Code for QSS Chapter 5: Discovery

Kosuke Imai

First Printing

Section 5.1: Textual Data

Section 5.1.1: The Disputed Authorship of 'The Federalist Papers'

```
## load two required libraries
library(tm, SnowballC)
```

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

```
## load the raw corpus
corpus.raw <- VCorpus(DirSource(directory = "federalist", pattern = "fp"))
corpus.raw
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 0
## Content:  documents: 85
```

```
## make lower case
corpus.prep <- tm_map(corpus.raw, content_transformer(tolower))
## remove white space
corpus.prep <- tm_map(corpus.prep, stripWhitespace)
## remove punctuation
corpus.prep <- tm_map(corpus.prep, removePunctuation)

## remove numbers
corpus.prep <- tm_map(corpus.prep, removeNumbers)

head(stopwords("english"))
```

```
## [1] "i"      "me"     "my"     "myself" "we"     "our"
```

```
## remove stop words
corpus <- tm_map(corpus.prep, removeWords, stopwords("english"))

## finally stem remaining words
corpus <- tm_map(corpus, stemDocument)

## the output is truncated here to save space
content(corpus[[10]]) # Essay No. 10
```

```
##   [1] "among numer advantag promis wellconstruct union none"
##   [2] "deserv accur develop tendenc break"
##   [3] "control violenc faction friend popular govern never"
##   [4] "find much alarm charact fate"
##   [5] "contempl propens danger vice will fail"
##   [6] "therefor set due valu plan without violat"
##   [7] "principl attach provid proper cure"
##   [8] "instabl injustic confus introduc public council"
##   [9] "truth mortal diseas popular govern"
##  [10] "everywher perish continu favorit fruit"
##  [11] "topic adversari liberti deriv specious"
##  [12] "declam valuabl improv made american constitut"
##  [13] "popular model ancient modern certain"
##  [14] "much admir unwarrant partial contend"
##  [15] "effectu obviat danger side"
##  [16] "wish expect complaint everywher heard consider"
##  [17] "virtuous citizen equal friend public privat faith"
##  [18] "public person liberti govern unstabl"
##  [19] "public good disregard conflict rival parti"
##  [20] "measur often decid accord rule"
##  [21] "justic right minor parti superior forc"
##  [22] "interest overbear major howev anxious may wish"
##  [23] "complaint foundat evid known fact"
##  [24] "will permit us deni degre true will"
##  [25] "found inde candid review situat"
##  [26] "distress labor erron charg oper"
##  [27] "govern will found time"
##  [28] "caus will alon account mani heaviest misfortun"
##  [29] "particular prevail increas distrust public engag"
##  [30] "alarm privat right echo one end contin"
##  [31] "must chiefli wholli effect unsteadi"
##  [32] "injustic factious spirit taint public administr faction understand number citizen whether amou
##  [33] "major minor whole unit actuat"
##  [34] "common impuls passion interest advers right"
##  [35] "citizen perman aggreg interest"
##  [36] "communiti two method cure mischief faction one"
##  [37] "remov caus control effect two method remov caus faction one"
##  [38] "destroy liberti essenti exist"
##  [39] "give everi citizen opinion passion"
##  [40] "interest never truli said first remedi"
##  [41] "wors diseas liberti faction air fire"
##  [42] "aliment without instant expir less folli"
##  [43] "abolish liberti essenti polit life nourish"
##  [44] "faction wish annihil air essenti"
##  [45] "anim life impart fire destruct agenc second expedi impractic first unwis"
##  [46] "long reason man continu fallibl liberti"
##  [47] "exercis differ opinion will form long connect"
##  [48] "subsist reason selflov opinion passion"
##  [49] "will reciproc influenc former will"
##  [50] "object latter will attach divers"
##  [51] "faculti men right properti origin"
##  [52] "less insuper obstacl uniform interest protect"
##  [53] "faculti first object govern protect"
##  [54] "differ unequ faculti acquir properti possess"
##  [55] "differ degre kind properti immedi result"
##  [56] "influenc sentiment view respect proprietor"
##  [57] "ensu divis societi differ interest parti latent caus faction thus sown natur man"
##  [58] "see everywher brought differ degre activ accord"
##  [59] "differ circumst civil societi zeal differ"
##  [60] "opinion concern religion concern govern mani point"
##  [61] "well specul practic attach differ leader"
##  [62] "ambiti contend preemin power person"
##  [63] "descript whose fortun interest human passion"
##  [64] "turn divid mankind parti inflam mutual"
##  [65] "animos render much dispos vex oppress"
##  [66] "cooper common good strong propens"
##  [67] "mankind fall mutual animos substanti"
##  [68] "occas present frivol fanci distinct"
##  [69] "suffici kindl unfriend passion excit"
##  [70] "violent conflict common durabl sourc faction"
##  [71] "various unequ distribut properti hold"
##  [72] "without properti ever form distinct interest"
##  [73] "societi creditor debtor fall"
##  [74] "like discrimin land interest manufactur interest"
##  [75] "mercantil interest money interest mani lesser interest grow"
##  [76] "necess civil nation divid differ class"
##  [77] "actuat differ sentiment view regul various"
##  [78] "interf interest form princip task modern legisl"
##  [79] "involv spirit parti faction necessari ordinari"
##  [80] "oper govern man allow judg caus interest"
##  [81] "certain bias judgment improb corrupt integr"
##  [82] "equal nay greater reason bodi men unfit"
##  [83] "judg parti time yet mani import"
##  [84] "act legisl mani judici determin inde concern"
##  [85] "right singl person concern right larg bodi"
##  [86] "citizen differ class legisl advoc"
##  [87] "parti caus determin law propos concern"
##  [88] "privat debt question creditor parti"
##  [89] "one side debtor justic hold balanc"
##  [90] "yet parti must judg"
##  [91] "numer parti word power faction"
##  [92] "must expect prevail shall domest manufactur encourag"
##  [93] "degre restrict foreign manufactur question"
##  [94] "differ decid land manufactur"
##  [95] "class probabl neither sole regard justic"
##  [96] "public good apportion tax various descript"
##  [97] "properti act seem requir exact imparti"
##  [98] "yet perhap legisl act greater opportun"
##  [99] "temptat given predomin parti trampl rule"
## [100] "justic everi shill overburden inferior number"
## [101] "shill save pocket vain say enlighten statesmen will abl adjust"
## [102] "clash interest render subservi public"
## [103] "good enlighten statesmen will alway helm mani"
## [104] "case can adjust made without take view"
## [105] "indirect remot consider will rare prevail"
## [106] "immedi interest one parti may find disregard right"
## [107] "anoth good whole infer brought caus faction"
## [108] "remov relief sought mean"
## [109] "control effect faction consist less major relief suppli"
## [110] "republican principl enabl major defeat sinist"
## [111] "view regular vote may clog administr may convuls"
## [112] "societi will unabl execut mask violenc"
## [113] "form constitut major includ faction"
## [114] "form popular govern hand enabl sacrific"
## [115] "rule passion interest public good right"
## [116] "citizen secur public good privat right"
## [117] "danger faction time preserv spirit"
## [118] "form popular govern great object"
## [119] "inquiri direct let add great desideratum"
## [120] "form govern can rescu opprobrium"
## [121] "long labor recommend esteem adopt"
## [122] "mankind mean object attain evid one two"
## [123] "either exist passion interest major"
## [124] "time must prevent major coexist"
## [125] "passion interest must render number local situat"
## [126] "unabl concert carri effect scheme oppress"
## [127] "impuls opportun suffer coincid well know"
## [128] "neither moral religi motiv can reli adequ control"
## [129] "found injustic violenc individu"
## [130] "lose efficaci proport number combin togeth"
## [131] "proport efficaci becom need view subject may conclud pure democraci"
## [132] "mean societi consist small number citizen"
## [133] "assembl administ govern person can admit cure"
## [134] "mischief faction common passion interest will almost"
## [135] "everi case felt major whole communic concert"
## [136] "result form govern noth check"
## [137] "induc sacrific weaker parti obnoxi individu"
## [138] "henc democraci ever spectacl turbul"
## [139] "content ever found incompat person secur"
## [140] "right properti general short"
## [141] "live violent death theoret politician"
## [142] "patron speci govern erron suppos"
## [143] "reduc mankind perfect equal polit right"
## [144] "time perfect equal assimil"
## [145] "possess opinion passion republ mean govern scheme represent"
## [146] "take place open differ prospect promis cure"
## [147] "seek let us examin point vari pure"
## [148] "democraci shall comprehend natur cure"
## [149] "efficaci must deriv union two great point differ democraci republ"
## [150] "first deleg govern latter small"
## [151] "number citizen elect rest second greater number"
## [152] "citizen greater sphere countri latter may"
## [153] "extend effect first differ one hand refin"
## [154] "enlarg public view pass medium chosen"
## [155] "bodi citizen whose wisdom may best discern true interest"
## [156] "countri whose patriot love justic will least like"
## [157] "sacrific temporari partial consider regul"
## [158] "may well happen public voic pronounc repres"
## [159] "peopl will conson public good pronounc"
## [160] "peopl conven purpos hand"
## [161] "effect may invert men factious temper local prejudic"
## [162] "sinist design may intrigu corrupt mean"
## [163] "first obtain suffrag betray interest peopl"
## [164] "question result whether small extens republ"
## [165] "favor elect proper guardian public weal"
## [166] "clear decid favor latter two obvious consider first place remark howev small republ"
## [167] "may repres must rais certain number order"
## [168] "guard cabal howev larg may"
## [169] "must limit certain number order guard"
## [170] "confus multitud henc number repres"
## [171] "two case proport two constitu"
## [172] "proport greater small republ follow"
## [173] "proport fit charact less larg"
## [174] "small republ former will present greater option consequ"
## [175] "greater probabl fit choic next place repres will chosen greater"
## [176] "number citizen larg small republ will"
## [177] "difficult unworthi candid practic success vicious"
## [178] "art elect often carri suffrag"
## [179] "peopl free will like centr men possess"
## [180] "attract merit diffus establish charact must confess case"
## [181] "mean side inconveni will found lie enlarg"
## [182] "much number elector render repres littl"
## [183] "acquaint local circumst lesser interest"
## [184] "reduc much render unduli attach"
## [185] "littl fit comprehend pursu great nation object"
## [186] "feder constitut form happi combin respect great"
## [187] "aggreg interest refer nation local"
## [188] "particular state legislatur point differ greater number citizen"
## [189] "extent territori may brought within compass republican"
## [190] "democrat govern circumst princip"
## [191] "render factious combin less dread former"
## [192] "latter smaller societi fewer probabl will"
## [193] "distinct parti interest compos fewer distinct parti"
## [194] "interest frequent will major found"
## [195] "parti smaller number individu compos major"
## [196] "smaller compass within place easili"
## [197] "will concert execut plan oppress extend sphere"
## [198] "take greater varieti parti interest make"
## [199] "less probabl major whole will common motiv"
## [200] "invad right citizen common motiv exist"
## [201] "will difficult feel discov strength"
## [202] "act unison besid impedi may"
## [203] "remark conscious unjust dishonor"
## [204] "purpos communic alway check distrust proport"
## [205] "number whose concurr necessari henc clear appear advantag republ"
## [206] "democraci control effect faction enjoy"
## [207] "larg small republici enjoy union state"
## [208] "compos advantag consist substitut repres"
## [209] "whose enlighten view virtuous sentiment render superior"
## [210] "local prejudic scheme injustic will deni"
## [211] "represent union will like possess requisit"
## [212] "endow consist greater secur afford greater"
## [213] "varieti parti event one parti abl outnumb"
## [214] "oppress rest equal degre increas varieti"
## [215] "parti compris within union increas secur"
## [216] "fine consist greater obstacl oppos concert accomplish"
## [217] "secret wish unjust interest major"
## [218] "extent union give palpabl advantag influenc factious leader may kindl flame within particular
## [219] "state will unabl spread general conflagr"
## [220] "state religi sect may degener polit faction"
## [221] "part confederaci varieti sect dispers"
## [222] "entir face must secur nation council danger"
## [223] "sourc rage paper money abolit debt"
## [224] "equal divis properti improp wick project"
## [225] "will less apt pervad whole bodi union particular"
## [226] "member proport maladi like"
## [227] "taint particular counti district entir state extent proper structur union therefor behold"
## [228] "republican remedi diseas incid republican govern"
## [229] "accord degre pleasur pride feel republican"
## [230] "zeal cherish spirit support charact"
## [231] "federalist"
```

```
### Section 5.1.2: Document-Term Matrix

dtm <- DocumentTermMatrix(corpus)
dtm
```

```
## <<DocumentTermMatrix (documents: 85, terms: 4849)>>
## Non-/sparse entries: 44917/367248
## Sparsity           : 89%
## Maximal term length: 18
## Weighting          : term frequency (tf)
```

```
inspect(dtm[1:5, 1:8])
```

```
## <<DocumentTermMatrix (documents: 5, terms: 8)>>
## Non-/sparse entries: 4/36
## Sparsity           : 90%
## Maximal term length: 7
## Weighting          : term frequency (tf)
## Sample             :
##           Terms
## Docs       abandon abat abb abet abhorr abil abject abl
##   fp01.txt       0    0   0    0      0    0      0   1
##   fp02.txt       0    0   0    0      0    1      0   0
##   fp03.txt       0    0   0    0      0    0      0   2
##   fp04.txt       0    0   0    0      0    0      0   1
##   fp05.txt       0    0   0    0      0    0      0   0
```

