Assessment 4

Jacob John

Complete all **Exercises**, and submit answers to **VtopBeta**

Datasets

```
### load packages
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
library(knitr)
```

Iris dataset for training and testing

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

Split it into training set and testing set and validation set

```
ir_data=iris
set.seed(100)
head(ir_data)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
intrain <- createDataPartition(y = ir_data$Species, p= 0.7, list = FALSE)
training<-iris[intrain,]
testing<-ir_data[-intrain,]
dim(training);dim(testing)
```

```
## [1] 105   5
```

```
## [1] 45  5
```

```
summary(ir_data)
```

```
##   Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
##  Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
##  1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
##  Median :5.800   Median :3.000   Median :4.350   Median :1.300
##  Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
##  3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
##  Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
##        Species
##  setosa    :50
##  versicolor:50
##  virginica :50
##
##
##
```

```
training[["Species"]] = factor(training[["Species"]])
trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
```

The results of confusion matrix show that this time the accuracy on the test set is **95.56%**.

using e1071

```
library(e1071)
model <- naiveBayes(Species ~., data = training)
class(model)
```

```
## [1] "naiveBayes"
```

```
summary(model)
```

```
##         Length Class  Mode
## apriori 3      table  numeric
## tables  4      -none- list
## levels  3      -none- character
## call    4      -none- call
```

```
print(model)
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     setosa versicolor  virginica
##  0.3333333  0.3333333  0.3333333
##
## Conditional probabilities:
##             Sepal.Length
## Y                [,1]      [,2]
##   setosa     5.071429 0.3409083
##   versicolor 5.825714 0.4667427
##   virginica  6.540000 0.6611932
##
##             Sepal.Width
## Y                [,1]      [,2]
##   setosa     3.517143 0.3416962
##   versicolor 2.748571 0.2974118
##   virginica  2.962857 0.3263756
##
##             Petal.Length
## Y                [,1]      [,2]
##   setosa     1.471429 0.1856173
##   versicolor 4.182857 0.4712223
##   virginica  5.525714 0.5653437
##
##             Petal.Width
## Y                 [,1]      [,2]
##   setosa     0.2514286 0.1039554
##   versicolor 1.3114286 0.1794951
##   virginica  1.9885714 0.2857101
```

```
preds <- predict(model, newdata = training)
table(preds,training$Species)
```

```
##
## preds        setosa versicolor virginica
##   setosa         35          0         0
##   versicolor      0         33         3
##   virginica       0          2        32
```

```
(35+33+32)/(35+33+2+32+3)#change this according to the diagonal element of the previous statement result
```

```
## [1] 0.952381
```

Accuracy is **95.2381%**.

Using mlbench

```
library(mlbench)
data("HouseVotes84")
data(HouseVotes84, package = "mlbench")
model <- naiveBayes(Class ~ ., data = HouseVotes84)
predict(model, HouseVotes84[1:10,])
```

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

```
predict(model, HouseVotes84[1:10,], type = "raw")
```

```
##           democrat   republican
##  [1,] 1.029209e-07 9.999999e-01
##  [2,] 5.820415e-08 9.999999e-01
##  [3,] 5.684937e-03 9.943151e-01
##  [4,] 9.985798e-01 1.420152e-03
##  [5,] 9.666720e-01 3.332802e-02
##  [6,] 8.121430e-01 1.878570e-01
##  [7,] 1.751512e-04 9.998248e-01
##  [8,] 8.300100e-06 9.999917e-01
##  [9,] 8.277705e-08 9.999999e-01
## [10,] 1.000000e+00 5.029425e-11
```

```
pred <- predict(model, HouseVotes84)
table(pred, HouseVotes84$Class)
```

```
##
## pred         democrat republican
##   democrat        238         13
##   republican       29        155
```

```
(238+155)/(238+155+29+13)
```

```
## [1] 0.9034483
```

Accuracy is **90.34483%**.

```
## using laplace smoothing:
model <- naiveBayes(Class ~ ., data = HouseVotes84, laplace = 3)
pred <- predict(model, HouseVotes84[,-1])
table(pred, HouseVotes84$Class)
```

```
##
## pred         democrat republican
##   democrat        237         12
##   republican       30        156
```

```
(237+156)/(237+156+12+30)
```

```
## [1] 0.9034483
```

Accuracy is still **90.34483%**.

