Topic Modeling of Child Deaths

Gia Elise Barboza

December 20, 2018

Text Mining Coroner Autopsy Reports

Child fatality is a significant public health problem, and understanding its causes is critical for successful prevention efforts. Briefly, data limitations make it difficult to understand the risk factors for child death. The goal of this analysis is to begin to make sense of, or structure, textual data, which is messy (unstructured). This type of analysis allows us to utilize vast amounts of information that can be mined from the web.

This is a tutorial on how to perform Topic Modeling on narrative text information. The data is based on the unstructured corpus of the description surrounding death of children under 6 years old who died in Los Angeles County between 2000 - 2017.

Read data set

This will be a description of what the following code is doing:

```
source("F:/New Papers/text mining/child maltreatment/R code/multiplot.R")
source("F:/GSU/swords.R")

deathdat <- read.csv("E:/Summer Research/Data/Spatial Data/California/Homicide/FINAL CHILD DEATH
 DATA.csv", stringsAsFactors = FALSE)

deathdat[[9]] <- gsub( "GSW" , "gun shot wound" , deathdat[[9]])
deathdat[[9]] <- gsub( "Mgun" , "multiple gun shot wounds" , deathdat[[9]])
deathdat[[9]] <- gsub( "shoot" , "shot" , deathdat[[9]])


deathdat <- deathdat[c(1,3, 9,10)]
colnames(deathdat)[1] <- "doc_id"
colnames(deathdat)[2] <- "age"
colnames(deathdat)[3] <- "text"
colnames(deathdat)[4] <- "time_of_death"


docs <- VCorpus(DataframeSource(deathdat))

inspect(docs[1])
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 2
## Content:  documents: 1
##
## [[1]]
## <<PlainTextDocument>>
## Metadata:  7
## Content:  chars: 181
```

```
docs <- tm_map(docs,removePunctuation)
docs <- tm_map(docs, removeNumbers)
docs <- tm_map(docs,content_transformer(tolower))
docs <- tm_map(docs, removeWords, stopwords("english"))
docs <- tm_map(docs, removeWords, s_words)

docs <- tm_map(docs, stripWhitespace)
docs<-tm_map(docs, stemDocument)
docs <- tm_map(docs, PlainTextDocument)

docs[[18]]$content
```

```
## [1] "appar full term fetus trash dumpster plastic bag unknown mechan injuri newborn trash dum
pster believ deliv bathroom boat put trash"
```

```
tdm <- TermDocumentMatrix(docs, control=list(bounds = list(global = c(5,Inf))))
dim(tdm) # after Terms that appear in <5 documents are discarded
```

```
## [1] 678 437
```

```
dtm <- DocumentTermMatrix(docs, control=list(bounds = list(global = c(5,Inf))))
dim(dtm) # after Terms that appear in <5 documents are discarded
```

```
## [1] 437 678
```

```
rownames(dtm) <- deathdat$doc_id

freq <- colSums(as.matrix(dtm))
length(freq)
```

```
## [1] 678
```

```
ord <- order(freq)

term_tfidf <- tapply(dtm$v/slam::row_sums(dtm)[dtm$i], dtm$j, mean) *
  log2(tm::nDocs(dtm)/slam::col_sums(dtm > 0))
summary(term_tfidf)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.04952 0.12815 0.15332 0.16467 0.18640 0.62649
```

```
m_tdm <- as.matrix(tdm)


m_dtm <- as.matrix(dtm)
dim(m_dtm)
```

```
## [1] 437 678
```

```
v <- sort(rowSums(m_tdm),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)

# Find the sum of words in each Document and remove all docs w/out words
# or else an error will result below
rowTotals <- apply(dtm , 1, sum)
dtm   <- dtm[rowTotals> 0, ]

#dtm <- dtm[,term_tfidf >= 0.155]
summary(slam::col_sums(dtm))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    5.00    7.00   12.00   22.94   26.00  271.00
```

```
freqs <- slam::col_sums(dtm)

# Save this for Shiny app later on
write.csv(m_dtm, file=paste("F:/Examples/ITMViz-master/ITMViz-master/data/jedit-5.1.0", "Documen
tTermMatrix.csv", sep="/"))

dtm$dimnames$Terms
```

```
##   [1] "abdomen"        "abdomin"        "abnorm"         "abras"
##   [5] "abrupt"         "abus"           "accid"          "accident"
##   [9] "accord"         "activ"          "acut"           "addit"
##  [13] "admiss"         "admit"          "adult"          "air"
##  [17] "airlift"        "airway"         "aliv"           "allow"
##  [21] "alon"           "alter"          "anemia"         "angel"
##  [25] "anoth"          "anox"           "anterior"       "apneic"
##  [29] "appar"          "appear"         "approx"         "area"
##  [33] "arm"            "arrest"         "arriv"          "artifici"
##  [37] "asleep"         "asphyxi"        "asphyxia"       "assail"
##  [41] "assault"        "assum"          "asthma"         "attach"
##  [45] "attempt"        "attend"         "attent"         "aunt"
##  [49] "auto"           "avail"          "babi"           "babysitt"
##  [53] "back"           "bag"            "bath"           "bathroom"
##  [57] "bathtub"        "batter"         "beach"          "beat"
##  [61] "beaten"         "becam"          "bed"            "bedroom"
##  [65] "behind"         "believ"         "belt"           "bilater"
##  [69] "biolog"         "birth"          "bite"           "blanket"
##  [73] "bleed"          "blood"          "blunt"          "bodi"
##  [77] "born"           "bottl"          "box"            "boyfriend"
##  [81] "brain"          "breath"         "brother"        "brought"
##  [85] "bruis"          "build"          "buri"           "burn"
##  [89] "buttock"        "calib"          "call"           "car"
##  [93] "cardiac"        "care"           "caregiv"        "carpet"
##  [97] "carri"          "case"           "caus"           "center"
## [101] "cerebr"         "cesarean"       "chang"          "charg"
## [105] "check"          "cheek"          "chest"          "child"
## [109] "children"       "chin"           "chla"           "choke"
## [113] "chronic"        "circumst"       "citi"           "claim"
## [117] "clean"          "clinic"         "close"          "closet"
## [121] "cloth"          "cold"           "collect"        "color"
## [125] "comatos"        "come"           "commit"         "companion"
## [129] "complet"        "complic"        "concern"        "condit"
## [133] "confirm"        "conflict"       "consequ"        "consist"
## [137] "contact"        "continu"        "contus"         "cord"
## [141] "cosleep"        "couch"          "cough"          "counti"
## [145] "coupl"          "court"          "cover"          "cpr"
## [149] "cri"            "crib"           "critic"         "csection"
## [153] "current"        "cut"            "cyanot"         "daughter"
## [157] "dcfs"           "dead"           "death"          "deceas"
## [161] "deced"          "declar"         "declin"         "decreas"
## [165] "defect"         "degre"          "dehydr"         "deliv"
## [169] "deliveri"       "demis"          "deni"           "depart"
## [173] "deputi"         "despit"         "det"            "deterior"
## [177] "determin"       "develop"        "development"    "diagnos"
## [181] "diaper"         "die"            "disclos"        "discov"
## [185] "distend"        "distress"       "doctor"         "document"
## [189] "domest"         "donat"          "door"           "drain"
## [193] "drink"          "drive"          "driven"         "drop"
## [197] "drove"          "drown"          "drug"           "due"
## [201] "dump"           "dumpster"       "ear"            "earli"
## [205] "eat"            "edema"          "effort"         "emerg"
## [209] "encephalopathi" "enforc"         "enter"          "episod"
```

```
## [213] "estim"         "etoh"          "evacu"         "evalu"
## [217] "even"          "event"         "eventu"        "evid"
## [221] "exam"          "examin"        "exit"          "expect"
## [225] "experienc"     "expir"         "extrem"        "eye"
## [229] "face"          "factor"        "fail"          "failur"
## [233] "fall"          "fallen"        "famili"        "father"
## [237] "fed"           "feed"          "feet"          "fell"
## [241] "femal"         "fetal"         "fetus"         "fever"
## [245] "file"          "fill"          "find"          "fire"
## [249] "first"         "fist"          "float"         "floor"
## [253] "follow"        "food"          "foot"          "forc"
## [257] "forehead"      "form"          "foster"        "foul"
## [261] "four"          "fractur"       "franci"        "fresh"
## [265] "friend"        "front"         "full"          "fullterm"
## [269] "gang"          "garag"         "genit"         "gestat"
## [273] "get"           "girl"          "give"          "good"
## [277] "got"           "grandmoth"     "grave"         "ground"
## [281] "group"         "gun"           "gunshot"       "hand"
## [285] "handl"         "harvest"       "head"          "heal"
## [289] "health"        "healthi"       "hear"          "heard"
## [293] "heart"         "help"          "hematoma"      "hemorrhag"
## [297] "high"          "higher"        "hispan"        "histori"
## [301] "hit"           "hold"          "home"          "homicid"
## [305] "hospit"        "hotel"         "hous"          "husband"
## [309] "icu"           "ill"           "incid"         "includ"
## [313] "inconsist"     "indic"         "infant"        "infect"
## [317] "inflict"       "inform"        "initi"         "injur"
## [321] "injuri"        "insid"         "intent"        "intermitt"
## [325] "intern"        "intervent"     "interview"     "intracrani"
## [329] "intrauterin"   "intub"         "investig"      "involv"
## [333] "issu"          "jail"          "jump"          "just"
## [337] "kick"          "kill"          "kit"           "kitchen"
## [341] "knee"          "knife"         "know"          "known"
## [345] "labor"         "lacer"         "larg"          "last"
## [349] "law"           "leav"          "left"          "leg"
## [353] "letharg"       "lie"           "life"          "lifesav"
## [357] "lifesupport"   "like"          "limp"          "lip"
## [361] "live"          "liver"         "local"         "locat"
## [365] "lock"          "long"          "look"          "lot"
## [369] "low"           "lower"         "made"          "male"
## [373] "mani"          "mark"          "matern"        "may"
## [377] "mechan"        "med"           "medic"         "member"
## [381] "memori"        "mental"        "met"           "methamphetamin"
## [385] "minut"         "miss"          "mom"           "monday"
## [389] "monthold"      "morn"          "morti"         "mouth"
## [393] "move"          "multipl"       "murder"        "nap"
## [397] "natur"         "near"          "nearbi"        "neck"
## [401] "need"          "negat"         "neglect"       "neighbor"
## [405] "never"         "newborn"       "next"          "night"
## [409] "nonaccident"   "normal"        "nose"          "note"
## [413] "notic"         "notifi"        "number"        "numer"
## [417] "observ"        "obvious"       "occur"         "odor"
## [421] "offic"         "onscen"        "open"          "oral"
## [425] "organ"         "origin"        "outsid"        "pain"
```

```
## [429] "pale"          "paper"          "para"           "parent"
## [433] "pariet"        "park"           "partial"        "past"
## [437] "pattern"       "pediatrician"   "pelvic"         "perform"
## [441] "period"        "person"         "personnel"      "petechi"
## [445] "petechia"      "physic"         "physician"      "pick"
## [449] "picu"          "pillow"         "place"          "placenta"
## [453] "plastic"       "play"           "pleas"          "pneumonia"
## [457] "poison"        "polic"          "poor"           "posit"
## [461] "possibl"       "postmortem"     "pound"          "pregnanc"
## [465] "pregnant"      "preliminari"    "prematur"       "prenat"
## [469] "present"       "previous"       "prior"          "privat"
## [473] "problem"       "progress"       "prone"          "provid"
## [477] "pull"          "puls"           "push"           "put"
## [481] "question"      "ramirez"        "ran"            "reach"
## [485] "rear"          "receiv"         "recent"         "record"
## [489] "recov"         "rectal"         "red"            "refer"
## [493] "regard"        "relat"          "releas"         "remain"
## [497] "remov"         "report"         "request"        "requir"
## [501] "resid"         "respiratori"    "respond"        "result"
## [505] "resuscit"      "retin"          "return"         "reveal"
## [509] "reviv"         "rib"            "rigor"          "roll"
## [513] "room"          "rule"           "run"            "runni"
## [517] "said"          "save"           "saw"            "scan"
## [521] "scar"          "scene"          "school"         "seat"
## [525] "second"        "section"        "secur"          "seen"
## [529] "seizur"        "sent"           "sequela"        "set"
## [533] "sever"         "sexual"         "shake"          "shaken"
## [537] "sharp"         "sheriff"        "shirt"          "shook"
## [541] "shoot"         "short"          "shot"           "shoulder"
## [545] "show"          "shower"         "sibl"           "sickl"
## [549] "sid"           "side"           "sidewalk"       "sign"
## [553] "signific"      "sinc"           "sister"         "sit"
## [557] "sitter"        "skin"           "skull"          "sleep"
## [561] "small"         "son"            "sought"         "stab"
## [565] "staff"         "stage"          "stand"          "start"
## [569] "state"         "statement"      "station"        "status"
## [573] "stay"          "step"           "stepfath"       "stiff"
## [577] "still"         "stillborn"      "stomach"        "stop"
## [581] "store"         "stori"          "strike"         "struck"
## [585] "studi"         "subarachnoid"   "subdur"         "subject"
## [589] "suffer"        "suffoc"         "suicid"         "summon"
## [593] "sunset"        "supin"          "support"        "surgeri"
## [597] "surviv"        "suspect"        "suspici"        "sustain"
## [601] "swell"         "swollen"        "symptom"        "syndrom"
## [605] "tabl"          "take"           "taken"          "temperatur"
## [609] "term"          "thermal"        "thigh"          "thin"
## [613] "thought"       "three"          "threw"          "throat"
## [617] "today"         "toddler"        "tongu"          "top"
## [621] "torso"         "toward"         "towel"          "transfer"
## [625] "trash"         "traumat"        "treat"          "tri"
## [629] "troubl"        "tub"            "turn"           "twin"
## [633] "type"          "umbil"          "unabl"          "unattend"
## [637] "unawar"        "uncl"           "unclear"        "underw"
## [641] "undetermin"    "unestablish"    "unit"           "unknown"
```

```
## [645] "unrespons"      "upper"          "urin"           "use"
## [649] "vagin"          "veh"            "vehicl"         "vent"
## [653] "ventil"         "victim"         "violenc"        "visibl"
## [657] "visit"          "vomit"          "walk"           "watch"
## [661] "water"          "weapon"         "week"           "weigh"
## [665] "weight"         "well"           "wet"            "wife"
## [669] "window"         "wit"            "withdrawn"      "woke"
## [673] "work"           "worker"         "wound"          "wrap"
## [677] "yard"           "yet"
```

