Chapter 4: Classification using Naive Bayes

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Ctrl+Shift+K* to preview the HTML file).

Example: Filtering spam SMS messages

Step 1: Download the data

```
URL <- "http://cox.csueastbay.edu/~esuess/classes/Statistics_6620/Presentations/ml6/sms_spam.csv"
download.file(URL, destfile = "./sms_spam.csv", method="curl")
```

Step 2: Exploring and preparing the data —-

```
# read the sms data into the sms data frame
sms_raw <- read.csv("sms_spam.csv", stringsAsFactors = FALSE)

# examine the structure of the sms data
str(sms_raw)
```

```
## 'data.frame':    5559 obs. of  2 variables:
##  $ type: chr  "ham" "ham" "ham" "spam" ...
##  $ text: chr  "Hope you are having a good week. Just checking in" "K..give back my thanks." "Am also
```

```
# convert spam/ham to factor.
sms_raw$type <- factor(sms_raw$type)

# examine the type variable more carefully
str(sms_raw$type)
```

```
##  Factor w/ 2 levels "ham","spam": 1 1 1 2 2 1 1 1 2 1 ...
```

```
table(sms_raw$type)
```

```
##
##  ham spam
## 4812  747
```

```
# build a corpus using the text mining (tm) package
library(tm)
```

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

```r
sms_corpus <- VCorpus(VectorSource(sms_raw$text))

# examine the sms corpus
print(sms_corpus)
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 5559
```

```r
inspect(sms_corpus[1:2])
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 2
##
## [[1]]
## <<PlainTextDocument>>
## Metadata:  7
## Content:  chars: 49
##
## [[2]]
## <<PlainTextDocument>>
## Metadata:  7
## Content:  chars: 23
```

```r
as.character(sms_corpus[[1]])
```

```
## [1] "Hope you are having a good week. Just checking in"
```

```r
lapply(sms_corpus[1:2], as.character)
```

```
## $`1`
## [1] "Hope you are having a good week. Just checking in"
##
## $`2`
## [1] "K..give back my thanks."
```

```r
# clean up the corpus using tm_map()
sms_corpus_clean <- tm_map(sms_corpus, content_transformer(tolower))

# show the difference between sms_corpus and corpus_clean
as.character(sms_corpus[[1]])
```

```
## [1] "Hope you are having a good week. Just checking in"
```

```r
as.character(sms_corpus_clean[[1]])
```

```
## [1] "hope you are having a good week. just checking in"
```

```r
sms_corpus_clean <- tm_map(sms_corpus_clean, removeNumbers) # remove numbers
sms_corpus_clean <- tm_map(sms_corpus_clean, removeWords, stopwords()) # remove stop words
sms_corpus_clean <- tm_map(sms_corpus_clean, removePunctuation) # remove punctuation
```

```r
# tip: create a custom function to replace (rather than remove) punctuation
removePunctuation("hello...world")
```

```
## [1] "helloworld"
```

```r
replacePunctuation <- function(x) { gsub("[[:punct:]]+", " ", x) }
replacePunctuation("hello...world")
```

```
## [1] "hello world"
```

```r
# illustration of word stemming
library(SnowballC)
wordStem(c("learn", "learned", "learning", "learns"))
```

```
## [1] "learn" "learn" "learn" "learn"
```

```r
sms_corpus_clean <- tm_map(sms_corpus_clean, stemDocument)

sms_corpus_clean <- tm_map(sms_corpus_clean, stripWhitespace) # eliminate unneeded whitespace

# examine the final clean corpus
lapply(sms_corpus[1:3], as.character)
```

