246
democrat
251
republican
252
republican
253
democrat
254
republican
255
democrat
256
democrat
257
republican
258
republican
260
democrat
262
democrat
263
democrat
265
democrat
266
democrat
267
republican
268
democrat
269
democrat
270
democrat
271
democrat
272
democrat
274
republican
275
republican
277
republican
278
republican
279
republican
281
democrat

democrat
282
democrat
283
republican
284
republican
285
democrat
286
democrat
288
democrat
289
democrat
290
democrat
291
democrat
293
democrat
294
democrat
295
democrat
296
republican
297
republican
298
democrat
299
democrat
301
republican
302
democrat
303
republican
304
republican
305
republican
306
republican
307
republican
308
democrat
309
republican
311
republican
312
democrat
314
republican
315
republican
317
democrat
318
democrat
319
democrat
320

democrat
321
democrat
322
democrat
323
democrat
325
republican
326
republican
328
republican
329
democrat
330
democrat
331
republican
332
democrat
334
democrat
335
democrat
336
republican
337
democrat
338
democrat
339
democrat
341
republican
343
democrat
344
republican
346
republican
347
republican
348
republican
351
democrat
352
republican
353
democrat
354
republican
356
democrat
357
republican
358
republican
359
democrat
360
republican
361
democrat

362
democrat
363
democrat
364
democrat
365
republican
367
democrat
368
democrat
370
republican
371
democrat
372
democrat
374
democrat
376
republican
377
democrat
378
republican
379
republican
380
republican
381
democrat
383
republican
386
democrat
387
democrat
388
democrat
389
republican
390
democrat
391
democrat
393
republican
394
democrat
395
democrat
397
democrat
398
democrat
400
republican
401
republican
403
democrat
404
republican
405
republican

406
republican
407
democrat
408
republican
410
republican
411
republican
412
democrat
414
republican
415
democrat
416
democrat
418
democrat
419
democrat
420
democrat
421
republican
422
democrat
423
democrat
424
democrat
429
democrat
430
democrat
431
republican
433
republican

In [91]:

```
summary(svmpredict)
```

democrat
182
republican
123

As noted above summary() code gives a short brief summary about svmpredict variable.

Finally, we have to compare between the real data and the model that we set up by using SVM. When this model is compared with real data, the only column that needed to be taken is the first column because it has the classes of congressmen that are democrat and republican. Besides, we use table()code to make comparison between the real data and svmmodel and we assign the SVM_table variable because we will not write the table data later.

In [92]:

```
SVM_table<-table(testdata$V1,svmpredict)
```

In [93]:

```
SVM_table
```

svmpredict

```
          svmpredict
            democrat republican
  democrat       172          7
  republican      10        116
```

3.ANALYSIS

a.Logistic Regression Model Process

When we formed the first logistic regression model we used $$glm(V1~.,family=binomial(link="logit"), data= modeldata)$$code but it did not converge.For this reason we examined the data set and we tried to figure out the relationship between the class of congressmen and statues. Besides, we swithced the code to $$ glm(V1 ~ V4+V7+V9+V12+V15+V17, family=binomial(link="logit"), data= modeldata)$$ then this code converges but still it is not a best form of our logistic regression model because we have many independent variables in this code. Therefore, we use $$anova(logitmodel,test='Chisq')$$ code to state which of columns are used or not. When the colums are determined, the table of anova function is researched. The residual deviance of anova table tells the dependency of the columns and the class of congressmen by using the difference between the residual deviance for each of columns. When the difference of residual deviance is greater than the other one's, this means that the column has a stronger relationship than the others. Hence these columns are choosed by using method that is mentioned. Later we constituted the second logistic regression model by using the columns that are choosed again. After the prediction function of logitmodel formed, constructing a table for logitmodel we took a threshold for the probability. Then we got a table for comparing the real data. Besides we compared which model is better by using Chi-Square method and the second model is determined. The reason of this detection is having less independent variables although the second logistic regression model has a small chi-square number for some cases.

b. SVM Model Process

To begin with we download a library for using SVM model. Then we constructed our SVM model simply because it did not get a warning for not converging. Moreover, we predicted the SVM model that we formed and we made a comparison between the prediction of this model and the real data by using table function. According to table data,there are 18 congressmen are known wrongly which means that there are congressmen that are democrat but they pretend republican or vise versa. It is significant because if the republican or democrat congressmen attract supporter for themselves then they will need to find this off diagonal terms.

c. Chi-Square Method

Chi-Square method is one of the tests that detemines the existence of the relationship between two variables that are numerical or not in statistic. This method makes a comparison between the observed values and the expected values theoritically. To begin with, there is a subtraction between observed values and expected values. Then this test squares the difference that it found and divided by expected values. The formula of Chi-Square stated in [5] is as the following
\begin{equation}\chi^{2}*c*=*sum*{i=1}^{k}\frac{(O_i-E_i)^2}{E_i}\qquad \qquad\qquad\qquad
\qquad\qquad\qquad(3.1)\end{equation}

where the O_i is the observed values, E_i is the expected values and c is the degree of freedom. Chi- Square method tests whether there is a dependency in data set. Also if the calculated value of this method is bigger than the value of table data then there will be a relationship between the variables.

d. Chi-Square Method in R

When we use this method we use chisq.test(matrix()) code and we determine which model is statistically good according to the results of the chisq.test()code.

In [94]:

```
chisq.test(logistic_table2)
chisq.test(SVM_table)
```

```
 Pearson's Chi-squared test with Yates' continuity correction

data:  logistic_table2
X-squared = 167.14, df = 1, p-value < 2.2e-16
```

```
 Pearson's Chi-squared test with Yates' continuity correction

data:  SVM_table
X-squared = 235.14, df = 1, p-value < 2.2e-16
```

According to the results of these codes, if we compare of two model that we constituted the SVM model is better than the Logistic Regression Model because the X-squared value of SVM model is greater than Logistic Regression model and the SVM model is easier to form.

easier to form.

References

[1] : C. Manning.(2007),Logistic regression,p1.

[2] : R.B. Burns & R. Burns,(2008) Business Research Methods and Statistics using SPSS,p.573

[3] : R.Berwick & V.Idiot,(n.d.) An Idiot's guide to Support vector machines (SVMs),pp.4-10.

[4] : Support Vector Machines(SVM),Retrieved from http://www.statsoft.com/Textbook/Support-Vector-Machines

[5] : S.Deviant,The Practically Cheating Statistics Handbook,Retrieved from http://www.statisticshowto.com/probability-and-statistics/chi-square/

In [ ]:

```

```