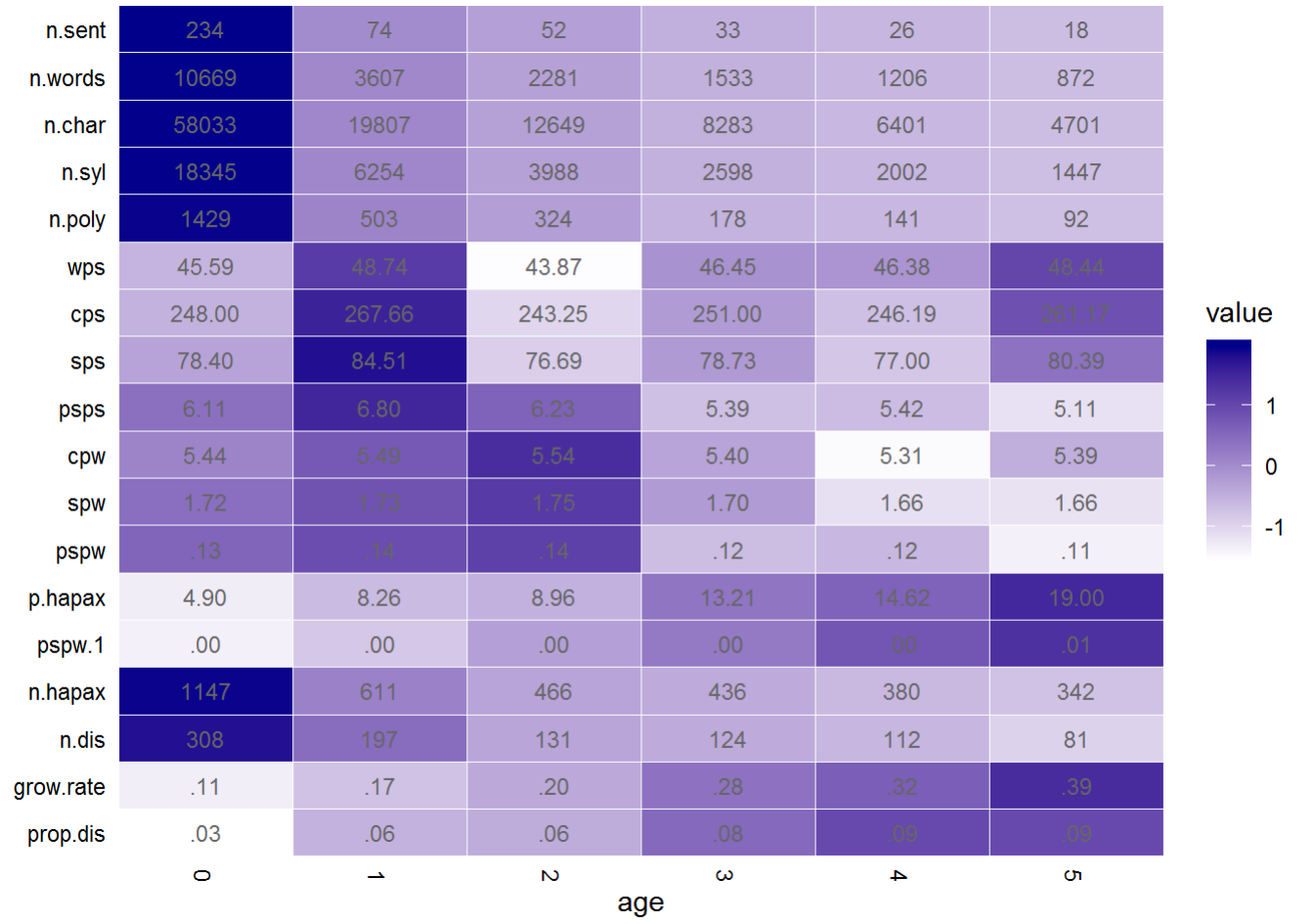
```
dat <-data.frame(text=unlist(sapply(docs, `[`, "content")), stringsAsFactors=F)
dat <- cbind(deathdat[, 1:2,  4], dat)

wordMat <- wfm(dat$text, dat$age )
ws <- word_stats(dat$text, dat$age, rm.incomplete = T)
```

```
## Warning in word_stats(dat$text, dat$age, rm.incomplete = T): Some sentences do not have stand
ard qdap punctuation endmarks.
##   Use $mpun for a list of observations with missing endmarks.
```

```
plot(ws, label = T, lab.digits = 2)
```

```
## Warning: attributes are not identical across measure variables; they will
## be dropped
```

```
## Warning: Ignoring unknown aesthetics: fill
```