```
## $`1`
## [1] "Hope you are having a good week. Just checking in"
##
## $`2`
## [1] "K..give back my thanks."
##
## $`3`
## [1] "Am also doing in cbe only. But have to pay."
```

```r
lapply(sms_corpus_clean[1:3], as.character)
```

```
## $`1`
## [1] "hope good week just check"
##
## $`2`
## [1] "kgive back thank"
##
## $`3`
## [1] "also cbe pay"
```

```r
# create a document-term sparse matrix
sms_dtm <- DocumentTermMatrix(sms_corpus_clean)

# alternative solution: create a document-term sparse matrix directly from the SMS corpus
sms_dtm2 <- DocumentTermMatrix(sms_corpus, control = list(
  tolower = TRUE,
  removeNumbers = TRUE,
  stopwords = TRUE,
  removePunctuation = TRUE,
  stemming = TRUE
))

# alternative solution: using custom stop words function ensures identical result
sms_dtm3 <- DocumentTermMatrix(sms_corpus, control = list(
  tolower = TRUE,
  removeNumbers = TRUE,
  stopwords = function(x) { removeWords(x, stopwords()) },
  removePunctuation = TRUE,
```

```r
  stemming = TRUE
))

# compare the result
sms_dtm
```

```
## <<DocumentTermMatrix (documents: 5559, terms: 6559)>>
## Non-/sparse entries: 42147/36419334
## Sparsity           : 100%
## Maximal term length: 40
## Weighting          : term frequency (tf)
```

```r
sms_dtm2
```

```
## <<DocumentTermMatrix (documents: 5559, terms: 6961)>>
## Non-/sparse entries: 43221/38652978
## Sparsity           : 100%
## Maximal term length: 40
## Weighting          : term frequency (tf)
```

```r
sms_dtm3
```

```
## <<DocumentTermMatrix (documents: 5559, terms: 6559)>>
## Non-/sparse entries: 42147/36419334
## Sparsity           : 100%
## Maximal term length: 40
## Weighting          : term frequency (tf)
```

```r
# creating training and test datasets
sms_dtm_train <- sms_dtm[1:4169, ]
sms_dtm_test  <- sms_dtm[4170:5559, ]

# also save the labels
sms_train_labels <- sms_raw[1:4169, ]$type
sms_test_labels  <- sms_raw[4170:5559, ]$type

# check that the proportion of spam is similar
prop.table(table(sms_train_labels))
```

```
## sms_train_labels
##       ham      spam
## 0.8647158 0.1352842
```

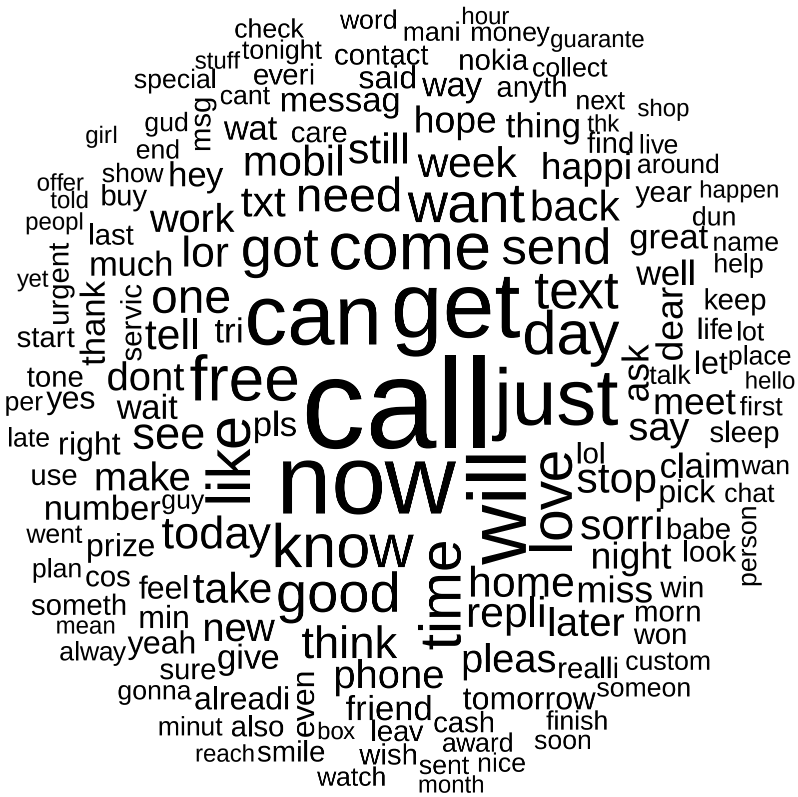
```r
prop.table(table(sms_test_labels))
```

```
## sms_test_labels
##       ham      spam
## 0.8683453 0.1316547
```

```r
# word cloud visualization
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```r
wordcloud(sms_corpus_clean, min.freq = 50, random.order = FALSE)
```

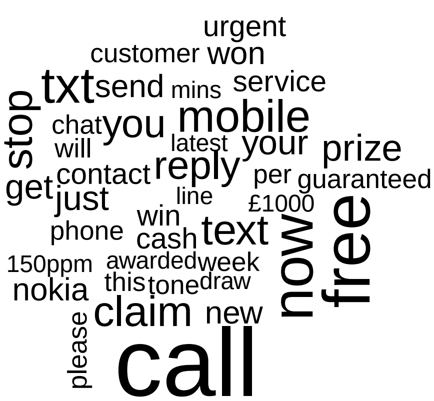



```r
# subset the training data into spam and ham groups
spam <- subset(sms_raw, type == "spam")
ham  <- subset(sms_raw, type == "ham")

wordcloud(spam$text, max.words = 40, scale = c(3, 0.5))
```

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

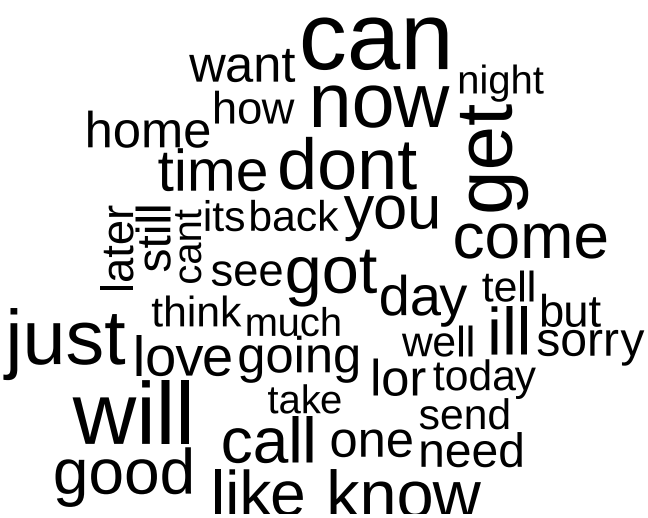
```
## Warning in tm_map.SimpleCorpus(corpus, function(x) tm::removeWords(x,
## tm::stopwords())): transformation drops documents
```




```r
wordcloud(ham$text, max.words = 40, scale = c(3, 0.5))
```

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

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




```r
sms_dtm_freq_train <- removeSparseTerms(sms_dtm_train, 0.999)
sms_dtm_freq_train
```

```
## <<DocumentTermMatrix (documents: 4169, terms: 1104)>>
## Non-/sparse entries: 24827/4577749
## Sparsity           : 99%
## Maximal term length: 19
## Weighting          : term frequency (tf)
```

```r
# indicator features for frequent words
findFreqTerms(sms_dtm_train, 5)
```

```