Using a contingency table

```
data(Titanic)
m <- naiveBayes(Survived ~ ., data = Titanic)
m
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.formula(formula = Survived ~ ., data = Titanic)
##
## A-priori probabilities:
## Survived
##       No      Yes
## 0.676965 0.323035
##
## Conditional probabilities:
##         Class
## Survived        1st        2nd        3rd       Crew
##      No  0.08187919 0.11208054 0.35436242 0.45167785
##      Yes 0.28551336 0.16596343 0.25035162 0.29817159
##
##         Sex
## Survived       Male     Female
##      No  0.91543624 0.08456376
##      Yes 0.51617440 0.48382560
##
##         Age
## Survived      Child      Adult
##      No  0.03489933 0.96510067
##      Yes 0.08016878 0.91983122
```

```
predict(m, as.data.frame(Titanic))
```

```
##  [1] Yes No  No  No  Yes Yes Yes Yes No  No  No  No  Yes Yes Yes Yes Yes
## [18] No  No  No  Yes Yes Yes Yes No  No  No  No  Yes Yes Yes Yes
## Levels: No Yes
```

Sentiment Analysis of Movie Reviews

```
# Load additional libraries
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
library(RTextTools)
```

```
## Loading required package: SparseM
```

```
##
## Attaching package: 'SparseM'
```

```
## The following object is masked from 'package:base':
##
##     backsolve
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Library for parallel processing
library(doMC)
```

```
## Loading required package: foreach
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
registerDoMC(cores=detectCores())  # Use all available cores
```

Reading the data

```
df<- read.csv("movie-pang02.csv", stringsAsFactors = FALSE)
glimpse(df)
```

```
## Observations: 2,000
## Variables: 2
## $ class <chr> "Pos", "Pos", "Pos", "Pos", "Pos", "Pos", "Pos", "Pos", ...
## $ text  <chr> " films adapted from comic books have had plenty of succ...
```

```
# Randomize the dataset
set.seed(1)
df <- df[sample(nrow(df)), ]
df <- df[sample(nrow(df)), ]
glimpse(df)
```

```
## Observations: 2,000
## Variables: 2
## $ class <chr> "Neg", "Pos", "Neg", "Neg", "Neg", "Neg", "Neg", "Neg", ...
## $ text  <chr> " frank detorri s   bill murray   a single dad who lives...
```

```
# Convert the 'class' variable from character to factor.
df$class <- as.factor(df$class)
```

Bag of Words Tokenisation

```
corpus <- Corpus(VectorSource(df$text))
corpus
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
## Content:  documents: 2000
```

```
inspect(corpus[1:3])
```

```
## <<SimpleCorpus>>
## Metadata:  corpus specific: 1, document level (indexed): 0
```

Content: documents: 3
##
[1] frank detorri s bill murray a single dad who lives on beer and junk food with no apparent unders tanding of sanitation or hygiene much to the dismay of his preteen daughter shane elena franklin wh en he uses the 10 second rule to retrieve a hard boiled egg from a chimp s cage at the zoo and downs it he introduces a lethal bacteria into his system inside his skin the city of frank is in turmoil thanks to the vote pandering of mayor phlegmming voice of william shatner so it s up to one frank pd white bl ood cell voice of chris rock to save the day in peter and bobby farrelly s osmosis jones the city of frank is a brightly animated animation directed by piet kroon and tom sito cellular municipality wher e osmosis jones is a typical rogue cop looking for another chance he s inadvertently teamed up with drix voice of david hyde pierce tv s frasier a cold capsule with 12 hours worth of painkillers to dispe nse this quarrelling duo are about to go on a fantastic voyage in order to hunt down thrax voice of laurence fishburne the virus intent on shutting down frank while the animation is certainly colorful to look at osmosis jones story is a hackneyed one the story cries out for puny puns but we only get occasional sprinklings of wit or bodily humor drix graduated phi beta capsule he departs on a bus headed for bladder neither the hero or villain is particularly interesting thrax looks like an animated pr edator although hyde pierce is a delightful sidekick adults can desperately keep their eyes peeled for small amusements the animators dot along the landscape meanwhile back in live action land bill mu rray is reduced to nothing more than a walking gross out joke there s no particular enjoyment to be found watching him vomit on molly shannon she plays shane s teacher mrs boyd or hoisting his ingrown toena il onto a restaurant table one must wonder how the climatic flatlining of a child s father will play to t he family audience as well rest assured the whole enchilada is wrapped up with a fart joke while far less offensive than the farrelly s last effort me myself and irene that film at least spiked some co mic highs with jim carrey s hijinx osmosis jones will probably be ok for the kids but the farrelly s playing for the family audience is like watching marilyn manson croon a phil collins tune