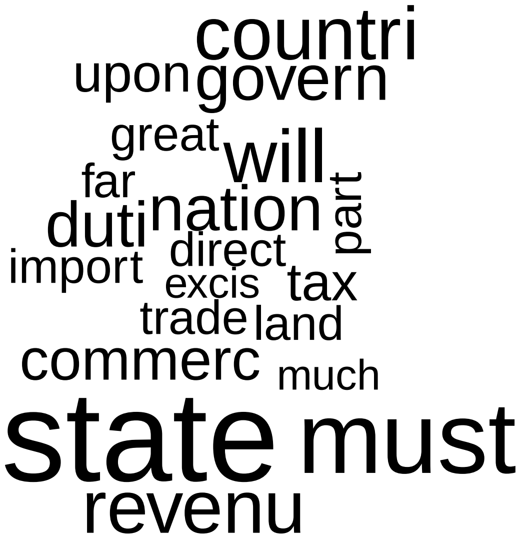
```
dtm.mat <- as.matrix(dtm)
```

Section 5.1.3: Topic Discovery

```
library(wordcloud)
```

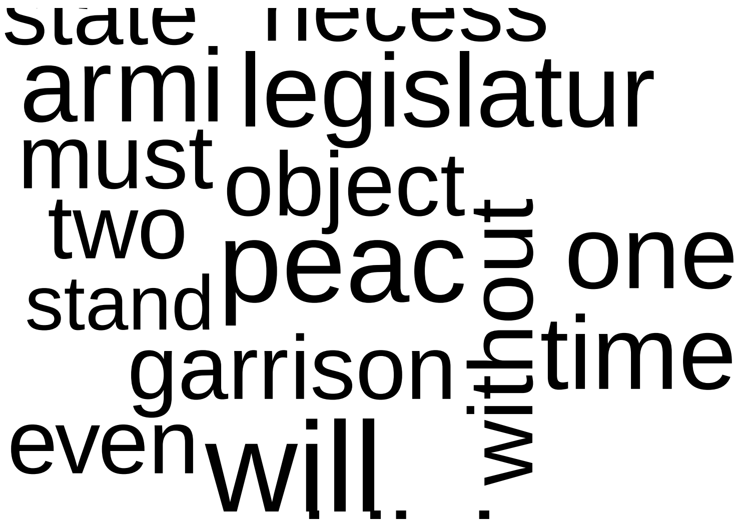
```
## Loading required package: RColorBrewer
```

```
wordcloud(colnames(dtm.mat), dtm.mat[12, ], max.words = 20)  # essay No. 12
```



```
wordcloud(colnames(dtm.mat), dtm.mat[24, ], max.words = 20)  # essay No. 24
```

```
## Warning in wordcloud(colnames(dtm.mat), dtm.mat[24, ], max.words = 20): upon
## could not be fit on page. It will not be plotted.
```



```
stemCompletion(c("revenu", "commerc", "peac", "army"), corpus.prep)
```

```
##     revenu    commerc       peac       army
##  "revenue" "commerce"    "peace"     "army"
```

```
dtm.tfidf <- weightTfIdf(dtm) # tf-idf calculation

dtm.tfidf.mat <- as.matrix(dtm.tfidf)  # convert to matrix

## 10 most important words for Paper No. 12
head(sort(dtm.tfidf.mat[12, ], decreasing = TRUE), n = 10)
```

```
##     revenu contraband     patrol      excis      coast      trade        per
## 0.01905877 0.01886965 0.01886965 0.01876560 0.01592559 0.01473504 0.01420342
##        tax       cent     gallon
## 0.01295466 0.01257977 0.01257977
```

```r
## 10 most important words for Paper No. 24
head(sort(dtm.tfidf.mat[24, ], decreasing = TRUE), n = 10)
```

```
##   garrison   dockyard settlement      spain       armi   frontier    arsenal
## 0.02965511 0.01962294 0.01962294 0.01649040 0.01544256 0.01482756 0.01308196
##    western       post     nearer
## 0.01306664 0.01236780 0.01166730
```

```r
k <- 4  # number of clusters
## subset The Federalist papers written by Hamilton
hamilton <- c(1, 6:9, 11:13, 15:17, 21:36, 59:61, 65:85)
dtm.tfidf.hamilton <- dtm.tfidf.mat[hamilton, ]

## run k-means
km.out <- kmeans(dtm.tfidf.hamilton, centers = k)
km.out$iter # check the convergence; number of iterations may vary
```

```
## [1] 3
```

```r
## label each centroid with the corresponding term
colnames(km.out$centers) <- colnames(dtm.tfidf.hamilton)

for (i in 1:k) { # loop for each cluster
    cat("CLUSTER", i, "\n")
    cat("Top 10 words:\n") # 10 most important terms at the centroid
    print(head(sort(km.out$centers[i, ], decreasing = TRUE), n = 10))
    cat("\n")
    cat("Federalist Papers classified: \n") # extract essays classified
    print(rownames(dtm.tfidf.hamilton)[km.out$cluster == i])
    cat("\n")
}
```

```
## CLUSTER 1
## Top 10 words:
##     pardon    treason      guilt    clemenc     conniv      crime      impun
## 0.04472060 0.02894567 0.02510566 0.02367348 0.02367348 0.01929712 0.01788824
##      plead      sedit       weak
## 0.01673710 0.01492075 0.01470109
##
## Federalist Papers classified:
## [1] "fp74.txt"
##
## CLUSTER 2
## Top 10 words:
##        armi     militia    militari       navig   disciplin         war
## 0.011624485 0.011450433 0.008761049 0.005321748 0.004948897 0.004854514
##        peac    northern    frontier confederaci
## 0.004668017 0.004661314 0.004559462 0.004540867
##
## Federalist Papers classified:
## [1] "fp06.txt" "fp08.txt" "fp11.txt" "fp13.txt" "fp24.txt" "fp25.txt" "fp26.txt"
## [8] "fp28.txt" "fp29.txt"
##
## CLUSTER 3
## Top 10 words:
```

```
##       senat      presid    governor     appoint       nomin      vacanc
## 0.019382349 0.015789668 0.009857989 0.009838966 0.009551661 0.009328505
##       offic     impeach        fill      treati
## 0.007941282 0.006589793 0.006552566 0.006460916
##
## Federalist Papers classified:
## [1] "fp66.txt" "fp67.txt" "fp68.txt" "fp69.txt" "fp75.txt" "fp76.txt" "fp77.txt"
## [8] "fp79.txt"
##
## CLUSTER 4
## Top 10 words:
##       court        upon        juri         tax       taxat        land
## 0.007567168 0.004042350 0.003898096 0.003515715 0.003193611 0.003161991
##   jurisdict      revenu       claus      exclus
## 0.003136312 0.002902347 0.002880987 0.002505574
##
## Federalist Papers classified:
##  [1] "fp01.txt" "fp07.txt" "fp09.txt" "fp12.txt" "fp15.txt" "fp16.txt"
##  [7] "fp17.txt" "fp21.txt" "fp22.txt" "fp23.txt" "fp27.txt" "fp30.txt"
## [13] "fp31.txt" "fp32.txt" "fp33.txt" "fp34.txt" "fp35.txt" "fp36.txt"
## [19] "fp59.txt" "fp60.txt" "fp61.txt" "fp65.txt" "fp70.txt" "fp71.txt"
## [25] "fp72.txt" "fp73.txt" "fp78.txt" "fp80.txt" "fp81.txt" "fp82.txt"
## [31] "fp83.txt" "fp84.txt" "fp85.txt"
```