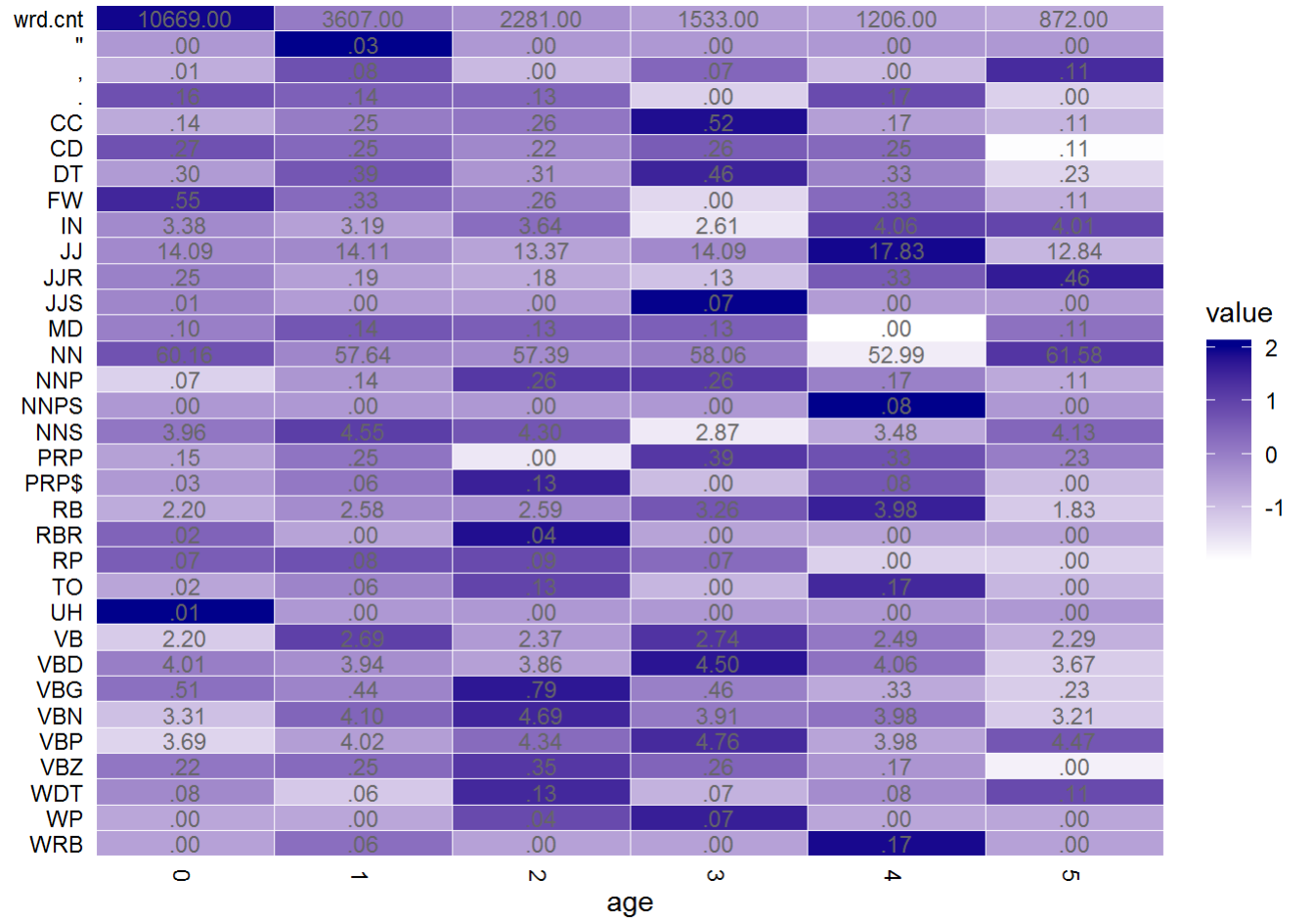



```
posbydf <- pos_by(dat$text, grouping.var = dat$age)
names(posbydf)
```

```
##  [1] "text"         "POStagged"    "POSprop"      "POSfreq"
##  [5] "POSrnp"       "percent"      "zero.replace" "pos.by.freq"
##  [9] "pos.by.prop"  "pos.by.rnp"
```

```
plot(posbydf, values = T, digits = 2)
```

```
## Warning: Ignoring unknown aesthetics: fill
```




```
automated_readability_index(dat$text, dat$age)
```

```
##   age word.count sentence.count character.count Automated_Readability_Index
## 1   0      10669            234           58033                      26.987
## 2   1       3607             74           19807                      28.805
## 3   2       2281             52           12649                      26.621
## 4   3       1533             33            8283                      27.246
## 5   4       1206             26            6401                      26.761
## 6   5        872             18            4701                      28.184
```

```
diversity(dat$text, dat$age)
```

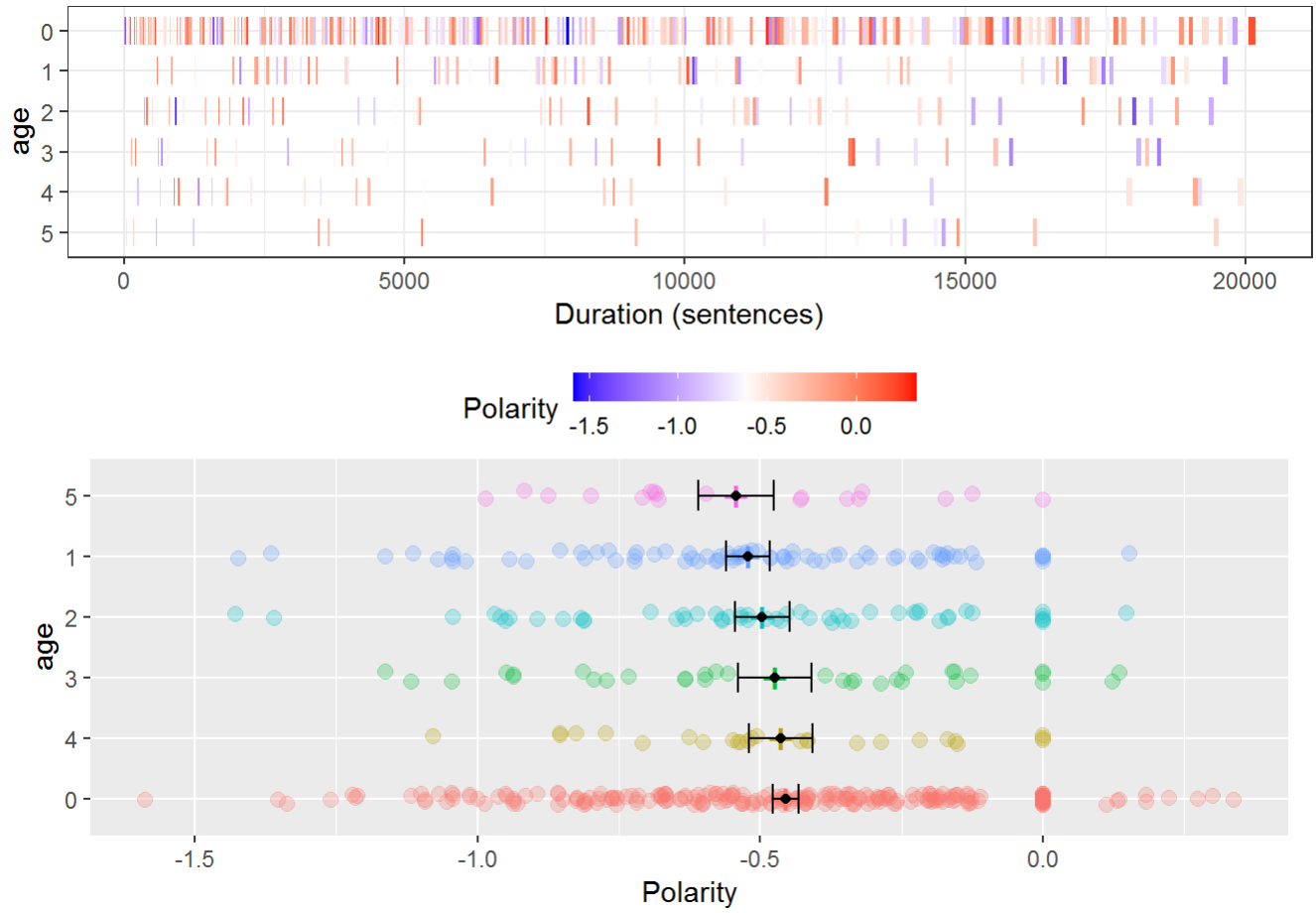
```
##   age    wc simpson shannon collision berger_parker brillouin
## 1   0 10669   0.997   6.753     5.951         0.019     6.470
## 2   1  3607   0.997   6.404     5.763         0.014     5.996
## 3   2  2281   0.997   6.150     5.543         0.018     5.697
## 4   3  1533   0.997   6.166     5.691         0.019     5.610
## 5   4  1206   0.997   6.079     5.702         0.012     5.484
## 6   5   872   0.996   5.847     5.346         0.028     5.207
```

```
pol <- polarity(dat$text, dat$age)
plot(pol)
```