##    [1] "£wk"                   "€~m"                   "€~s"
##    [4] "abiola"                "abl"                   "abt"
##    [7] "accept"                "access"                "account"
##   [10] "across"                "act"                   "activ"
##   [13] "actual"                "add"                   "address"
##   [16] "admir"                 "adult"                 "advanc"
##   [19] "aft"                   "afternoon"             "age"
##   [22] "ago"                   "aha"                   "ahead"
##   [25] "aight"                 "aint"                  "air"
##   [28] "aiyo"                  "alex"                  "almost"
##   [31] "alon"                  "alreadi"               "alright"
##   [34] "also"                  "alway"                 "angri"
##   [37] "announc"               "anoth"                 "answer"
##   [40] "anymor"                "anyon"                 "anyth"
##   [43] "anytim"                "anyway"                "apart"
##   [46] "app"                   "appli"                 "appreci"
##   [49] "arcad"                 "ard"                   "area"
##   [52] "argu"                  "argument"              "armand"
##   [55] "around"                "arrang"                "arriv"
##   [58] "asap"                  "ask"                   "askd"
##   [61] "attempt"               "auction"               "avail"
##   [64] "ave"                   "avoid"                 "await"
##   [67] "awak"                  "award"                 "away"
##   [70] "awesom"                "babe"                  "babi"
```

```
##   [73] "back"          "bad"           "bag"
##   [76] "bank"          "bare"          "basic"
##   [79] "bath"          "batteri"       "bcoz"
##   [82] "bday"          "beauti"        "becom"
##   [85] "bed"           "bedroom"       "beer"
##   [88] "begin"         "believ"        "best"
##   [91] "better"        "bid"           "big"
##   [94] "bill"          "bird"          "birthday"
##   [97] "bit"           "black"         "blank"
##  [100] "bless"         "blue"          "bluetooth"
##  [103] "bold"          "bonus"         "boo"
##  [106] "book"          "boost"         "bore"
##  [109] "boss"          "bother"        "bout"
##  [112] "box"           "boy"           "boytoy"
##  [115] "break"         "breath"        "bring"
##  [118] "brother"       "bslvyl"        "btnationalr"
##  [121] "buck"          "bus"           "busi"
##  [124] "buy"           "cabin"         "call"
##  [127] "caller"        "callertun"     "camcord"
##  [130] "came"          "camera"        "campus"
##  [133] "can"           "cancel"        "cancer"
##  [136] "cant"          "car"           "card"
##  [139] "care"          "carlo"         "case"
##  [142] "cash"          "cashbal"       "catch"
##  [145] "caus"          "celebr"        "cell"
##  [148] "centr"         "chanc"         "chang"
##  [151] "charg"         "chat"          "cheap"
##  [154] "cheaper"       "check"         "cheer"
##  [157] "chennai"       "chikku"        "childish"
##  [160] "children"      "choic"         "choos"
##  [163] "christma"      "claim"         "class"
##  [166] "clean"         "clear"         "close"
##  [169] "club"          "code"          "coffe"
##  [172] "cold"          "colleagu"      "collect"
##  [175] "colleg"        "colour"        "come"
##  [178] "comin"         "comp"          "compani"
##  [181] "competit"      "complet"       "complimentari"
##  [184] "comput"        "condit"        "confirm"
##  [187] "congrat"       "congratul"     "connect"
##  [190] "contact"       "content"       "contract"
##  [193] "cook"          "cool"          "copi"
##  [196] "correct"       "cos"           "cost"
##  [199] "cost£pm"       "costa"         "coupl"
##  [202] "cours"         "cover"         "coz"
##  [205] "crave"         "crazi"         "creat"
##  [208] "credit"        "cri"           "cross"
##  [211] "cuddl"         "cum"           "cup"
##  [214] "current"       "custcar"       "custom"
##  [217] "cut"           "cute"          "cuz"
##  [220] "dad"           "daddi"         "darl"
##  [223] "darlin"        "darren"        "dat"
##  [226] "date"          "day"           "dead"
##  [229] "deal"          "dear"          "decid"
##  [232] "decim"         "decis"         "deep"
```

```