[2] synopsis in phantom menace the galaxy is divided into power groups whose interests will inevitably collide in later sequels there is an overarching galactic united nations type organization called the sen ate presided by a weak chancellor within the senate two camps are at odds a bickering isolationist al liance called the republic and their aggressive rival the trade federation preserving law and order are a council of jedi knights who are meanwhile searching for a prophesied chosen one of virgin birth manipulat ing events behind the scenes is a dangerous reemerging clan called the dark lords of sith so shadowy and secretive that they comprise a phantom menace jedi knight qui gon jinn liam neeson and his appren tice obi wan kenobi ewan mcgregor witness an invasion of teenage queen amidala s home planet naboo and b efriend a gungan named jar jar ahmed best on the desert planet of tatooine the two jedi jar jar a nd amidala natalie portman attend a lengthy drag race involving the young boy anakin skywalker jake ll oyd the five protagonists try to solicit help for freeing naboo by visiting the city planet of coruscan t where a lot of debate and political maneuvering takes place can they free amidala s helpless planet opinion on tv last night i watched young wannabe celebs pay $400 a ticket and come running out of theate rs to bask in front of news cameras gushing with testimonials of the phantom menace s greatness in exchang e for a few seconds of being on national television given this kind of media mania i wondered if phantom menace the most anticipated movie of 1999 could possibly live up to the extraordinary hype that preceded it does phantom menace match the exaggerated hype director george lucas answers it s only a movie to me any movie with russian sounding accents for bad guys jamaican accents for good guys and middle e astern sounding accents for seedy gamblers accents can be expected to be more tongue in cheek than profound visually star wars episode i the phantom menace 1999 is a kid show where parents can take their yo ung ones to marvel at child friendly cgi characters and wondrous backdrops even if the character dialogue mostly geopolitics is beyond the level of children it is left to parents to patiently explain the conve rsation droid origins family lineage the definitions of terms like blockade appeasement federation alliance symbiosis satellite controlled robots et cetera at least this much is clear there s plenty of eye candy and in the last few minutes it s good guys and joe camel lookalikes versus a caped horned r ed devil character and his mechanical hordes weaknesses weaknesses lie in the writing and in the perfor mance at first it seems like the film is to be an invasion story but then phantom takes an hour long de tour to cover one chariot race before returning to the invasion theme this dilutes the central story a dditionally smaller scenes seem written self consciously as if they were added more to fill us in on ext raneous background information for other movies rather than form an integral part of the present movie ve teran actors liam neeson and ewan mcgregor noticeably outperform the other acting leads better ensemble c hemistry between the five leads and background information that is central to a tight story line could have made have given phantom stronger performances and storytelling punch strengths on the bright side phant om menace as a big budget production is far ahead of the competition in terms of making whimsical creatures worlds and vehicles appear real the film boasts sophisticated top of the line visuals and quality exoti c costumes a musical score entertaining enough to stand alone and three worthwhile sequences in the seco nd half bottom line seeing the film is entertaining and informative like a visual theme park with st ar wars filler information serving as dialogue between impressive money shots we are bound to be complete ly inundated by star wars publicity music and tie ins for the next few months
[3] terrence malick made an excellent 90 minute film adaptation of james jones world war ii novel un fortunately he buried it within an overlong and overreaching 3 hour long pseudo epic this is a shame be cause the film features an outstanding performance by nick nolte the best scene is when nick nolte s char acter lt col tall is forced to deal with the direct refusal by capt staros elias koteas to exe cute an order nolte s reaction and transformation may be the best work of his career had terrence mali ck concentrated on the great performances of nolte and koteas as well as those by sean penn woody harrelso n and john cusack he could have made a truly great film instead malick saddled the film with ploddi ng pacing unnecessary flashbacks and a voice over narration all designed to telegraph the great philosop

```
hical underpinnings of the story    the narration was especially annoying as much of it sounded like very ba
d high school poetry    with a lot of editing    the core story could be transformed into a truly classic war
film    hopefully    the dvd version of this film will feature options to suppress the narration    and perhap
s will even provide for an alternate    shorter version of the film    i give this film
```

Data Cleanup

```
# Use dplyr's  %>% (pipe) utility to do this neatly.
corpus.clean <- corpus %>%
  tm_map(content_transformer(tolower)) %>%
  tm_map(removePunctuation) %>%
  tm_map(removeNumbers) %>%
  tm_map(removeWords, stopwords(kind="en")) %>%
  tm_map(stripWhitespace)
```

```
## Warning in tm_map.SimpleCorpus(., content_transformer(tolower)):
## transformation drops documents
```

```
## Warning in tm_map.SimpleCorpus(., removePunctuation): transformation drops
## documents
```

```
## Warning in tm_map.SimpleCorpus(., removeNumbers): transformation drops
## documents
```

```
## Warning in tm_map.SimpleCorpus(., removeWords, stopwords(kind = "en")):
## transformation drops documents
```

```
## Warning in tm_map.SimpleCorpus(., stripWhitespace): transformation drops
## documents
```