Section 5.1.4: Authorship Prediction

```
## document-term matrix converted to matrix for manipulation
dtm1 <- as.matrix(DocumentTermMatrix(corpus.prep))
tfm <- dtm1 / rowSums(dtm1) * 1000 # term frequency per 1000 words

## words of interest
words <- c("although", "always", "commonly", "consequently",
           "considerable", "enough", "there", "upon", "while", "whilst")

## select only these words
tfm <- tfm[, words]

## essays written by Madison: `hamilton' defined earlier
madison <- c(10, 14, 37:48, 58)

## average among Hamilton/Madison essays
tfm.ave <- rbind(colSums(tfm[hamilton, ]) / length(hamilton),
                 colSums(tfm[madison, ]) / length(madison))
tfm.ave
```

```
##        although    always  commonly consequently considerable    enough
## [1,] 0.01756975 0.7527744 0.2630876   0.02600857    0.5435127 0.3955031
## [2,] 0.27058809 0.2006710 0.0000000   0.44878468    0.1601669 0.0000000
##         there      upon     while      whilst
## [1,] 4.417750 4.3986828 0.3700484 0.007055719
## [2,] 1.113252 0.2000269 0.0000000 0.380113114
```

```
author <- rep(NA, nrow(dtm1)) # a vector with missing values
author[hamilton] <- 1  # 1 if Hamilton
```

```
author[madison] <- -1  # -1 if Madison

## data frame for regression
author.data <- data.frame(author = author[c(hamilton, madison)],
                          tfm[c(hamilton, madison), ])

hm.fit <- lm(author ~ upon + there + consequently + whilst,
             data = author.data)
hm.fit
```

```
##
## Call:
## lm(formula = author ~ upon + there + consequently + whilst, data = author.data)
##
## Coefficients:
##  (Intercept)          upon         there  consequently        whilst
##     -0.26288       0.16678       0.09494      -0.44012      -0.65875
```

```
hm.fitted <- fitted(hm.fit) # fitted values
sd(hm.fitted)
```

```
## [1] 0.7180769
```

Section 5.1.5: Cross-Validation

```
## proportion of correctly classified essays by Hamilton
mean(hm.fitted[author.data$author == 1] > 0)
```

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

```
## proportion of correctly classified essays by Madison
mean(hm.fitted[author.data$author == -1] < 0)
```

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

```
n <- nrow(author.data)
hm.classify <- rep(NA, n) # a container vector with missing values

for (i in 1:n) {
    ## fit the model to the data after removing the ith observation
    sub.fit <- lm(author ~ upon + there + consequently + whilst,
                  data = author.data[-i, ]) # exclude ith row
    ## predict the authorship for the ith observation
    hm.classify[i] <- predict(sub.fit, newdata = author.data[i, ])
}

## proportion of correctly classified essays by Hamilton
mean(hm.classify[author.data$author == 1] > 0)
```

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

```
## proportion of correctly classified essays by Madison
mean(hm.classify[author.data$author == -1] < 0)
```

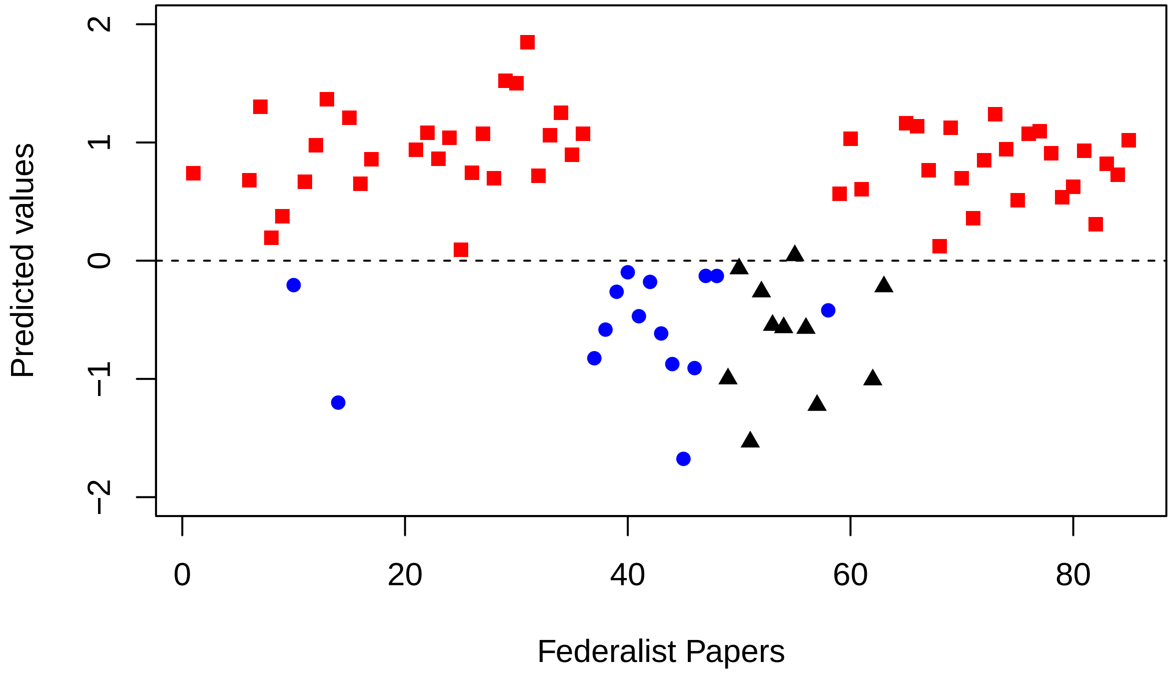
```
## [1] 1
```

```
disputed <- c(49, 50:57, 62, 63) # 11 essays with disputed authorship
tf.disputed <- as.data.frame(tfm[disputed, ])
```

```
## prediction of disputed authorship
pred <- predict(hm.fit, newdata = tf.disputed)
pred # predicted values
```

```
##     fp49.txt    fp50.txt    fp51.txt    fp52.txt    fp53.txt    fp54.txt
## -0.99831799 -0.06759254 -1.53243206 -0.26288400 -0.54584900 -0.56566555
##     fp55.txt    fp56.txt    fp57.txt    fp62.txt    fp63.txt
##  0.04376632 -0.57115610 -1.22289415 -1.00675456 -0.21939646
```

```
## fitted values for essays authored by Hamilton; red squares
plot(hamilton, hm.fitted[author.data$author == 1], pch = 15,
     xlim = c(1, 85), ylim  = c(-2, 2), col = "red",
     xlab = "Federalist Papers", ylab = "Predicted values")
abline(h = 0, lty = "dashed")

## essays authored by Madison; blue circles
points(madison, hm.fitted[author.data$author == -1],
       pch = 16, col = "blue")

## disputed authorship; black triangles
points(disputed, pred, pch = 17)
```