## [235] "definit"          "del"              "deliv"
## [238] "deliveri"         "den"              "depend"
## [241] "detail"           "didnt"            "die"
## [244] "diet"             "differ"           "difficult"
## [247] "digit"            "din"              "dinner"
## [250] "direct"           "dis"              "discount"
## [253] "discuss"          "disturb"          "dnt"
## [256] "doc"              "doctor"           "doesnt"
## [259] "dog"              "doin"             "don"
## [262] "done"             "dont"             "door"
## [265] "doubl"            "download"         "draw"
## [268] "dream"            "drink"            "drive"
## [271] "drop"             "drug"             "dude"
## [274] "due"              "dun"              "dunno"
## [277] "dvd"              "earli"            "earlier"
## [280] "earth"            "easi"             "eat"
## [283] "eatin"            "egg"              "either"
## [286] "els"              "email"            "embarass"
## [289] "end"              "energi"           "england"
## [292] "enjoy"            "enough"           "enter"
## [295] "entitl"           "entri"            "envelop"
## [298] "etc"              "euro"             "eve"
## [301] "even"             "ever"             "everi"
## [304] "everybodi"        "everyon"          "everyth"
## [307] "exact"            "exam"             "excel"
## [310] "excit"            "excus"            "expect"
## [313] "experi"           "expir"            "extra"
## [316] "eye"              "face"             "facebook"
## [319] "fact"             "fall"             "famili"
## [322] "fanci"            "fantasi"          "fantast"
## [325] "far"              "fast"             "fat"
## [328] "father"           "fault"            "feb"
## [331] "feel"             "felt"             "fetch"
## [334] "fight"            "figur"            "file"
## [337] "fill"             "film"             "final"
## [340] "find"             "fine"             "finger"
## [343] "finish"           "first"            "fix"
## [346] "flag"             "flat"             "flight"
## [349] "flower"           "follow"           "fone"
## [352] "food"             "forev"            "forget"
## [355] "forgot"           "forward"          "found"
## [358] "freak"            "free"             "freemsg"
## [361] "freephon"         "fren"             "fri"
## [364] "friday"           "friend"           "friendship"
## [367] "frm"              "frnd"             "frnds"
## [370] "full"             "fullonsmscom"     "fun"
## [373] "funni"            "futur"            "gal"
## [376] "game"             "gap"              "gas"
## [379] "gave"             "gay"              "gentl"
## [382] "get"              "gettin"           "gift"
## [385] "girl"             "girlfrnd"         "give"
## [388] "glad"             "god"              "goe"
## [391] "goin"             "gone"             "gonna"
## [394] "good"             "goodmorn"         "goodnight"
```

```
## [397] "got"                 "goto"                "gotta"
## [400] "great"               "grin"                "guarante"
## [403] "gud"                 "guess"               "guy"
## [406] "gym"                 "haf"                 "haha"
## [409] "hai"                 "hair"                "half"
## [412] "hand"                "handset"             "hang"
## [415] "happen"              "happi"               "hard"
## [418] "hate"                "hav"                 "havent"
## [421] "head"                "hear"                "heard"
## [424] "heart"               "heavi"               "hee"
## [427] "hell"                "hello"               "help"
## [430] "hey"                 "hgsuiteland"         "hit"
## [433] "hiya"                "hmm"                 "hmmm"
## [436] "hmv"                 "hol"                 "hold"
## [439] "holder"              "holiday"             "home"
## [442] "hook"                "hop"                 "hope"
## [445] "horni"               "hospit"              "hot"
## [448] "hotel"               "hour"                "hous"
## [451] "how"                 "howev"               "howz"
## [454] "hrs"                 "httpwwwurawinnercom" "hug"
## [457] "huh"                 "hungri"              "hurri"
## [460] "hurt"                "ice"                 "idea"
## [463] "identifi"            "ignor"               "ill"
## [466] "immedi"              "import"              "inc"
## [469] "includ"              "india"               "info"
## [472] "inform"              "insid"               "instead"
## [475] "interest"            "invit"               "ipod"
## [478] "irrit"               "ish"                 "island"
## [481] "issu"                "ive"                 "izzit"
## [484] "januari"             "jay"                 "job"
## [487] "john"                "join"                "joke"
## [490] "joy"                 "jst"                 "jus"
## [493] "just"                "juz"                 "kate"
## [496] "keep"                "kept"                "kick"
## [499] "kid"                 "kill"                "kind"
## [502] "kinda"               "king"                "kiss"
## [505] "knew"                "know"                "knw"
## [508] "ladi"                "land"                "landlin"
## [511] "laptop"              "lar"                 "last"
## [514] "late"                "later"               "latest"
## [517] "laugh"               "lazi"                "ldn"
## [520] "lead"                "learn"               "least"
## [523] "leav"                "lect"                "left"
## [526] "leh"                 "lei"                 "less"
## [529] "lesson"              "let"                 "letter"
## [532] "liao"                "librari"             "lie"
## [535] "life"                "lift"                "light"
## [538] "like"                "line"                "link"
## [541] "list"                "listen"              "littl"
## [544] "live"                "lmao"                "load"
## [547] "loan"                "local"               "locat"
## [550] "log"                 "lol"                 "london"
## [553] "long"                "longer"              "look"
## [556] "lookin"              "lor"                 "lose"
```

```
## [559] "lost"          "lot"          "lovabl"
## [562] "love"          "lover"        "loyalti"
## [565] "ltd"           "luck"         "lucki"
## [568] "lunch"         "luv"          "mad"
## [571] "made"          "mah"          "mail"
## [574] "make"          "malaria"      "man"
## [577] "mani"          "march"        "mark"
## [580] "marri"         "match"        "mate"
## [583] "matter"        "maxim"        "maxmin"
## [586] "may"           "mayb"         "meal"
## [589] "mean"          "meant"        "med"
## [592] "medic"         "meet"         "meetin"
## [595] "meh"           "member"       "men"
## [598] "merri"         "messag"       "met"
## [601] "mid"           "midnight"     "might"
## [604] "min"           "mind"         "mine"
## [607] "minut"         "miracl"       "miss"
## [610] "mistak"        "moan"         "mob"
## [613] "mobil"         "mobileupd"    "mode"
## [616] "mom"           "moment"       "mon"
## [619] "monday"        "money"        "month"
## [622] "morn"          "mother"       "motorola"
## [625] "move"          "movi"         "mrng"
## [628] "mrt"           "mrw"          "msg"
## [631] "msgs"          "mths"         "much"
## [634] "mum"           "murder"       "music"
## [637] "must"          "muz"          "nah"
## [640] "nake"          "name"         "nation"
## [643] "natur"         "naughti"      "near"
## [646] "need"          "net"          "network"
## [649] "neva"          "never"        "new"
## [652] "news"          "next"         "nice"
## [655] "nigeria"       "night"        "nite"
## [658] "nobodi"        "noe"          "nokia"
## [661] "noon"          "nope"         "normal"
## [664] "normpton"      "noth"         "notic"
## [667] "now"           "num"          "number"
## [670] "nyt"           "obvious"      "offer"
## [673] "offic"         "offici"       "okay"
## [676] "oki"           "old"          "omg"
## [679] "one"           "onlin"        "onto"
## [682] "oop"           "open"         "oper"
## [685] "opinion"       "opt"          "optout"
## [688] "orang"         "orchard"      "order"
## [691] "oredi"         "oso"          "other"
## [694] "otherwis"      "outsid"       "pack"
## [697] "page"          "paid"         "pain"
## [700] "paper"         "parent"       "park"
## [703] "part"          "parti"        "partner"
## [706] "pass"          "passion"      "password"
## [709] "past"          "pay"          "peopl"
## [712] "per"           "person"       "pete"
## [715] "phone"         "photo"        "pic"
## [718] "pick"          "pictur"       "pin"
```

```
## [721] "piss"             "pix"              "pizza"
## [724] "place"            "plan"             "play"
## [727] "player"           "pleas"            "pleasur"
## [730] "plenti"           "pls"              "plus"
## [733] "plz"              "pmin"             "pmsg"
## [736] "pobox"            "point"            "poli"
## [739] "polic"            "poor"             "pop"
## [742] "possess"          "possibl"          "post"
## [745] "pound"            "power"            "ppm"
## [748] "pray"             "present"          "press"
## [751] "pretti"           "previous"         "price"
## [754] "princess"         "privat"           "prize"
## [757] "prob"             "probabl"          "problem"
## [760] "project"          "promis"           "pub"
## [763] "put"              "qualiti"          "question"
## [766] "quick"            "quit"             "quiz"
## [769] "quot"             "rain"             "random"
## [772] "rang"             "rate"             "rather"
## [775] "rcvd"             "reach"            "read"
## [778] "readi"            "real"             "reali"
## [781] "realli"           "reason"           "receipt"
## [784] "receiv"           "recent"           "record"
## [787] "refer"            "regard"           "regist"
## [790] "relat"            "relax"            "remain"
## [793] "rememb"           "remind"           "remov"
## [796] "rent"             "rental"           "repli"
## [799] "repres"           "request"          "respond"
## [802] "respons"          "rest"             "result"
## [805] "return"           "reveal"           "review"
## [808] "reward"           "right"            "ring"
## [811] "rington"          "rite"             "road"
## [814] "rock"             "role"             "room"
## [817] "roommat"          "rose"             "round"
## [820] "rowwjhl"          "rpli"             "rreveal"
## [823] "run"              "rush"             "sad"
## [826] "sae"              "safe"             "said"
## [829] "sale"             "sat"              "saturday"
## [832] "savamob"          "save"             "saw"
## [835] "say"              "sch"              "school"
## [838] "scream"           "sea"              "search"
## [841] "sec"              "second"           "secret"
## [844] "see"              "seem"             "seen"
## [847] "select"           "self"             "sell"
## [850] "semest"           "send"             "sens"
## [853] "sent"             "serious"          "servic"
## [856] "set"              "settl"            "sex"
## [859] "sexi"             "shall"            "share"
## [862] "shd"              "ship"             "shirt"
## [865] "shop"             "short"            "show"
## [868] "shower"           "sick"             "side"
## [871] "sigh"             "sight"            "sign"
## [874] "silent"           "simpl"            "sinc"
## [877] "singl"            "sipix"            "sir"
## [880] "sis"              "sister"           "sit"
```

```
