Question 1

1. Stemming the Federalists

Such is a segment from file Federalist01.txt in folder fp\_hamilton\_test\_clean:

general introduct

independ journal

peopl state new york

unequivoc experi ineffici

subsist feder govern call upon deliber

new constitut unit state america subject

speak import comprehend consequ

noth less exist union safeti welfar

part compos fate empir mani

respect interest world frequent

remark seem reserv peopl

countri conduct exampl decid import

question whether societi men realli capabl

establish good govern reflect choic whether

forev destin depend polit

constitut accid forc truth

remark crisi arriv may proprieti

regard era decis made

wrong elect part shall act may view deserv

consid general misfortun mankind

idea will add induc philanthropi

patriot heighten solicitud consider

good men must feel event happi will choic

1. Read Federalists into Workspace. This is segment of infile No.15 in hamilton.test:

[[15]]

[1] "subject" "continu" "concern" "power" "congress"

[6] "regul" "elect" "member" "new" "york"

[11] "packet" "tuesday" "februari" "26" "1788"

[16] "peopl" "state" "new" "york" "seen"

[21] "uncontrol" "power" "elect" "feder" "govern"

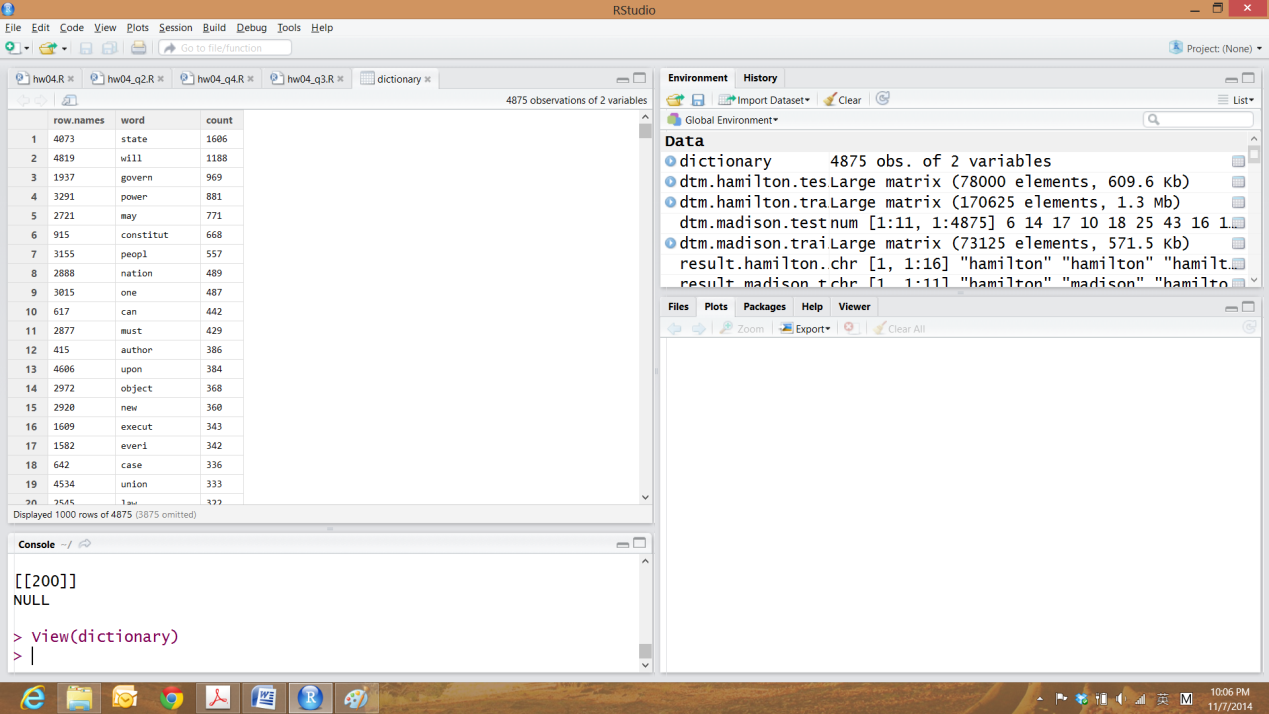
[26] "without" "hazard" "commit" "state" "legislatur"

[31] "let" "us" "now" "see" "danger"

[36] "side" "confid" "ultim" "right" "regul"