Section 5.2: Network Data

Section 5.2.1: Marriage Network in Renaissance Florence

```
## the first column "FAMILY" of the CSV file represents row names
florence <- read.csv("florentine.csv", row.names = "FAMILY")
florence <- as.matrix(florence) # coerce into a matrix

## print out the adjacency (sub)matrix for the first 5 families
```

```
florence[1:5, 1:5]
```

```
##           ACCIAIUOL ALBIZZI BARBADORI BISCHERI CASTELLAN
## ACCIAIUOL         0       0         0        0         0
## ALBIZZI           0       0         0        0         0
## BARBADORI         0       0         0        0         1
## BISCHERI          0       0         0        0         0
## CASTELLAN         0       0         1        0         0
```

```
rowSums(florence)
```

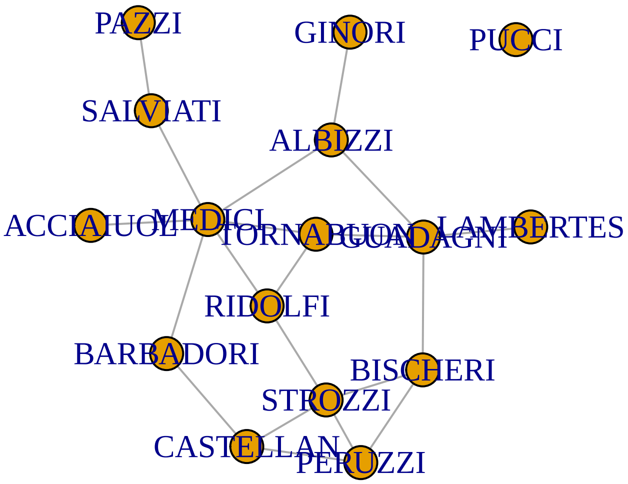
```
## ACCIAIUOL   ALBIZZI BARBADORI  BISCHERI CASTELLAN    GINORI  GUADAGNI LAMBERTES 
##         1         3         2         3         3         1         4         1 
##    MEDICI     PAZZI   PERUZZI     PUCCI   RIDOLFI  SALVIATI   STROZZI TORNABUON 
##         6         1         3         0         3         2         4         3
```

Section 5.2.2: Undirected Graph and Centrality Measures

```
library("igraph")  # load the package
```

```
## 
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
## 
##     decompose, spectrum

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

```
florence <- graph.adjacency(florence, mode = "undirected", diag = FALSE)

plot(florence) # plot the graph
```



```
degree(florence)
```

```
## ACCIAIUOL   ALBIZZI BARBADORI  BISCHERI CASTELLAN    GINORI  GUADAGNI LAMBERTES 
##         1         3         2         3         3         1         4         1 
##    MEDICI     PAZZI   PERUZZI     PUCCI   RIDOLFI  SALVIATI   STROZZI TORNABUON 
##         6         1         3         0         3         2         4         3
```

```
closeness(florence)
```

```
##  ACCIAIUOL    ALBIZZI  BARBADORI   BISCHERI  CASTELLAN     GINORI   GUADAGNI
## 0.02631579 0.03448276 0.03125000 0.02857143 0.02777778 0.02380952 0.03333333
##  LAMBERTES     MEDICI      PAZZI    PERUZZI      PUCCI    RIDOLFI   SALVIATI
## 0.02325581 0.04000000 0.02040816 0.02631579        NaN 0.03571429 0.02777778
##    STROZZI  TORNABUON
## 0.03125000 0.03448276
```

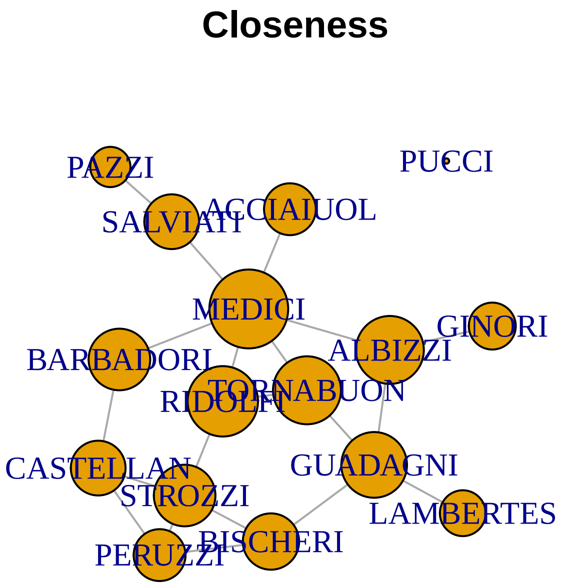
```
1 / (closeness(florence) * 15)
```

```
## ACCIAIUOL   ALBIZZI BARBADORI  BISCHERI CASTELLAN    GINORI  GUADAGNI LAMBERTES
##  2.533333  1.933333  2.133333  2.333333  2.400000  2.800000  2.000000  2.866667
##    MEDICI     PAZZI   PERUZZI     PUCCI   RIDOLFI  SALVIATI   STROZZI TORNABUON
##  1.666667  3.266667  2.533333       NaN  1.866667  2.400000  2.133333  1.933333
```

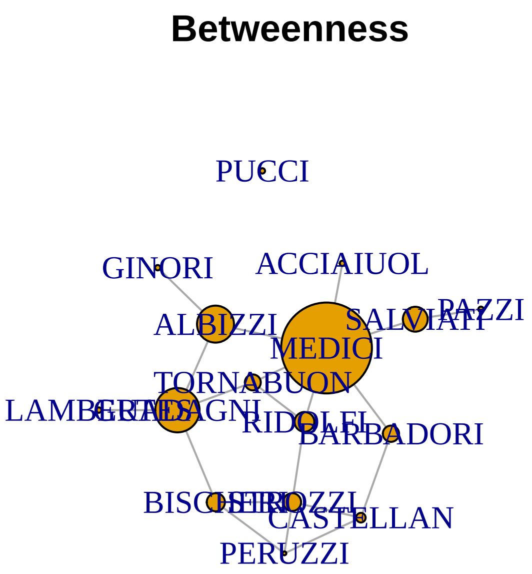
```
betweenness(florence)
```

```
## ACCIAIUOL   ALBIZZI BARBADORI  BISCHERI CASTELLAN    GINORI  GUADAGNI LAMBERTES
##  0.000000 19.333333  8.500000  9.500000  5.000000  0.000000 23.166667  0.000000
##    MEDICI     PAZZI   PERUZZI     PUCCI   RIDOLFI  SALVIATI   STROZZI TORNABUON
## 47.500000  0.000000  2.000000  0.000000 10.333333 13.000000  9.333333  8.333333
```

```
close <- closeness(florence)
close["PUCCI"] <- 0
plot(florence, vertex.size = close * 1000,
     main = "Closeness")
```



```
plot(florence, vertex.size = betweenness(florence),
     main = "Betweenness")
```



Section 5.2.3: Twitter-Following Network

```
twitter <- read.csv("twitter-following.csv", stringsAsFactors = FALSE)
senator <- read.csv("twitter-senator.csv", stringsAsFactors = FALSE)

n <- nrow(senator) # number of senators

## initialize adjacency matrix
twitter.adj <- matrix(0, nrow = n, ncol = n)

## assign screen names to rows and columns
colnames(twitter.adj) <- rownames(twitter.adj) <- senator$screen_name

## change `0' to `1' when edge goes from node `i' to node `j'
for (i in 1:nrow(twitter)) {
    twitter.adj[twitter$following[i], twitter$followed[i]] <- 1
}