##  [883] "situat"             "skxh"               "skype"
##  [886] "slave"              "sleep"              "slept"
##  [889] "slow"               "slowli"             "small"
##  [892] "smile"              "smoke"              "sms"
##  [895] "smth"               "snow"               "sofa"
##  [898] "sol"                "somebodi"           "someon"
##  [901] "someth"             "sometim"            "somewher"
##  [904] "song"               "soni"               "sonyericsson"
##  [907] "soon"               "sorri"              "sort"
##  [910] "sound"              "south"              "space"
##  [913] "speak"              "special"            "specialcal"
##  [916] "spend"              "spent"              "spoke"
##  [919] "spree"              "stand"              "start"
##  [922] "statement"          "station"            "stay"
##  [925] "std"                "step"               "still"
##  [928] "stockport"          "stone"              "stop"
##  [931] "store"              "stori"              "street"
##  [934] "student"            "studi"              "stuff"
##  [937] "stupid"             "style"              "sub"
##  [940] "subscrib"           "success"            "suck"
##  [943] "suit"               "summer"             "sun"
##  [946] "sunday"             "sunshin"            "sup"
##  [949] "support"            "suppos"             "sure"
##  [952] "surf"               "surpris"            "sweet"
##  [955] "swing"              "system"             "take"
##  [958] "talk"               "tampa"              "tariff"
##  [961] "tcs"                "tea"                "teach"
##  [964] "tear"               "teas"               "tel"
##  [967] "tell"               "ten"                "tenerif"
##  [970] "term"               "test"               "text"
##  [973] "thank"              "thanx"              "that"
##  [976] "thing"              "think"              "thinkin"
##  [979] "thk"                "tho"                "though"
##  [982] "thought"            "throw"              "thru"
##  [985] "tht"                "thur"               "tick"
##  [988] "ticket"             "til"                "till"
##  [991] "time"               "tire"               "titl"
##  [994] "tmr"                "toclaim"            "today"
##  [997] "togeth"             "told"               "tomo"
## [1000] "tomorrow"           "tone"               "tonight"
## [1003] "tonit"              "took"               "top"
## [1006] "torch"              "tot"                "total"
## [1009] "touch"              "tough"              "tour"
## [1012] "toward"             "town"               "track"
## [1015] "train"              "transact"           "travel"
## [1018] "treat"              "tri"                "trip"
## [1021] "troubl"             "true"               "trust"
## [1024] "truth"              "tscs"               "ttyl"
## [1027] "tuesday"            "turn"               "twice"
## [1030] "two"                "txt"                "txting"
## [1033] "txts"               "type"               "ufind"
## [1036] "ugh"                "ull"                "uncl"
## [1039] "understand"         "unless"             "unlimit"
## [1042] "unredeem"           "unsub"              "unsubscrib"
```

```
## [1045] "updat"              "ure"                 "urgent"
## [1048] "urself"             "use"                 "user"
## [1051] "usf"                "usual"               "uve"
## [1054] "valentin"           "valid"               "valu"
## [1057] "via"                "video"               "vikki"
## [1060] "visit"              "vodafon"             "voic"
## [1063] "vomit"              "voucher"             "wait"
## [1066] "wake"               "walk"                "wan"
## [1069] "wana"               "wanna"               "want"
## [1072] "wap"                "warm"                "wast"
## [1075] "wat"                "watch"               "water"
## [1078] "way"                "weak"                "wear"
## [1081] "weather"            "wed"                 "wednesday"
## [1084] "weed"               "week"                "weekend"
## [1087] "welcom"             "well"                "wen"