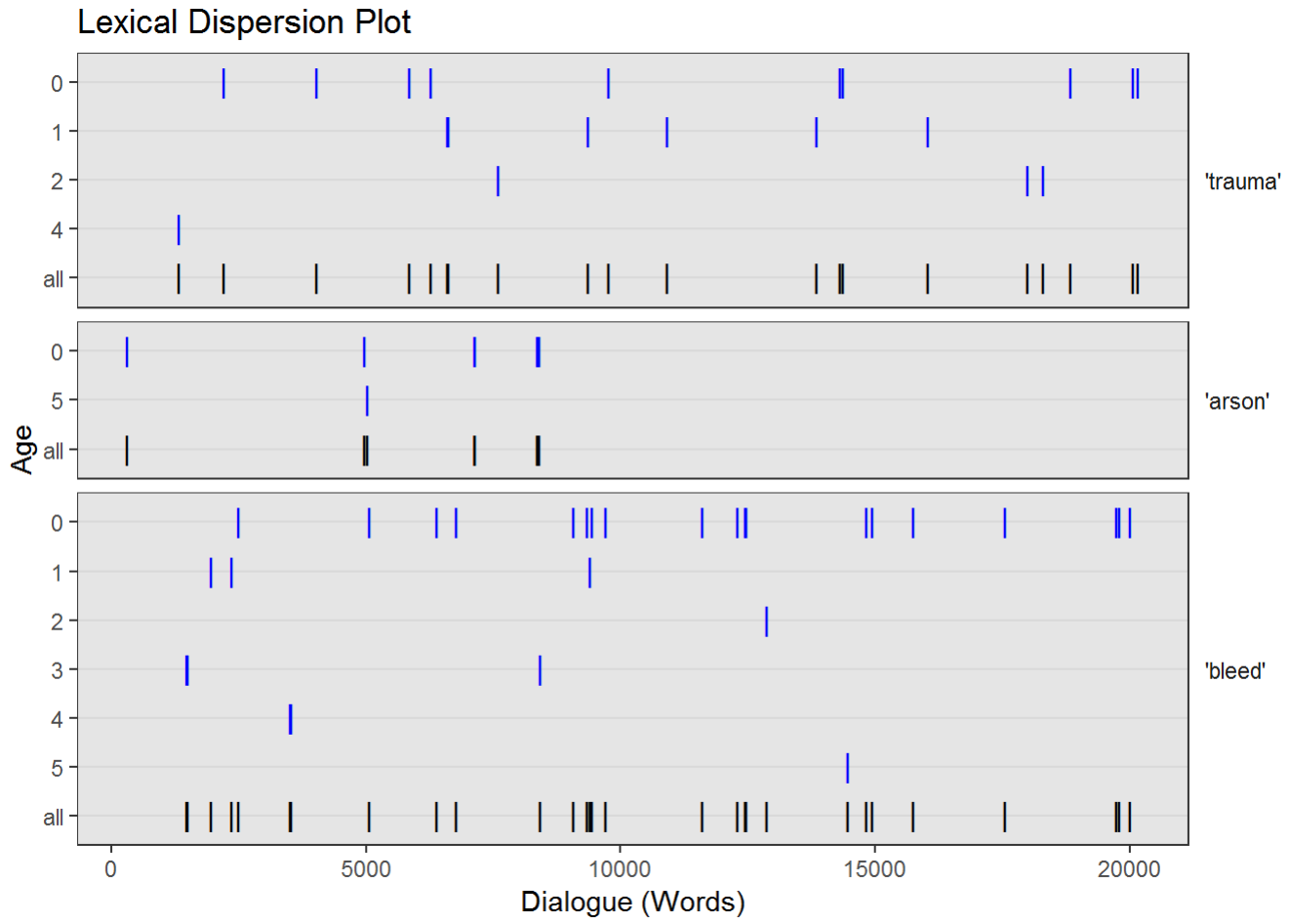
```
## Warning: `show_guide` has been deprecated. Please use `show.legend`
## instead.
```

```
## Warning: Ignoring unknown aesthetics: x
```

```
## Warning: `show_guide` has been deprecated. Please use `show.legend`
## instead.
```

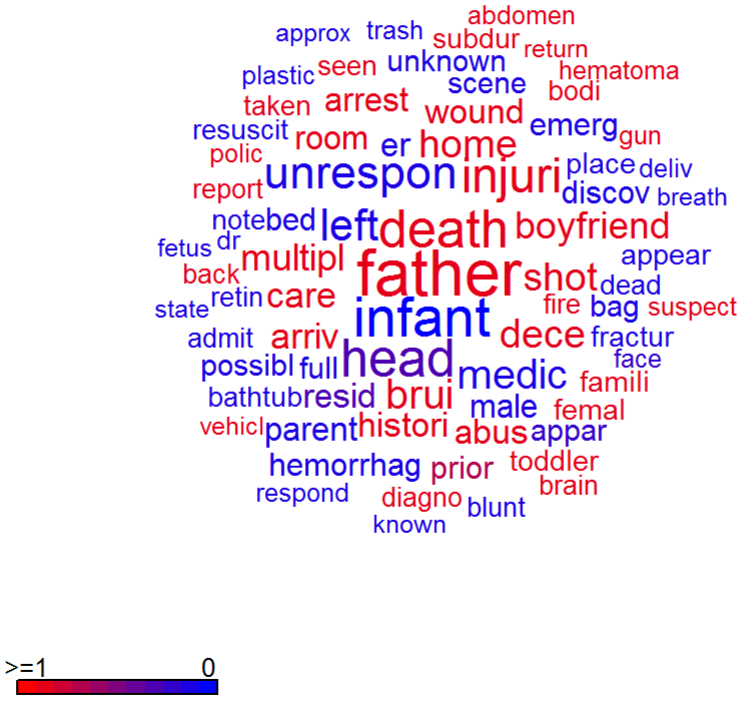



```
dispersion_plot(dat$text, c("trauma", "arson", "bleed", "gestation"), dat$age)
```




```
dat$agegrp <- NA
dat$agegrp <- ifelse(dat$age < 1, "0", ">=1")
gradient_cloud(dat$text, dat$agegrp, min.freq = 50, stem = T, max.word.size = 2)
```

```
## Warning in stemmer(text.var): The following row(s) do have standard qdap punctuation endmark
s:
##   rows: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 2
4, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 4
8, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 7
2, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 9
6, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 1
16, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 13
5, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 15
4, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 17
3, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 19
2, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 21
1, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 23
0, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 24
9, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 26
8, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 28
7, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 30
6, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 32
5, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 34
4, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 36
3, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 38
2, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 40
1, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 42
0, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437
```



```
############################################################
```

```
p1 <- ggplot(subset(d[1:50,], freq>15), aes(x = reorder(word, -freq), y = freq)) +
  geom_bar(stat = "identity") +
  theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("")

dtm_tfidf <- DocumentTermMatrix(docs, control = list(weighting = weightTfIdf))
dtm_tfidf = removeSparseTerms(dtm_tfidf, 0.95)
dtm_tfidf
```

```
## <<DocumentTermMatrix (documents: 437, terms: 169)>>
## Non-/sparse entries: 7678/66175
## Sparsity           : 90%
## Maximal term length: 9
## Weighting          : term frequency - inverse document frequency (normalized) (tf-idf)
```

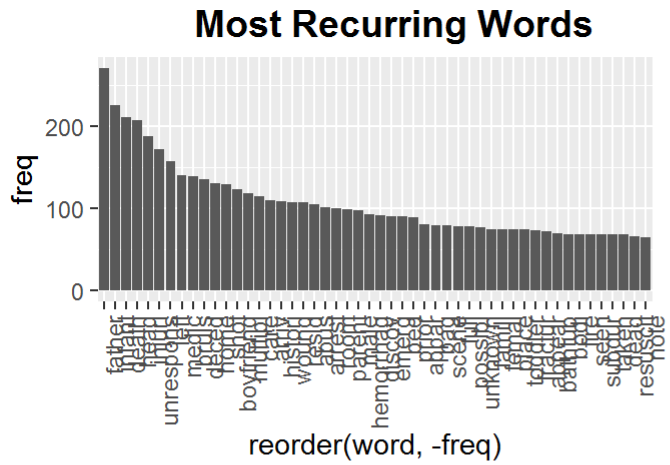
```
freq = data.frame(sort(colSums(as.matrix(dtm_tfidf)), decreasing=TRUE))

freq$words <- rownames(freq)
colnames(freq)[1] <- "termFreq"

p2 <- ggplot(freq[1:50, ], aes(x = reorder(words, -termFreq), y = termFreq)) +
  geom_bar(stat = "identity") +
  theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("")

death_bigram <- tokens(dat$text) %>%
  tokens_remove("\\p{P}", valuetype = "regex", padding = TRUE) %>%
  tokens_remove(stopwords("english"), padding  = TRUE) %>%
  tokens_ngrams(n = 2) %>%
  dfm()
topfeatures(death_bigram)
```

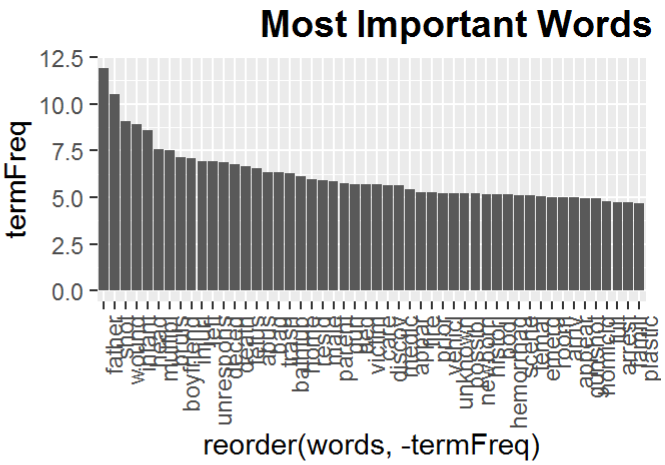
```
##      emerg_room retin_hemorrhag        gun_shot      shot_wound
##              70              58              52              50
## subdur_hematoma     full_arrest  cardiac_arrest     known_medic
##              48              40              37              37
##     plastic_bag   gunshot_wound
##              36              33
```

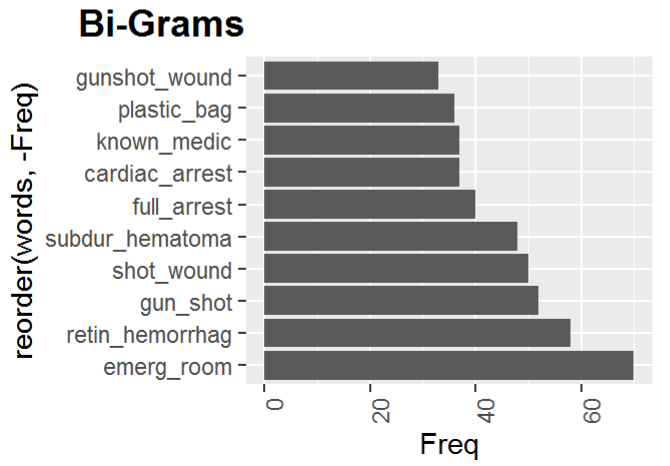
```
bi <- data.frame(topfeatures(death_bigram))

bi$words <- rownames(bi)
colnames(bi)[1] <- "Freq"

p3 <- ggplot(bi, aes(x = reorder(words, -Freq), y = Freq)) +
  geom_bar(stat = "identity") + coord_flip() +
  theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("")


##Create tri-grams
death_trigram <- tokens(dat$text) %>%
  tokens_remove("\\p{P}", valuetype = "regex", padding = TRUE) %>%
  tokens_remove(stopwords("english"), padding  = TRUE) %>%
  tokens_ngrams(n = 3) %>%
  dfm()
topfeatures(death_trigram)
```