twitter.adj <- graph.adjacency(twitter.adj, mode = "directed", diag = FALSE)
```

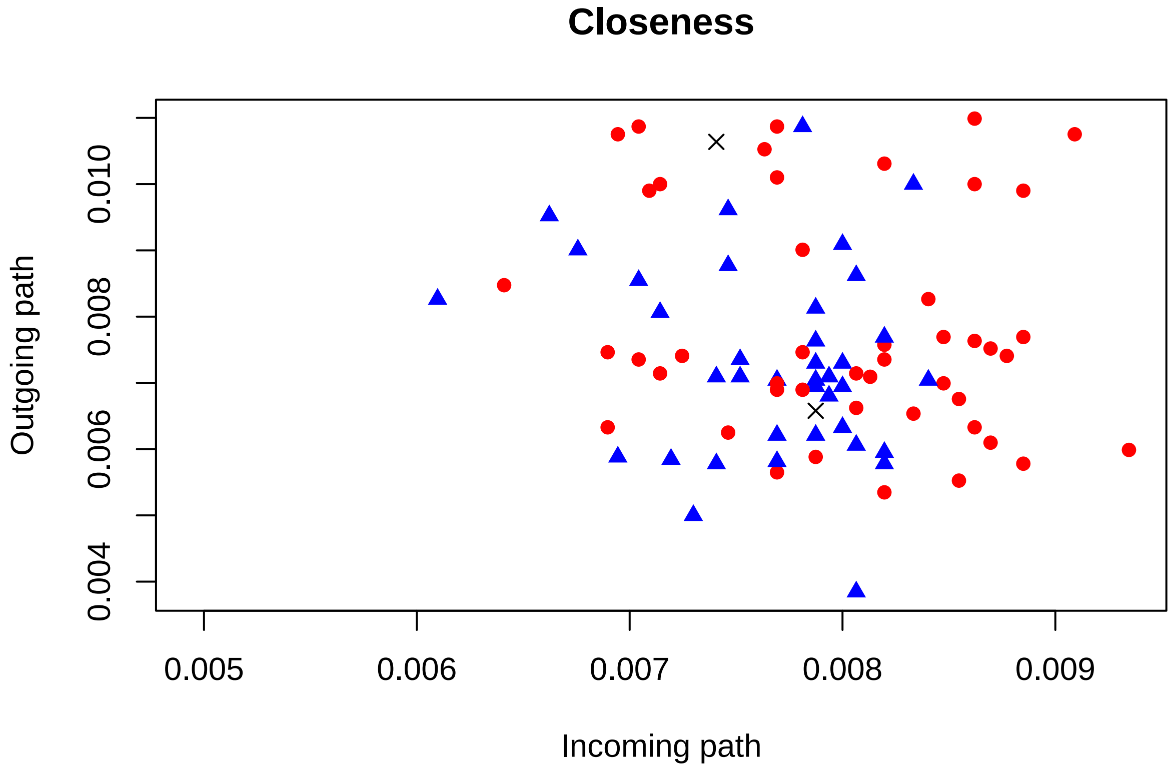
Section 5.2.4: Directed Graph and Centrality

```
senator$indegree <- degree(twitter.adj, mode = "in")
senator$outdegree <- degree(twitter.adj, mode = "out")

in.order <- order(senator$indegree, decreasing = TRUE)
out.order <- order(senator$outdegree, decreasing = TRUE)

## 3 greatest indegree
senator[in.order[1:3], ]
```

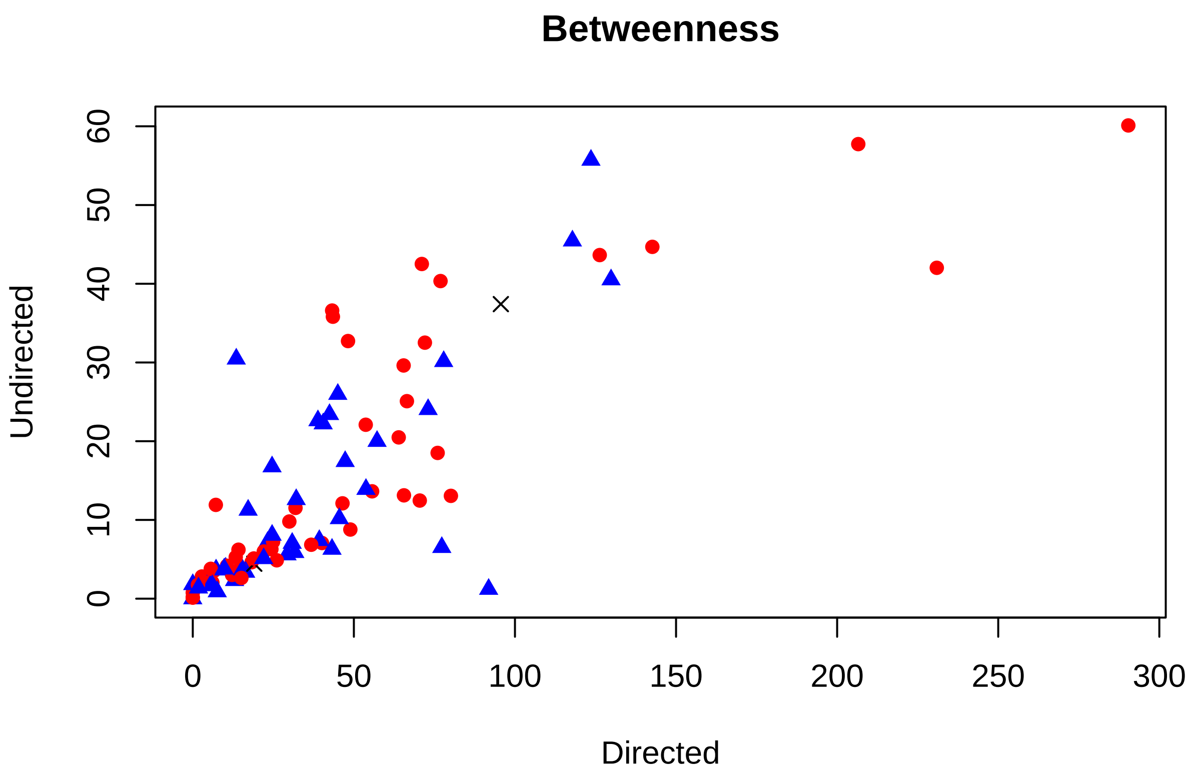
```
##       screen_name             name party state indegree outdegree
## 51   SenJohnMcCain      John McCain     R    AZ       64        15
## 57   lisamurkowski   Lisa Murkowski     R    AK       60        87
## 18 SenatorCollins Susan M. Collins     R    ME       58        79
## 3 greatest outdegree
senator[out.order[1:3], ]
```

```
##        screen_name                name party state indegree outdegree
## 37  SenDeanHeller          Dean Heller     R    NV       55        89
## 21    SenBobCasey Robert P. Casey, Jr.     D    PA       43        88
## 65 sendavidperdue         David Perdue     R    GA       30        88
```

```
n <- nrow(senator)
## color: Democrats = `blue', Republicans = `red', Independent = `black'
col <- rep("red", n)
col[senator$party == "D"] <- "blue"
col[senator$party == "I"] <- "black"

## pch: Democrats = circle, Republicans = diamond, Independent = cross
pch <- rep(16, n)
pch[senator$party == "D"] <- 17
pch[senator$party == "I"] <- 4

## plot for comparing two closeness measures (incoming vs. outgoing)
plot(closeness(twitter.adj, mode = "in"),
     closeness(twitter.adj, mode = "out"), pch = pch, col = col,
     main = "Closeness", xlab = "Incoming path", ylab = "Outgoing path")
```



```
## plot for comparing directed and undirected betweenness
plot(betweenness(twitter.adj, directed = TRUE),
     betweenness(twitter.adj, directed = FALSE), pch = pch, col = col,
     main = "Betweenness", xlab = "Directed", ylab = "Undirected")
```

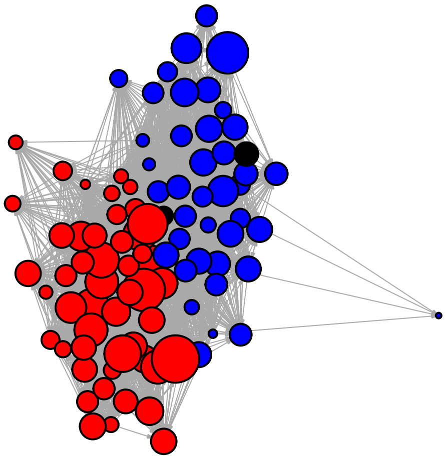


```r
senator$pagerank <- page.rank(twitter.adj)$vector

## `col' parameter is defined earlier
plot(twitter.adj, vertex.size = senator$pagerank * 1000,
     vertex.color = col, vertex.label = NA,
     edge.arrow.size = 0.1, edge.width = 0.5)

PageRank <- function(n, A, d, pr) { # function takes 4 inputs
    deg <- degree(A, mode = "out") # outdegree calculation
    for (j in 1:n) {
        pr[j] <- (1 - d) / n +  d * sum(A[ ,j] * pr / deg)
    }
    return(pr)
}

nodes <- 4

## adjacency matrix with arbitrary values
adj <- matrix(c(0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 0),
              ncol = nodes, nrow = nodes, byrow = TRUE)
adj
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    0    1    0    1
## [2,]    1    0    1    0
## [3,]    0    1    0    0
## [4,]    0    1    0    0
```

```r
adj <- graph.adjacency(adj)  # turn it into an igraph object
```

```
d <- 0.85  # typical choice of constant
pr <- rep(1 / nodes, nodes) # starting values