Document Term Matrix

```
dtm <- DocumentTermMatrix(corpus.clean)
inspect(dtm[40:50, 10:15])
```

```
## <<DocumentTermMatrix (documents: 11, terms: 6)>>
## Non-/sparse entries: 6/60
## Sparsity           : 91%
## Maximal term length: 8
## Weighting          : term frequency (tf)
## Sample             :
##     Terms
## Docs apparent assured audience back bacteria beer
##   40        0       0        1    1        0    0
##   41        0       0        1    0        0    0
##   42        0       0        0    0        0    0
##   43        0       0        0    0        0    0
##   44        0       0        0    1        0    0
##   45        0       0        0    0        0    0
##   46        0       0        2    0        0    0
##   47        0       0        0    0        0    0
##   48        0       0        0    0        0    0
##   50        0       0        2    0        0    0
```

Paritioning

```
df.train <- df[1:1500,]
df.test <- df[1501:2000,]

dtm.train <- dtm[1:1500,]
dtm.test <- dtm[1501:2000,]

corpus.clean.train <- corpus.clean[1:1500]
corpus.clean.test <- corpus.clean[1501:2000]
```

Feature set selection

```
dim(dtm.train)
```

```
## [1]  1500 38957
```

```
fivefreq <- findFreqTerms(dtm.train, 5)
length((fivefreq))
```

```
## [1] 12144
```

```
# Use only 5 most frequent words (fivefreq) to build the DTM
dtm.train.nb <- DocumentTermMatrix(corpus.clean.train, control=list(dictionary = fivefreq))
dim(dtm.train.nb)
```

```
## [1]  1500 12144
```

```
dtm.test.nb <- DocumentTermMatrix(corpus.clean.test, control=list(dictionary = fivefreq))
dim(dtm.train.nb)
```

```
## [1]  1500 12144
```

Boolean feature Multinomial Naive Bayes

```
# Function to convert the word frequencies to yes (presence) and no (absence) labels
convert_count <- function(x) {
  y <- ifelse(x > 0, 1,0)
  y <- factor(y, levels=c(0,1), labels=c("No", "Yes"))
  y
}

# Apply the convert_count function to get final training and testing DTMs
trainNB <- apply(dtm.train.nb, 2, convert_count)
testNB <- apply(dtm.test.nb, 2, convert_count)

# Train the classifier
system.time( classifier <- naiveBayes(trainNB, df.train$class, laplace = 1) )
```

```
##    user  system elapsed
##   9.563   0.804  11.693
```

```
# Use the NB classifier we built to make predictions on the test set.
system.time( pred <- predict(classifier, newdata=testNB) )
```

```
##    user  system elapsed
## 287.448   8.662 365.658
```

```
# Create a truth table by tabulating the predicted class labels with the actual class labels
table("Predictions"= pred,  "Actual" = df.test$class )
```

```
##            Actual
## Predictions Neg Pos
##         Neg 224  54
##         Pos  41 181
```

Confusion Matrix

```
conf.mat <- confusionMatrix(pred, df.test$class)
conf.mat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction Neg Pos
##        Neg 224  54
##        Pos  41 181
##
##                Accuracy : 0.81
##                  95% CI : (0.7728, 0.8435)
##     No Information Rate : 0.53
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.6174
##  Mcnemar's Test P-Value : 0.2183
##
##             Sensitivity : 0.8453
##             Specificity : 0.7702
##          Pos Pred Value : 0.8058
##          Neg Pred Value : 0.8153
##              Prevalence : 0.5300
##          Detection Rate : 0.4480
##    Detection Prevalence : 0.5560
##       Balanced Accuracy : 0.8077
##
##        'Positive' Class : Neg
##
```

```
conf.mat$byClass
```

```
##          Sensitivity          Specificity       Pos Pred Value
##            0.8452830            0.7702128            0.8057554
##       Neg Pred Value            Precision               Recall
##            0.8153153            0.8057554            0.8452830
##                   F1           Prevalence       Detection Rate
##            0.8250460            0.5300000            0.4480000
## Detection Prevalence    Balanced Accuracy
##            0.5560000            0.8077479
```

```
conf.mat$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   8.100000e-01   6.174291e-01   7.728180e-01   8.434678e-01   5.300000e-01
## AccuracyPValue  McnemarPValue
##   3.570547e-39   2.182578e-01
```

```
# Prediction Accuracy
conf.mat$overall['Accuracy']
```

```
## Accuracy
##     0.81
```