## [1090] "went"               "what"                "whatev"
## [1093] "whenev"             "whole"               "wid"
## [1096] "wif"                "wife"                "wil"
## [1099] "will"               "win"                 "wine"
## [1102] "winner"             "wish"                "wit"
## [1105] "within"             "without"             "wiv"
## [1108] "wkli"               "wks"                 "wnt"
## [1111] "woke"               "won"                 "wonder"
## [1114] "wont"               "word"                "work"
## [1117] "workin"             "world"               "worri"
## [1120] "wors"               "worth"               "wot"
## [1123] "wow"                "write"               "wrong"
## [1126] "wwq"                "wwwgetzedcouk"       "xmas"
## [1129] "xxx"                "yahoo"               "yar"
## [1132] "yeah"               "year"                "yep"
## [1135] "yes"                "yesterday"           "yet"
## [1138] "yoga"               "yup"
```

```r
# save frequently-appearing terms to a character vector
sms_freq_words <- findFreqTerms(sms_dtm_train, 5)
str(sms_freq_words)
```

```
##  chr [1:1139] "£wk" "€~m" "€~s" "abiola" "abl" "abt" "accept" "access" ...
```

```r
# create DTMs with only the frequent terms
sms_dtm_freq_train <- sms_dtm_train[ , sms_freq_words]
sms_dtm_freq_test <- sms_dtm_test[ , sms_freq_words]

# convert counts to a factor
convert_counts <- function(x) {
  x <- ifelse(x > 0, "Yes", "No")
}

# apply() convert_counts() to columns of train/test data
sms_train <- apply(sms_dtm_freq_train, MARGIN = 2, convert_counts)
sms_test  <- apply(sms_dtm_freq_test, MARGIN = 2, convert_counts)
```

Step 3: Training a model on the data —-

```
library(e1071)
sms_classifier <- naiveBayes(sms_train, sms_train_labels)
```

Step 4: Evaluating model performance —-

```
sms_test_pred <- predict(sms_classifier, sms_test)

head(sms_test_pred)
```

```
## [1] ham  ham  ham  ham  spam ham
## Levels: ham spam
```

```
library(gmodels)
CrossTable(sms_test_pred, sms_test_labels,
           prop.chisq = FALSE, prop.t = FALSE, prop.r = FALSE,
           dnn = c('predicted', 'actual'))
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  1390
##
##
##              | actual
##    predicted |       ham |      spam | Row Total |
## -------------|-----------|-----------|-----------|
##          ham |      1201 |        30 |      1231 |
##              |     0.995 |     0.164 |           |
## -------------|-----------|-----------|-----------|
##         spam |         6 |       153 |       159 |
##              |     0.005 |     0.836 |           |
## -------------|-----------|-----------|-----------|
## Column Total |      1207 |       183 |      1390 |
##              |     0.868 |     0.132 |           |
## -------------|-----------|-----------|-----------|
##
##
```

Step 5: Improving model performance —-

```
sms_classifier2 <- naiveBayes(sms_train, sms_train_labels, laplace = 1)
sms_test_pred2 <- predict(sms_classifier2, sms_test)
CrossTable(sms_test_pred2, sms_test_labels,
```

```r
           prop.chisq = FALSE, prop.t = FALSE, prop.r = FALSE,
           dnn = c('predicted', 'actual'))
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Col Total |
## |-------------------------|
##
##
## Total Observations in Table:  1390
##
##
##              | actual
##    predicted |       ham |      spam | Row Total |
## -------------|-----------|-----------|-----------|
##          ham |      1202 |        28 |      1230 |
##              |     0.996 |     0.153 |           |
## -------------|-----------|-----------|-----------|
##         spam |         5 |       155 |       160 |
##              |     0.004 |     0.847 |           |
## -------------|-----------|-----------|-----------|
## Column Total |      1207 |       183 |      1390 |
##              |     0.868 |     0.132 |           |
## -------------|-----------|-----------|-----------|
##
##
```