## maximum absolute difference; use a value greater than threshold
diff <- 100

## while loop with 0.001 being the threshold
while (diff > 0.001) {
    pr.pre <- pr # save the previous iteration
    pr <- PageRank(n = nodes, A = adj, d = d, pr = pr)
    diff <- max(abs(pr - pr.pre))
}
```



```
pr
```

```
## [1] 0.2213090 0.4316623 0.2209565 0.1315563
```

Section 5.3: Spatial Data

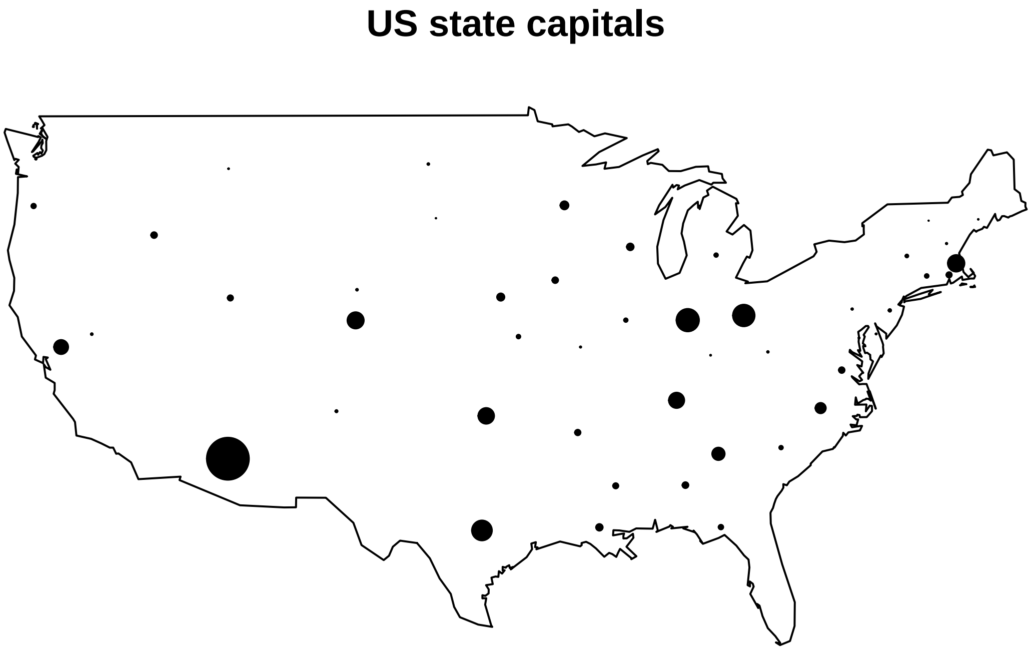
Section 5.3.1: The 1854 Cholera Outbreak in Action

Section 5.3.2: Spatial Data in R

```
library(maps)
data(us.cities)
head(us.cities)
```

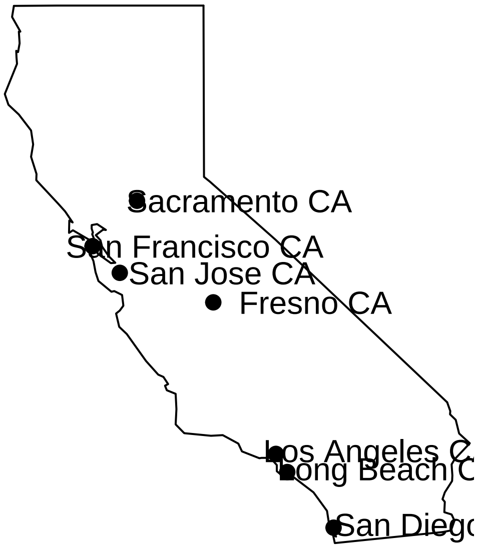
```
##          name country.etc    pop   lat    long capital
## 1 Abilene TX          TX 113888 32.45  -99.74       0
## 2   Akron OH          OH 206634 41.08  -81.52       0
## 3 Alameda CA          CA  70069 37.77 -122.26       0
## 4  Albany GA          GA  75510 31.58  -84.18       0
## 5  Albany NY          NY  93576 42.67  -73.80       2
## 6  Albany OR          OR  45535 44.62 -123.09       0
```

```
map(database = "usa")
capitals <- subset(us.cities, capital == 2) # subset state capitals

## add points proportional to population using latitude and longitude
points(x = capitals$long, y = capitals$lat,
       cex = capitals$pop / 500000, pch = 19)
title("US state capitals") # add a title
```



```
map(database = "state", regions = "California")

cal.cities <- subset(us.cities, subset = (country.etc == "CA"))
sind <- order(cal.cities$pop, decreasing = TRUE) # order by population
top7 <- sind[1:7] # seven cities with largest population

map(database = "state", regions = "California")

points(x = cal.cities$long[top7], y = cal.cities$lat[top7], pch = 19)

## add a constant to latitude to avoid overlapping with circles
text(x = cal.cities$long[top7] + 2.25, y = cal.cities$lat[top7],
     label = cal.cities$name[top7])
title("Largest cities of California")
```

Largest cities of California



```
usa <- map(database = "usa", plot = FALSE) # save map
names(usa)  # list elements
```

```
## [1] "x"     "y"     "range" "names"
```

```
length(usa$x)
```

```
## [1] 7252
```

```
head(cbind(usa$x, usa$y)) # first five coordinates of a polygon
```

```
##           [,1]     [,2]
## [1,] -101.4078 29.74224
## [2,] -101.3906 29.74224
## [3,] -101.3620 29.65056
## [4,] -101.3505 29.63911
## [5,] -101.3219 29.63338
## [6,] -101.3047 29.64484
```

Section 5.3.3: Colors in R

```
allcolors <- colors()

head(allcolors)   # some colors
```

```
## [1] "white"         "aliceblue"     "antiquewhite"  "antiquewhite1"
## [5] "antiquewhite2" "antiquewhite3"
```

```
length(allcolors) # number of color names
```

```
## [1] 657
```

```
red <- rgb(red = 1, green = 0, blue = 0) # red
green <- rgb(red = 0, green = 1, blue = 0) # green
blue <- rgb(red = 0, green = 0, blue = 1) # blue
c(red, green, blue) # results
```

```
## [1] "#FF0000" "#00FF00" "#0000FF"
```

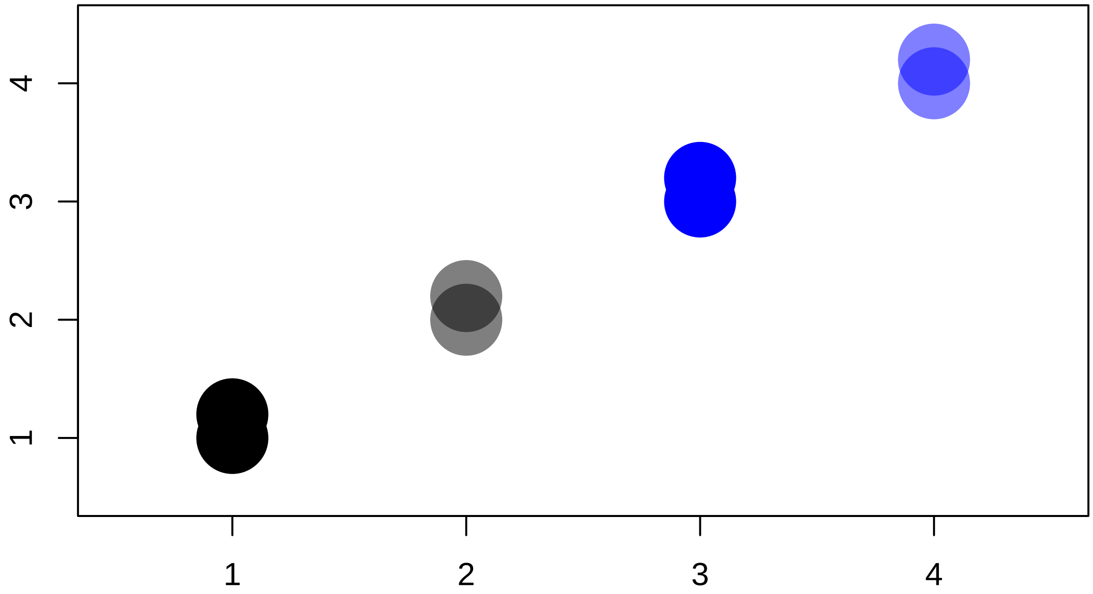
```
black <- rgb(red = 0, green = 0, blue = 0) # black
white <- rgb(red = 1, green = 1, blue = 1) # white
c(black, white) # results
```

```
## [1] "#000000" "#FFFFFF"
```

```
rgb(red = c(0.5, 1), green = c(0, 1), blue = c(0.5, 0))
```

```
## [1] "#800080" "#FFFF00"
```

```
## semi-transparent blue
blue.trans <- rgb(red = 0, green = 0, blue = 1, alpha = 0.5)