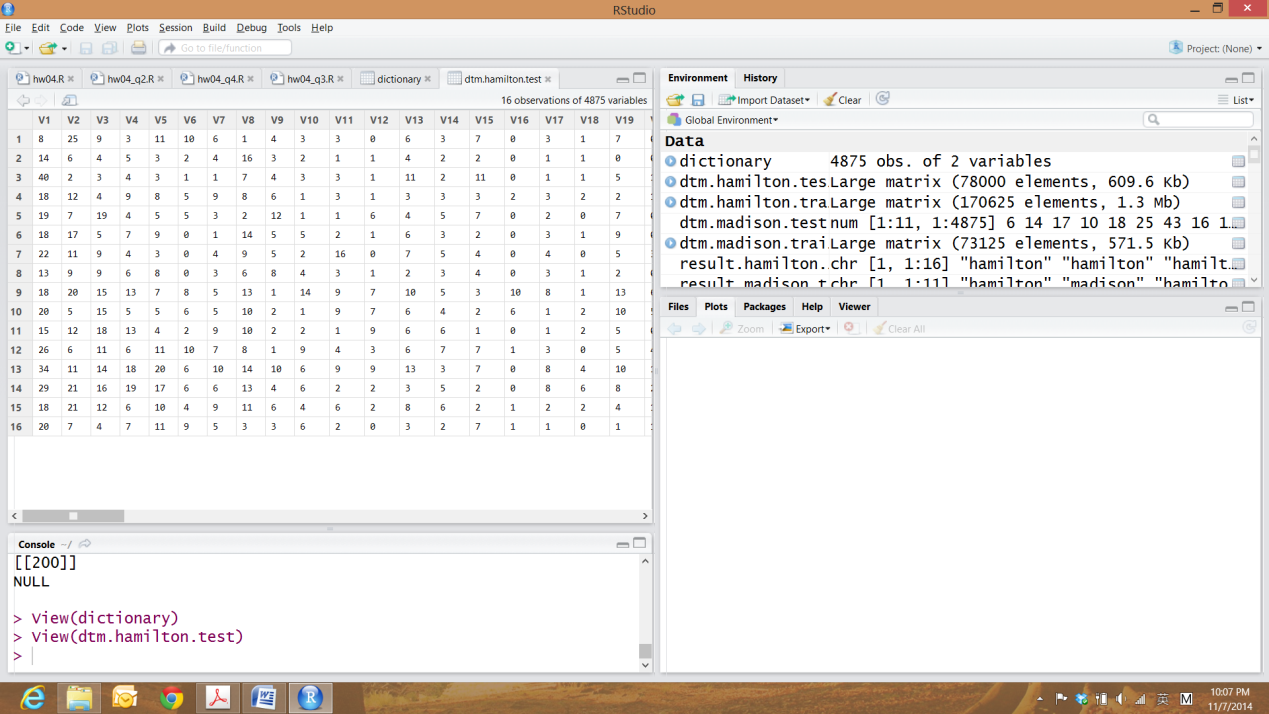
[41] "elect" "union" "pretend" "right" "ever"

1. Make a dictionary.



1. Calculate document-terms matrix.

This is a segment from dtm.hamilton.test



1. Calculate vectors of log probability with given penalty 

Such as a segment from logp.hamilton.test

> logp.hamilton.test

[1] -3.867745 -4.415339 -4.554825 -4.812970 -4.767515 -5.368599

[7] -5.206798 -4.696066 -5.341938 -5.438538 -5.368599 -5.760508

[13] -5.087765 -5.513738 -5.409973 -6.627443 -5.721304 -6.494034

[19] -5.140122 -6.145978 -5.721304 -6.008854 -5.467943 -5.409973

[25] -5.780703 -5.438538 -5.315970 -6.206565 -6.580967 -6.117008

[31] -5.865823 -5.665236 -7.273183 -6.304939 -5.934784 -6.175813

[37] -6.088853 -6.145978 -5.780703 -5.888286 -6.781431 -5.665236

[43] -5.888286 -6.206565 -6.627443 -5.721304 -6.453246 -6.238293

[49] -6.088853 -6.376345 -5.843854 -5.801313 -6.453246 -5.934784

[55] -7.590939 -6.206565 -6.034816 -5.482976 -7.368307 -6.145978

[61] -6.580967 -5.843854 -6.414056 -5.958869 -6.899072 -6.340005

Question 2

We use the basic formula Naive Bayes Classifier calculate the likelihood,



Take log() on both side, we have,

If, we assign this Federalist to Hamilton; otherwise, assign it to Madison.