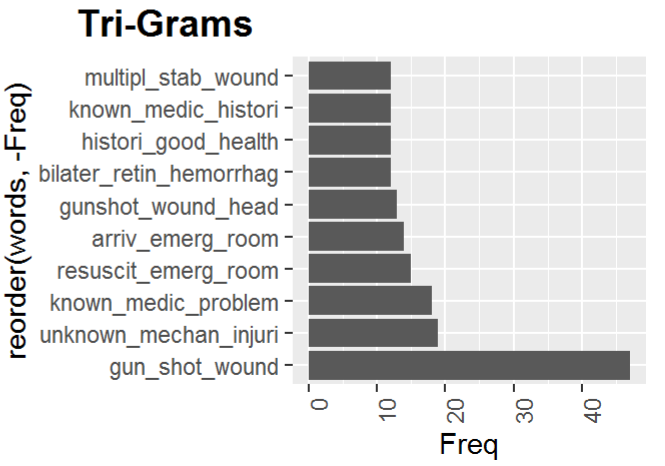
```
##          gun_shot_wound   unknown_mechan_injuri     known_medic_problem
##                      47                      19                      18
##     resuscit_emerg_room        arriv_emerg_room      gunshot_wound_head
##                      15                      14                      13
##      multipl_stab_wound bilater_retin_hemorrhag     histori_good_health
##                      12                      12                      12
##     known_medic_histori
##                      12
```

```
tri <- data.frame(topfeatures(death_trigram))

tri$words <- rownames(tri)
colnames(tri)[1] <- "Freq"

p4 <- ggplot(tri, aes(x = reorder(words, -Freq), y = Freq)) +
  geom_bar(stat = "identity") + coord_flip()+
  theme(axis.text.x=element_text(angle=90, hjust=1)) + ggtitle("")


plot_grid(p1,p2, p3,p4, labels=c('Most Recurring Words', 'Most Important Words (TF-IDF)', 'Bi-Gr
ams', 'Tri-Grams'))
```










```
summary(slam::col_sums(dtm))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    5.00    7.00   12.00   22.94   26.00  271.00
```

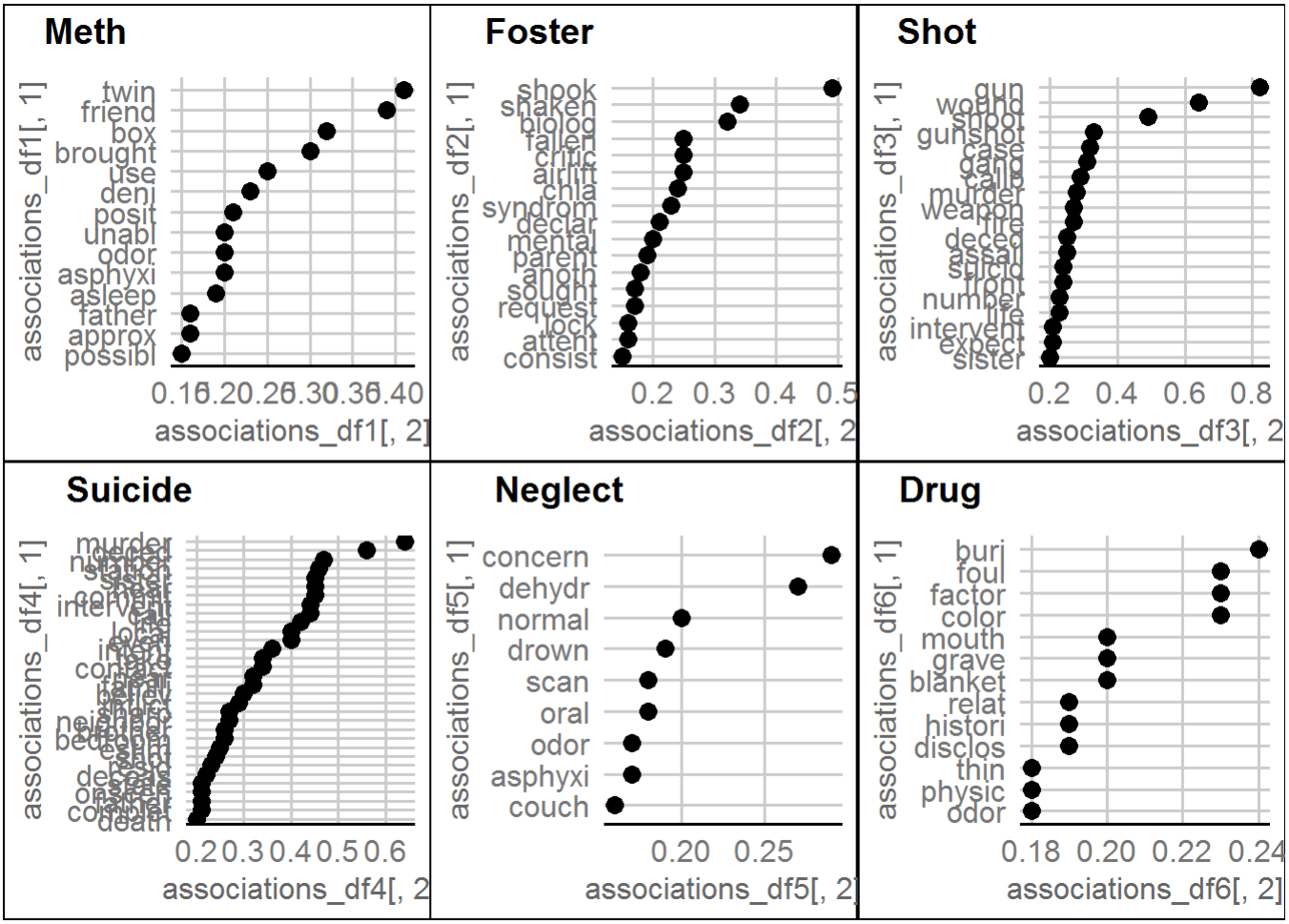
```
freqs <- slam::col_sums(dtm)

words <- colnames(dtm)
wordlist <- data.frame(words, freqs)
wordIndexes <- order(wordlist[, "freqs"], decreasing = TRUE)
wordlist <- wordlist[wordIndexes, ]

head(wordlist, 55)
```

```
##               words freqs
## father       father   271
## infant       infant   226
## death         death   211
## head           head   207
## injuri       injuri   188
## unrespons unrespons   172
## left           left   157
## medic         medic   141
## bruis         bruis   139
## deced         deced   135
## home           home   131
## shot           shot   130
## boyfriend boyfriend   123
## multipl     multipl   119
## care           care   115
## arriv         arriv   110
## histori     histori   109
## wound         wound   108
## resid         resid   107
## abus           abus   105
## arrest       arrest   101
## room           room   100
## parent       parent    99
## male           male    98
## hemorrhag hemorrhag    93
## discov       discov    92
## emerg         emerg    91
## bed             bed    90
## prior         prior    89
## appar         appar    81
## bag             bag    80
## scene         scene    79
## full           full    78
## possibl     possibl    78
## unknown     unknown    77
## famili       famili    75
## femal         femal    75
## place         place    75
## toddler     toddler    75
## fractur     fractur    73
## appear       appear    72
## bathtub     bathtub    70
## bodi           bodi    69
## fire           fire    69
## seen           seen    69
## subdur       subdur    69
## taken         taken    69
## dead           dead    68
## resuscit   resuscit    66
## note           note    65
## retin         retin    64
## admit         admit    63
```

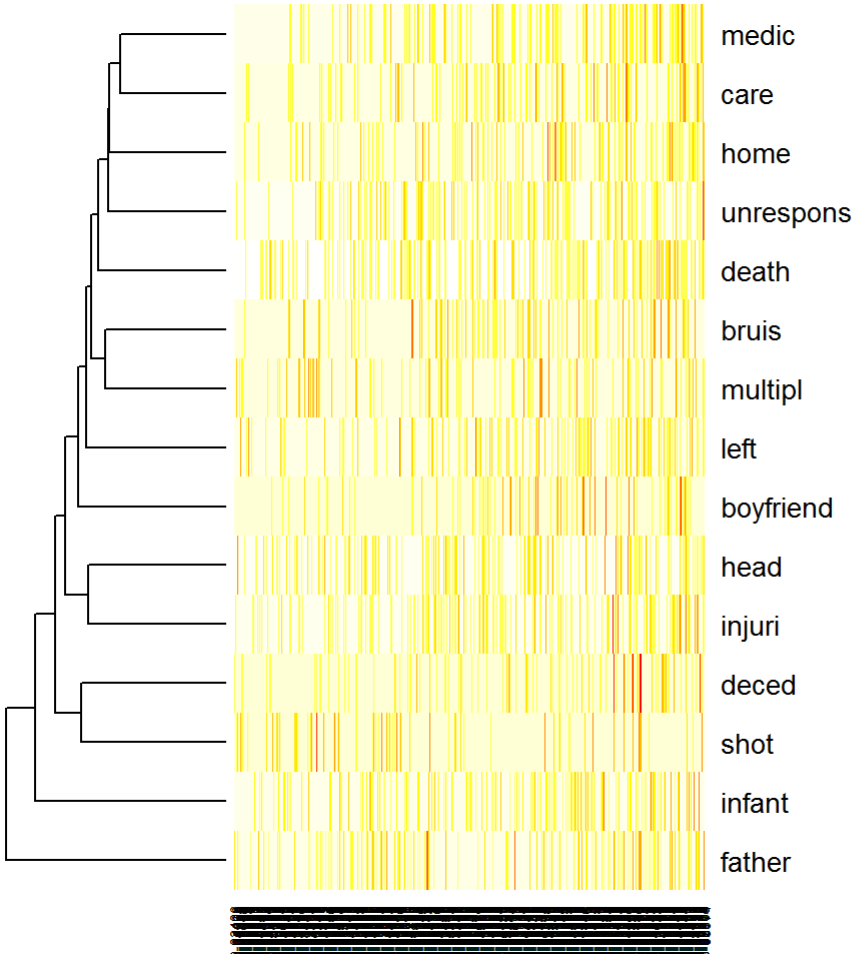
```
## report       report    63
## back           back    62
## diagnos     diagnos    62
```

```
tdms <- removeSparseTerms(tdm, 0.99)
# Create, save and plot associations
# This will be interactive in Shiny App later
associations <- findAssocs(tdm, "methamphetamin", 0.15)
associations_df1 <- list_vect2df(associations)[, 2:3]

p1<-ggplot(associations_df1, aes(y = associations_df1[, 1])) +
  geom_point(aes(x = associations_df1[, 2]),
             data = associations_df1, size = 3) +
  ggtitle("") +
  theme_gdocs()



associations <- findAssocs(tdm, "foster", 0.15)
associations_df2 <- list_vect2df(associations)[, 2:3]

p2<-ggplot(associations_df2, aes(y = associations_df2[, 1])) +
  geom_point(aes(x = associations_df2[, 2]),
             data = associations_df2, size = 3) +
  ggtitle("") +
  theme_gdocs()


associations <- findAssocs(tdm, "shot", 0.2)
associations_df3 <- list_vect2df(associations)[, 2:3]

p3<- ggplot(associations_df3, aes(y = associations_df3[, 1])) +
  geom_point(aes(x = associations_df3[, 2]),
             data = associations_df3, size = 3) +
  ggtitle("") +
  theme_gdocs()


associations <- findAssocs(tdm, "suicid", 0.2)
associations_df4 <- list_vect2df(associations)[, 2:3]

p4<- ggplot(associations_df4, aes(y = associations_df4[, 1])) +
  geom_point(aes(x = associations_df4[, 2]),
             data = associations_df4, size = 3) +
  ggtitle("") +
  theme_gdocs()


associations <- findAssocs(tdm, "neglect", 0.15)
associations_df5 <- list_vect2df(associations)[, 2:3]

p5<- ggplot(associations_df5, aes(y = associations_df5[, 1])) +
  geom_point(aes(x = associations_df5[, 2]),
             data = associations_df5, size = 3) +
  ggtitle("") +
  theme_gdocs()
```