## semi-transparent black
black.trans <- rgb(red = 0, green = 0, blue = 0, alpha = 0.5)

## completely colored dots; difficult to distinguish
plot(x = c(1, 1), y = c(1, 1.2), xlim = c(0.5, 4.5), ylim = c(0.5, 4.5),
     pch = 16, cex = 5, ann = FALSE, col = black)
points(x = c(3, 3), y = c(3, 3.2), pch = 16, cex = 5, col = blue)

## semi-transparent; easy to distinguish
points(x = c(2, 2), y = c(2, 2.2), pch = 16, cex = 5, col = black.trans)
points(x = c(4, 4), y = c(4, 4.2), pch = 16, cex = 5, col = blue.trans)
```

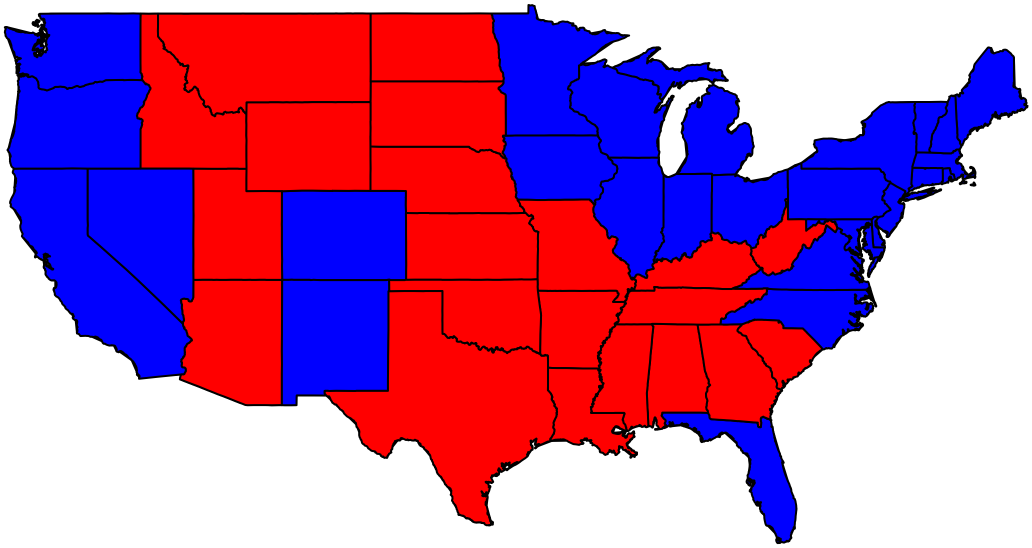


Section 5.3.4: US Presidential Elections

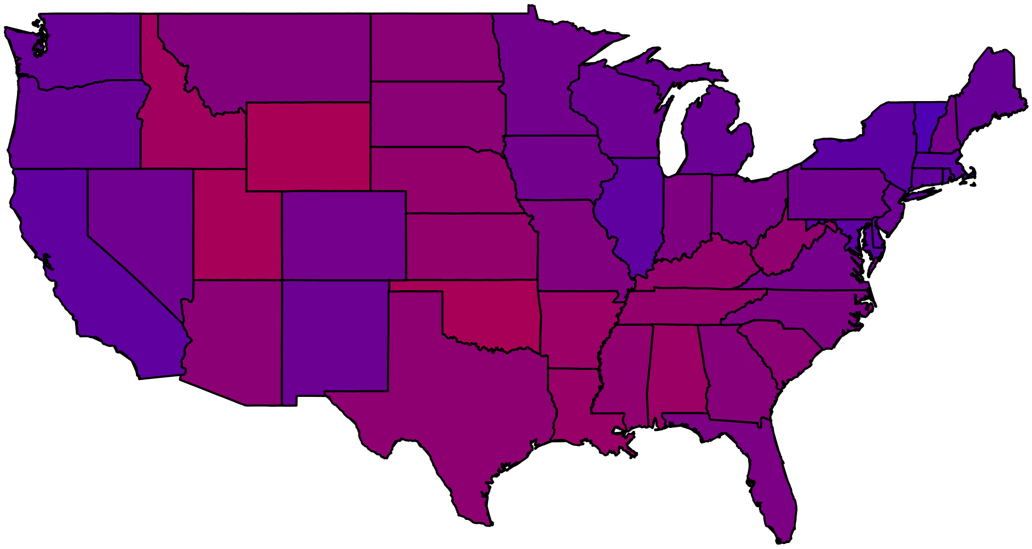
```
pres08 <- read.csv("pres08.csv")
## two-party vote share
pres08$Dem <- pres08$Obama / (pres08$Obama + pres08$McCain)
pres08$Rep <- pres08$McCain / (pres08$Obama + pres08$McCain)

## color for California
cal.color <- rgb(red = pres08$Rep[pres08$state == "CA"],
                 blue = pres08$Dem[pres08$state == "CA"],
                 green = 0)

## California as a blue state
map(database = "state", regions = "California", col = "blue",
    fill = TRUE)
```

```
## California as a purple state
map(database = "state", regions = "California", col = cal.color,
    fill = TRUE)
```

```
## America as red and blue states
map(database = "state") # create a map
map(database = "state") # for some reason this needs to be repeated twice to get the map correct if you
for (i in 1:nrow(pres08)) {
    if ((pres08$state[i] != "HI") & (pres08$state[i] != "AK") &
        (pres08$state[i] != "DC")) {
        maps::map(database = "state", regions = pres08$state.name[i],
            col = ifelse(pres08$Rep[i] > pres08$Dem[i], "red", "blue"),
            fill = TRUE, add = TRUE)
    }
}
```

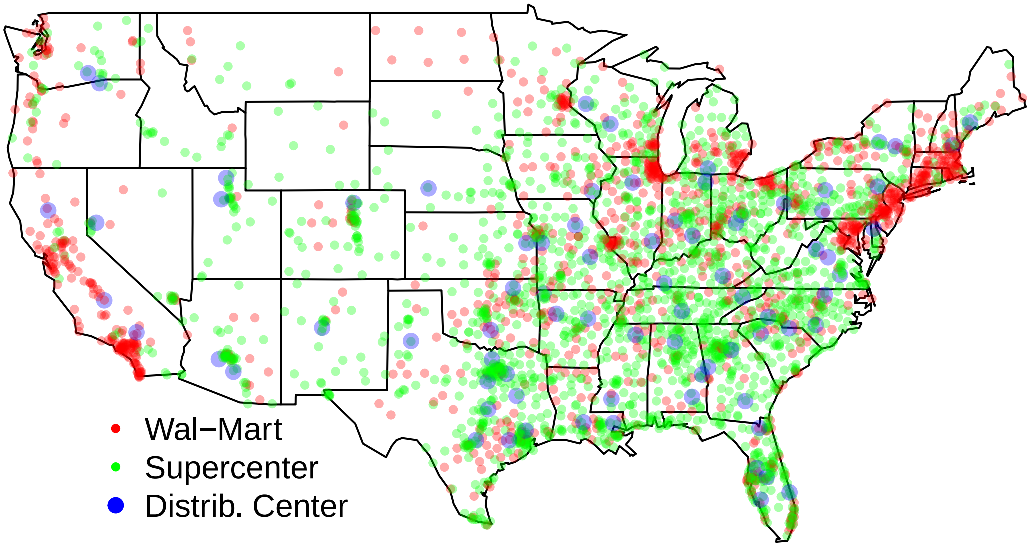


```
## America as purple states
map(database = "state") # create a map
for (i in 1:nrow(pres08)) {
    if ((pres08$state[i] != "HI") & (pres08$state[i] != "AK") &
        (pres08$state[i] != "DC")) {
        map(database = "state", regions = pres08$state.name[i],
            col = rgb(red = pres08$Rep[i], blue = pres08$Dem[i],
                green = 0), fill = TRUE, add = TRUE)
    }
}
```



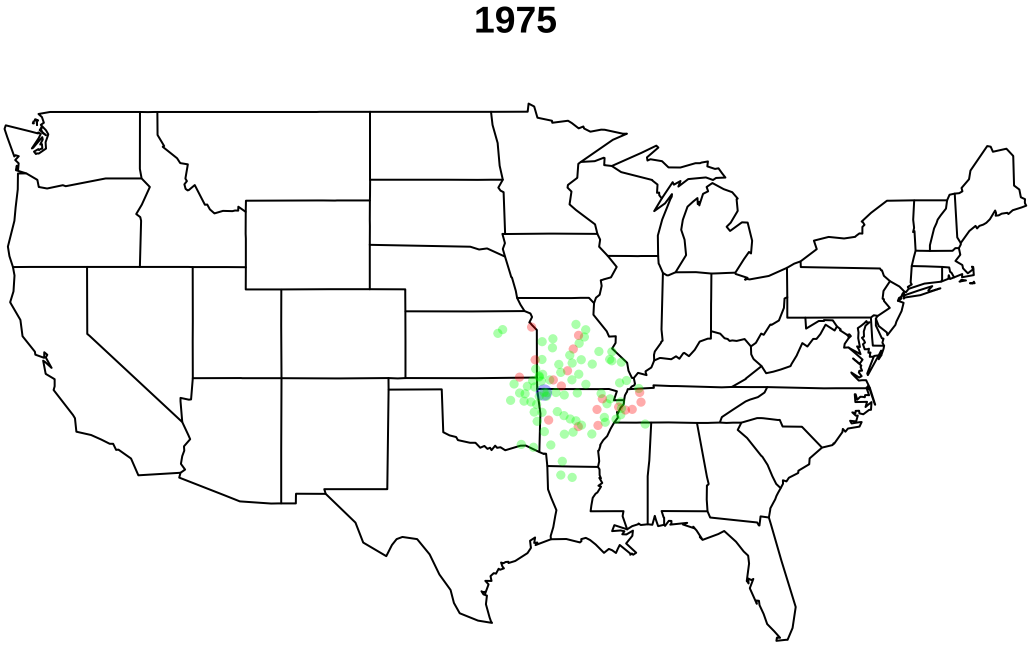
Section 5.3.5: Expansion of Walmart