## define a function

# Naive Bayes Classification Procedure

naiveBayes <- function(logp.hamilton.train, logp.madison.train,

log.prior.hamilton, log.prior.madison , dtm.test){

# initial result vector

result = c();

# calculate the words probability vector of each Federalist in dtm.test

for(i in 1: nrow(dtm.test)){

# probability = frenquency

temp= dtm.test[i,];

# directly calculate log likelihood of each Federalist i in dtm.test

# the number of word \* prob that its belongs to class k(k = 1, 2)

lh.hamilton = t(temp) %\*% logp.hamilton.train + log.prior.hamilton;

# print(lh.hamilton);

lh.madison = t(temp) %\*% logp.madison.train + log.prior.madison;

# print(lh.madison)

if(lh.hamilton >= lh.madison){

class = "hamilton";

}else{

class = "madison";

}

result = cbind(result, class);

}

return(result);

}

The we test this model with dtm.hamilton.test and dtm.madison.test. The results shows below:

> result.hamilton.test

class class class class class class

[1,] "hamilton" "hamilton" "hamilton" "hamilton" "hamilton" "hamilton"

class class class class class class

[1,] "hamilton" "hamilton" "hamilton" "hamilton" "hamilton" "hamilton"

class class class class

[1,] "hamilton" "hamilton" "hamilton" "hamilton"

> result.madison.test

class class class class class class

[1,] "hamilton" "madison" "hamilton" "madison" "madison" "madison"

class class class class class

[1,] "madison" "madison" "hamilton" "madison" "hamilton"

Then we see that all Federalists in Hamilton testing set are classified correctly. However, Federalist No.10, No.41, No.57, No.62 from Madison are wrong classified as from Hamilton.

Question 3

The result is as below,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Predicted Class | | Total |
| Null | Non-null |
| True Class | Null | 7(0.636) | 4(0.364) | 11 |
| Non-null | 0(0.000) | 16(1.000) | 16 |
| Tatal | | 7 | 20 |  |

> print(sprintf('The percentage of correctly classified paper is: %s', prec.correct(result.hamilton.test, result.madison.test)));

[1] "The percentage of correctly classified paper is: 0.851851851851852"

> # true positive

> print(sprintf("True Positive:%s",truePos(result.hamilton.test)));

[1] "True Positive:1"

> # true nagetive

> print(sprintf("True Negative:%s",trueNeg(result.madison.test)));

[1] "True Negative:0.636363636363636"

> # false\_positive

> print(sprintf("False Positive:%s",falsePos(result.madison.test)));

[1] "False Positive:0.363636363636364"

> # false\_negative

> print(sprintf("False Negaitive:%s",falseNeg(result.hamilton.test)));

[1] "False Negaitive:0"

Question 4

1. Using 5-fold cross-validation on the training set.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1/|D| | 10/|D| | 100/|D| | 1000/|D| | 10000/|D| |
| Correct rate | 0.76 | 0.82 | 0.86 | 0.92 | 0.72 |
| False Neg | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| False Pos | 0.80 | 0.60 | 0.47 | 0.27 | 0.93 |





Notice: This rate could be different if we use "randomly" Cross Validation and "consecutively" Cross Validation since there is slightly difference near 0 in false Negative rates. The "consecutively" CV test result is



This "consecutively" selected cross folds would cause overestimation on false negative rates. On the other hand, the "randomly" selected cross folds would underestimate this rate at the best . This situation also happens on correctly classification rate and false positive rate.



1. Best 

We could conclude from the graphs that seems to be the best penalty in this Naive Bayes classifier. On one hand, it produces a highest correctly classification rate, at about 92%; On the other hand, it produces a lowest false positive rate at about 0.27, given that their false negative rates are the same.

1. True rates

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1/|D| | 10/|D| | 100/|D| | 1000/|D| | 10000/|D| |
| Correct rate | 0.630 | 0.704 | 0.852 | 0.926 | 0.741 |
| False Neg | 0.000 | 0.000 | 0.000 | 0.063 | 0.000 |
| False Pos | 0.909 | 0.727 | 0.364 | 0.091 | 0.636 |







We could plot its ROC curve to test which thresholds generated by different parameter is a good classifier.



It is shown that is the best parameter because the threshold generated by this parameter is huge to the top left of the curve(high true positive rate and low false positive rate). This is consistent to our conclusion in b).

1. Difference between true error rate and cross validation error rates.

We calculate percentage error under the best value of .

|  |  |  |  |
| --- | --- | --- | --- |
| Rate | CorrectClassified | FalseNeg | FalsePos |
| Percentage | 0.64% | -100% | 193% |

We could see that the correct-classified rate and false positive rate are overestimated but the false negative rate is underestimated by the "randomly" selected cross-validation sampling method.

As discuss on the false negative rate, the flexibility of cross-validation sampling (randomly, consecutively, interleaved, etc) could be one of the reasons that leads to difference on such rates. In addition, one reason could be the numbers of training samples. The numbers of training samples in both hamilton.train and madison.train accounts for the prior significantly, so that the classification result. Also, the number of folds could leads to bias in estimation of such rates, but this effect should be so slight that can be ignored, like LOOCV and 10-folds CV are almost same.

In general, the flexibility (the variability in how the observations are divided into 5-folds) is related to the estimation of classification error in Cross-validation sampling. This significantly depends on the value of . So choosing a best penalty in Naive Bayes Classification is the way to control the flexibility, hence the best way to minimum misclassification rate.