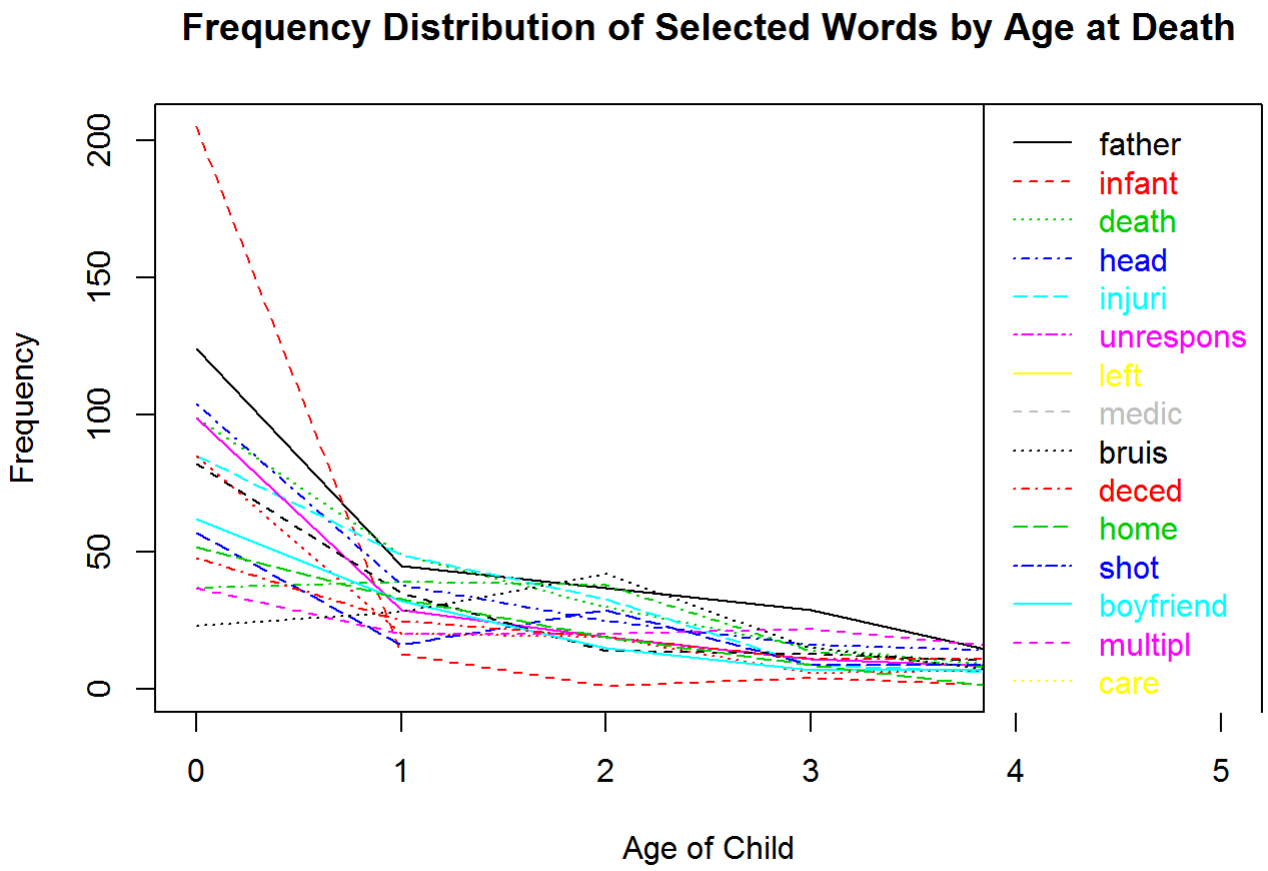
```
associations <- findAssocs(tdm, "drug", 0.18)
associations_df6 <- list_vect2df(associations)[, 2:3]

p6<- ggplot(associations_df6, aes(y = associations_df6[, 1])) +
  geom_point(aes(x = associations_df6[, 2]),
             data = associations_df6, size = 3) +
  ggtitle("") +
  theme_gdocs()

plot_grid(p1,p2,p3,p4,p5,p6, labels=c('Meth', 'Foster', 'Shot', 'Suicide', 'Neglect', 'Drug'))
```



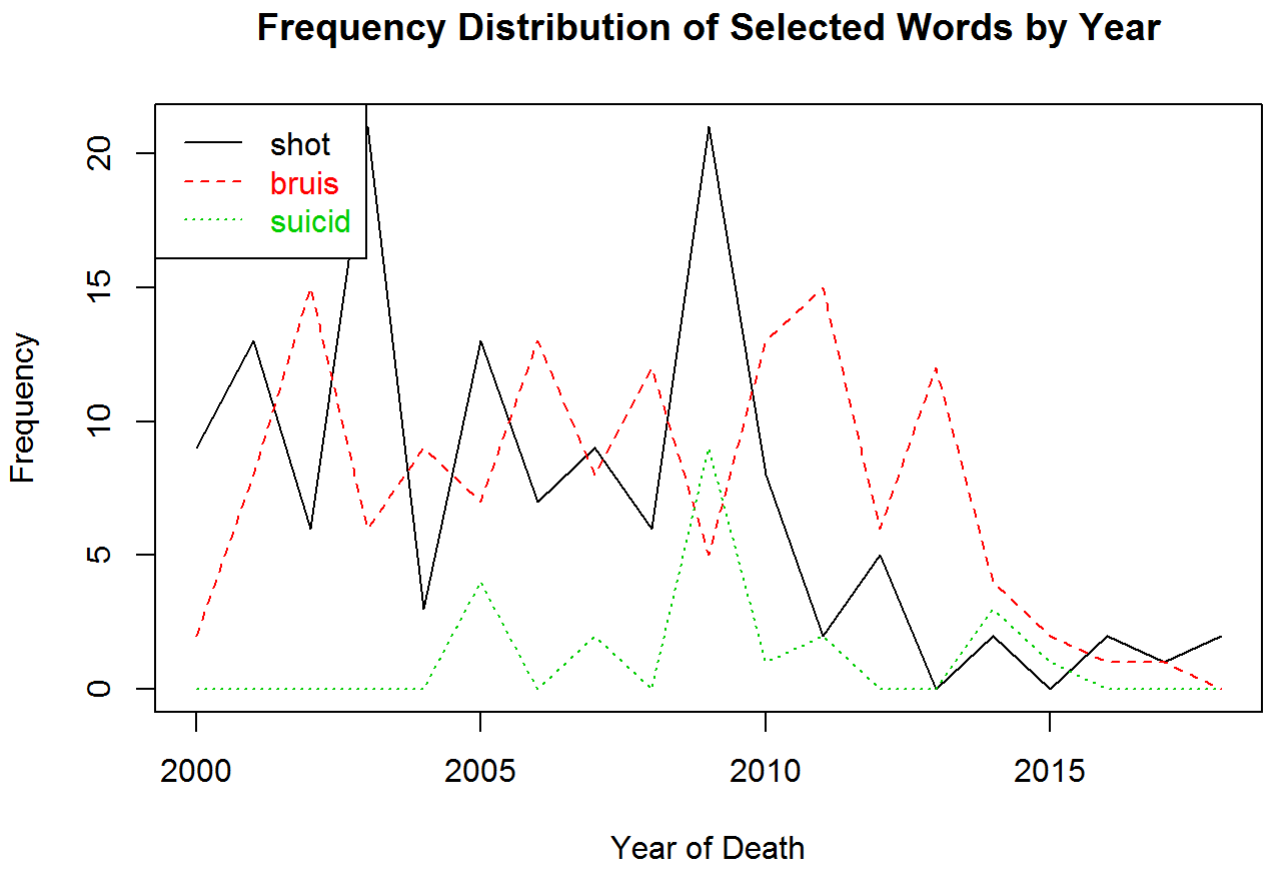

```
terms_to_observe <- c("father", "infant", "death", "head",
                      "injuri", "unrespons",
                      "left", "medic", "bruis",
                      "deced", "home", "shot", "boyfriend",
                      "multipl", "care")
DTM_reduced <- as.matrix(dtm[, terms_to_observe])
heatmap(t(DTM_reduced), Colv=NA, col = rev(heat.colors(256)), keep.dendro= FALSE, margins = c(2,
 15))
```