```
walmart <- read.csv("walmart.csv")

## red = WalMartStore, green = SuperCenter, blue = DistributionCenter
walmart$storecolors <- NA # create an empty vector

walmart$storecolors[walmart$type == "Wal-MartStore"] <-
    rgb(red = 1, green = 0, blue = 0, alpha = 1/3)
walmart$storecolors[walmart$type == "SuperCenter"] <-
    rgb(red = 0, green = 1, blue = 0, alpha = 1/3)
walmart$storecolors[walmart$type == "DistributionCenter"] <-
    rgb(red = 0, green = 0, blue = 1, alpha = 1/3)

## larger circles for DistributionCenter
walmart$storesize <- ifelse(walmart$type == "DistributionCenter", 1, 0.5)

## map with legend
map(database = "state")

points(walmart$long, walmart$lat, col = walmart$storecolors,
       pch = 19, cex = walmart$storesize)

legend(x = -120, y = 32, bty = "n",
       legend = c("Wal-Mart", "Supercenter", "Distrib. Center"),
       col = c("red", "green", "blue"), pch = 19, # solid circles
       pt.cex = c(0.5, 0.5, 1)) # size of circles
```

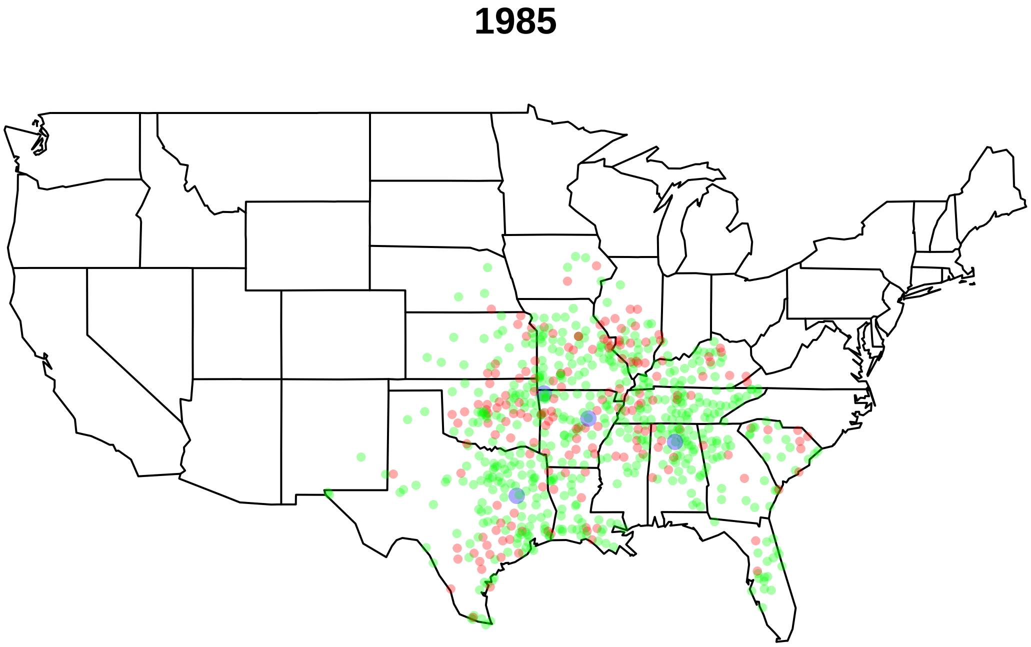


```
### Section 5.3.6: Animation in R

walmart.map <- function(data, date) {
    walmart <- subset(data, subset = (opendate <= date))
    map(database = "state")
    points(walmart$long, walmart$lat, col = walmart$storecolors,
           pch = 19, cex = walmart$storesize)
}
```

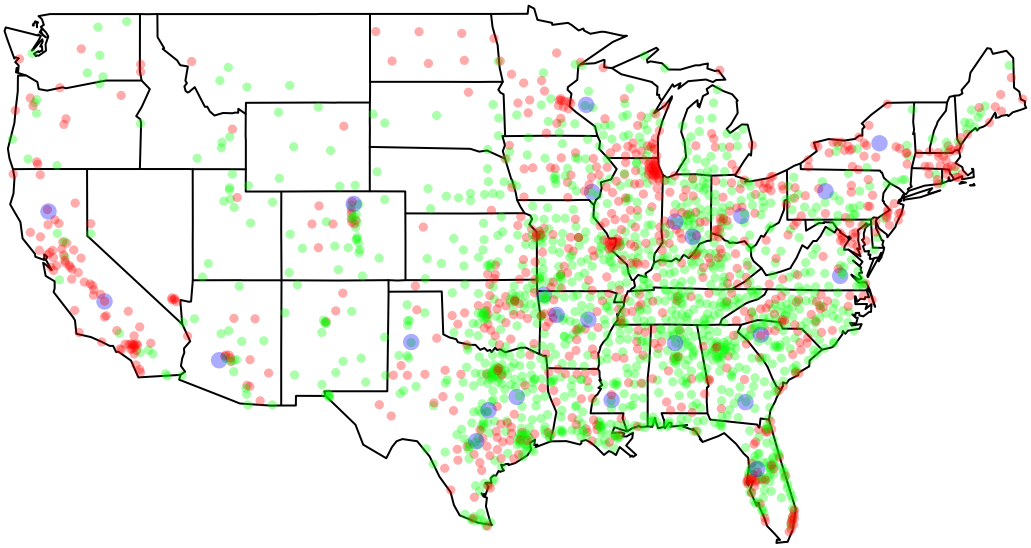
```r
walmart$opendate <- as.Date(walmart$opendate)

walmart.map(walmart, as.Date("1974-12-31"))
title("1975")
```



```r
walmart.map(walmart, as.Date("1984-12-31"))
title("1985")
```

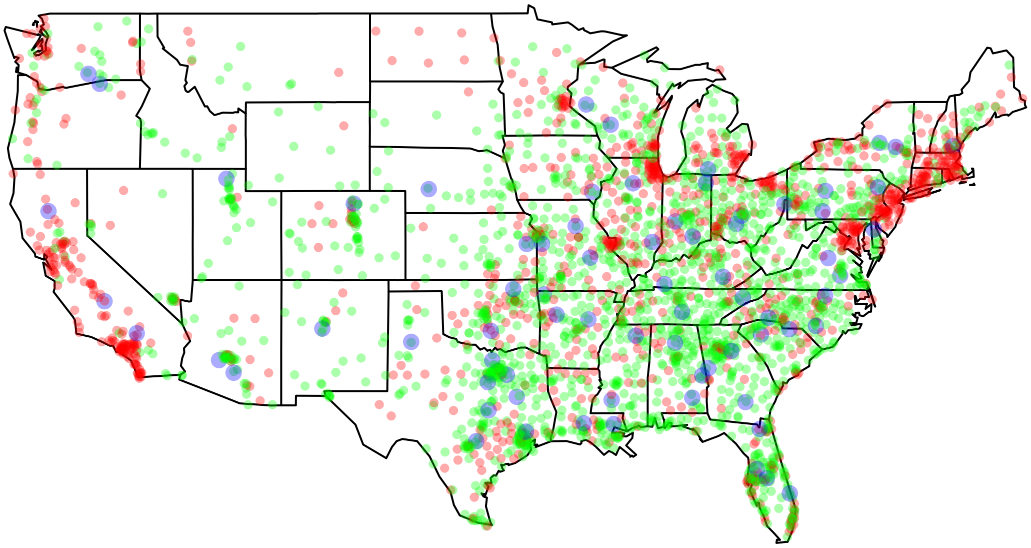


```r
walmart.map(walmart, as.Date("1994-12-31"))
title("1995")
```



1995

```
walmart.map(walmart, as.Date("2004-12-31"))
title("2005")
```



2005

```
n <- 25 # number of maps to animate
dates <- seq(from = min(walmart$opendate),
             to = max(walmart$opendate), length.out = n)
## library("animation")
## saveHTML({
##     for (i in 1:length(dates)) {
##         walmart.map(walmart, dates[i])
##         title(dates[i])
##     }
## }, title = "Expansion of Walmart", htmlfile = "walmart.html",
##           outdir = getwd(), autobrowse = FALSE)
```

5.4: Summary