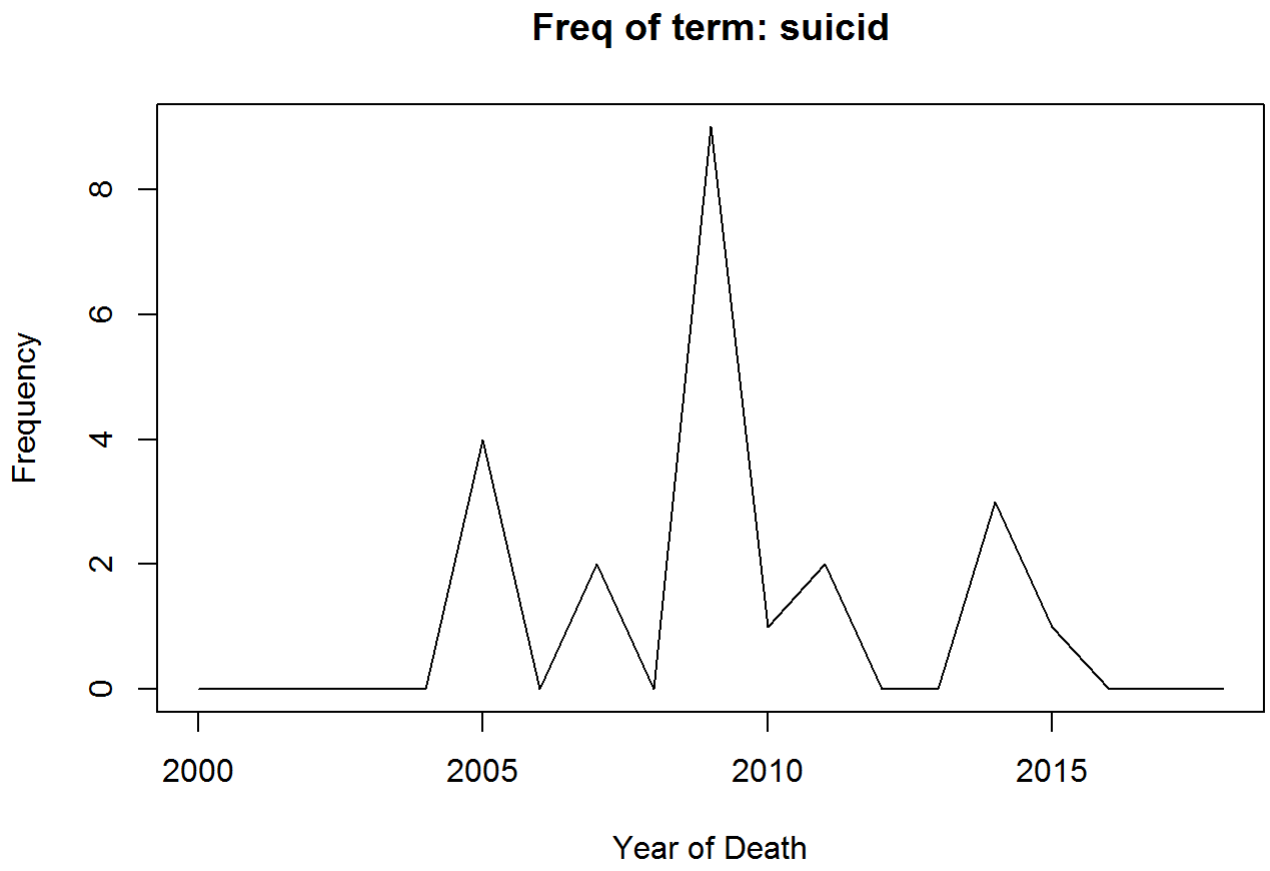

```
# Plot multiple frequencies not one at a time ala above
counts_per_age<- aggregate(DTM_reduced, by = list(age = deathdat$age), sum)
ages <- counts_per_age$age
frequencies <- counts_per_age[, terms_to_observe]


matplot(ages, frequencies, type = "l", xlab = "Age of Child", ylab = "Frequency", main = "Freque
ncy Distribution of Selected Words by Age at Death")

l <- length(terms_to_observe)
legend('topright', legend = terms_to_observe, col=1:l, text.col = 1:l, lty = 1:l)
```



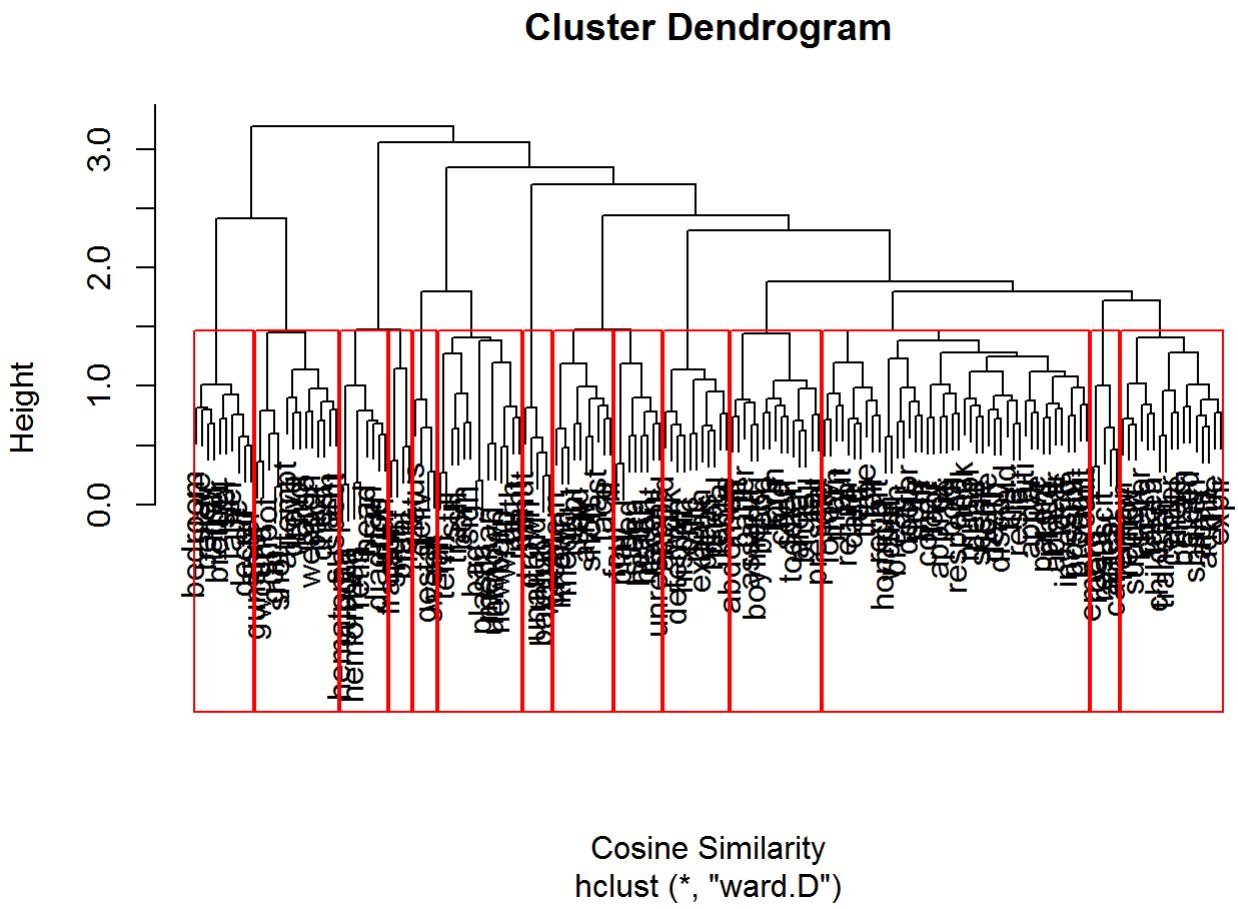

```
terms_to_observe <- c("shot",  "bruis", "suicid")
DTM_reduced <- as.matrix(dtm[, terms_to_observe])

deathdat$time_of_death <- as.Date(deathdat$time_of_death, format = "%m/%d/%Y")
deathdat$yearofdeath <- year(deathdat$time_of_death)
counts_per_year <- aggregate(DTM_reduced, by = list(deathdate = deathdat$yearofdeath), sum)
decades <- counts_per_year$deathdate
frequencies <- counts_per_year[, terms_to_observe]

# plot multiple frequencies
matplot(decades, frequencies, type = "l", xlab = "Year of Death", ylab = "Frequency", main = "Fr
equency Distribution of Selected Words by Year")

# add legend to the plot
l <- length(terms_to_observe)
legend('topleft', legend = terms_to_observe, col=1:l, text.col = 1:l, lty = 1:l)
```



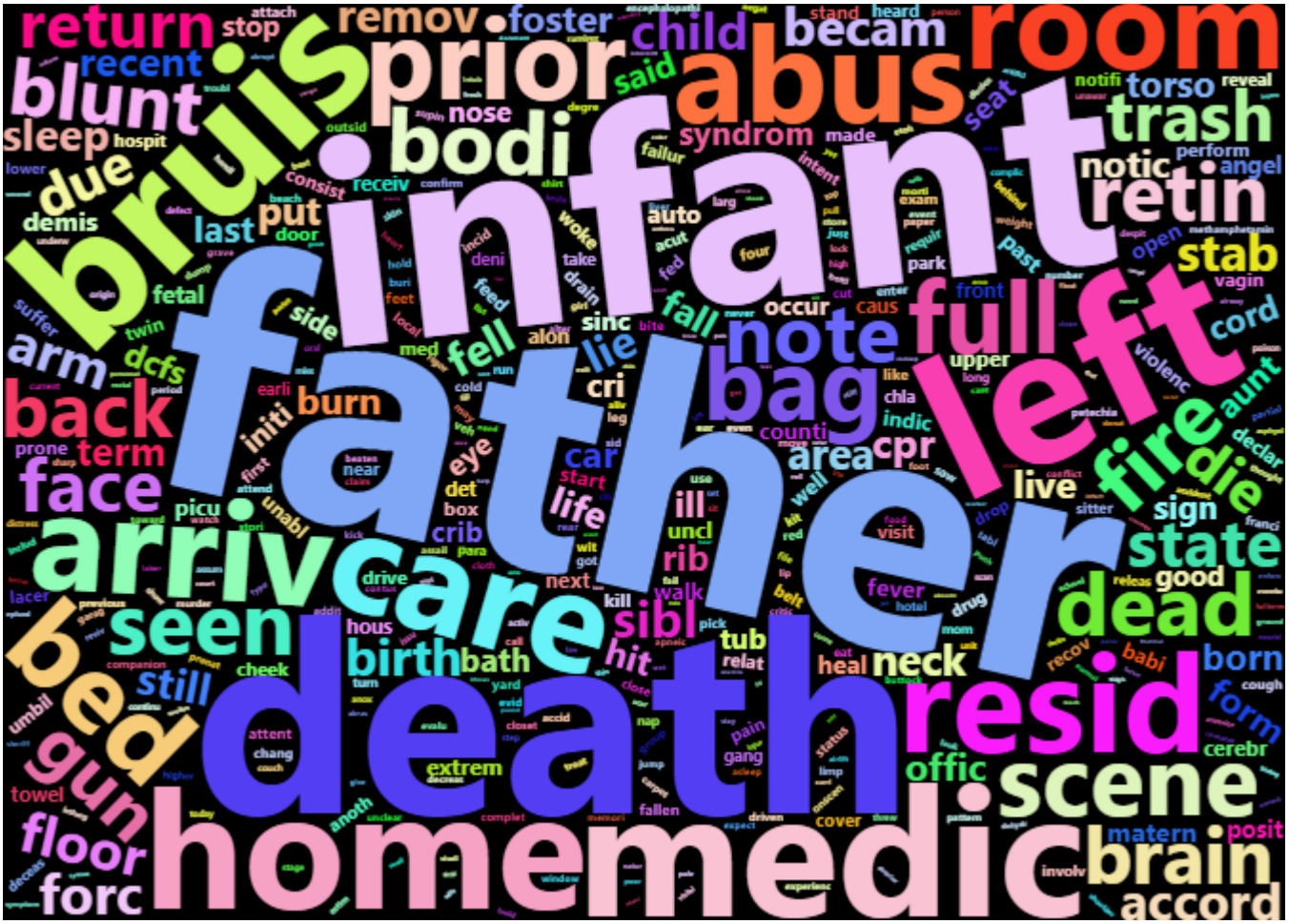

```
terms_to_observe <- c( "suicid")

DTM_reduced <- as.matrix(dtm[, terms_to_observe])

deathdat$time_of_death <- as.Date(deathdat$time_of_death, format = "%m/%d/%Y")
deathdat$yearofdeath <- year(deathdat$time_of_death)
counts_per_year <- aggregate(DTM_reduced, by = list(deathdate = deathdat$yearofdeath), sum)
decades <- counts_per_year$deathdate
frequencies <- counts_per_year[, terms_to_observe]

# plot multiple frequencies
matplot(decades, frequencies, type = "l", xlab = "Year of Death", ylab = "Frequency", main = pas
te("Freq of term", terms_to_observe, sep = ": "))
```




```
tdms <- removeSparseTerms(dtm, 0.95)
tf <- as.matrix(tdms)
idf <- log( ncol(tf) / ( 1 + rowSums(tf != 0) ) ) %>% diag()
xprod <- crossprod(tf, idf)
d1<- dist( xprod, method = "cosine" )
cluster1 <- hclust(d1, method = "ward.D")
plot.new()
plot(cluster1, xlab = "Cosine Similarity")
rect.hclust(cluster1, 14)
```




```
groups1 <- cutree(cluster1, 14)

dtms <- removeSparseTerms(dtm, 0.99)
freq <- colSums(as.matrix(dtm))
dark2 <- brewer.pal(6, "Dark2")
dtm2 <- as.matrix(dtms)
frequency <- colSums(dtm2)
frequency <- sort(frequency, decreasing=TRUE)
words <- names(frequency)
wordcloud2(data = data.frame(words, frequency), size = 1, ellipticity = .8,  color="random-ligh
t", backgroundColor="black")
```

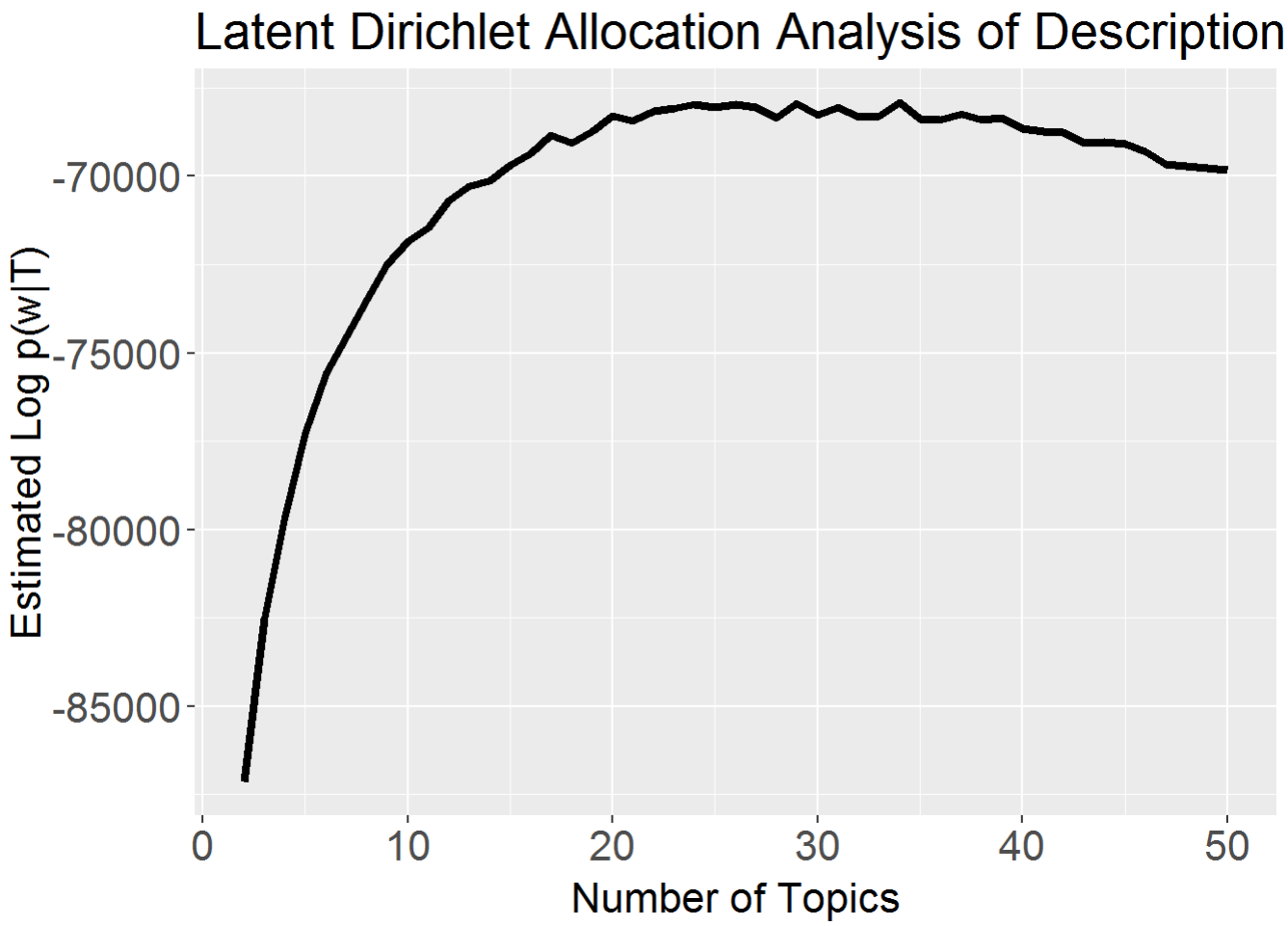



```
rowTotals <- apply(dtm , 1, sum) #Find the sum of words in each Document
dtm   <- dtm[rowTotals> 0, ]           #remove all docs without words
```

```
#############################################################################################################
set.seed(8745)

harmonicMean <- function(logLikelihoods, precision = 2000L) {
  llMed <- median(logLikelihoods)
  as.double(llMed - log(mean(exp(-mpfr(logLikelihoods, prec = precision) + llMed))))
}

seqk <- seq(2, 50, 1)
burnin <- 10
iter <- 10000
keep <- 50
system.time(fitted_many <- lapply(seqk, function(k) topicmodels::LDA(dtm, k = k, method = "Gibb
s",control = list(burnin = burnin, iter = iter, keep = keep) )))
```

```
##    user  system elapsed
## 1431.04    0.31 1441.97
```

```
logLiks_many <- lapply(fitted_many, function(L)  L@logLiks[-c(1:(burnin/keep))])
hm_many <- sapply(logLiks_many, function(h) harmonicMean(h))

ldaplot <- ggplot(data.frame(seqk, hm_many), aes(x=seqk, y=hm_many)) + geom_path(lwd=1.5) +
  theme(text = element_text(family= NULL),
        axis.title.y=element_text(vjust=1, size=16),
        axis.title.x=element_text(vjust=-.5, size=16),
        axis.text=element_text(size=16),
        plot.title=element_text(size=20)) +
  xlab('Number of Topics') +
  ylab("Estimated Log p(w|T)") +
  ggtitle("Latent Dirichlet Allocation Analysis of Description Surrounding Death")
ldaplot
```




```
seqk[which.max(hm_many)]
```

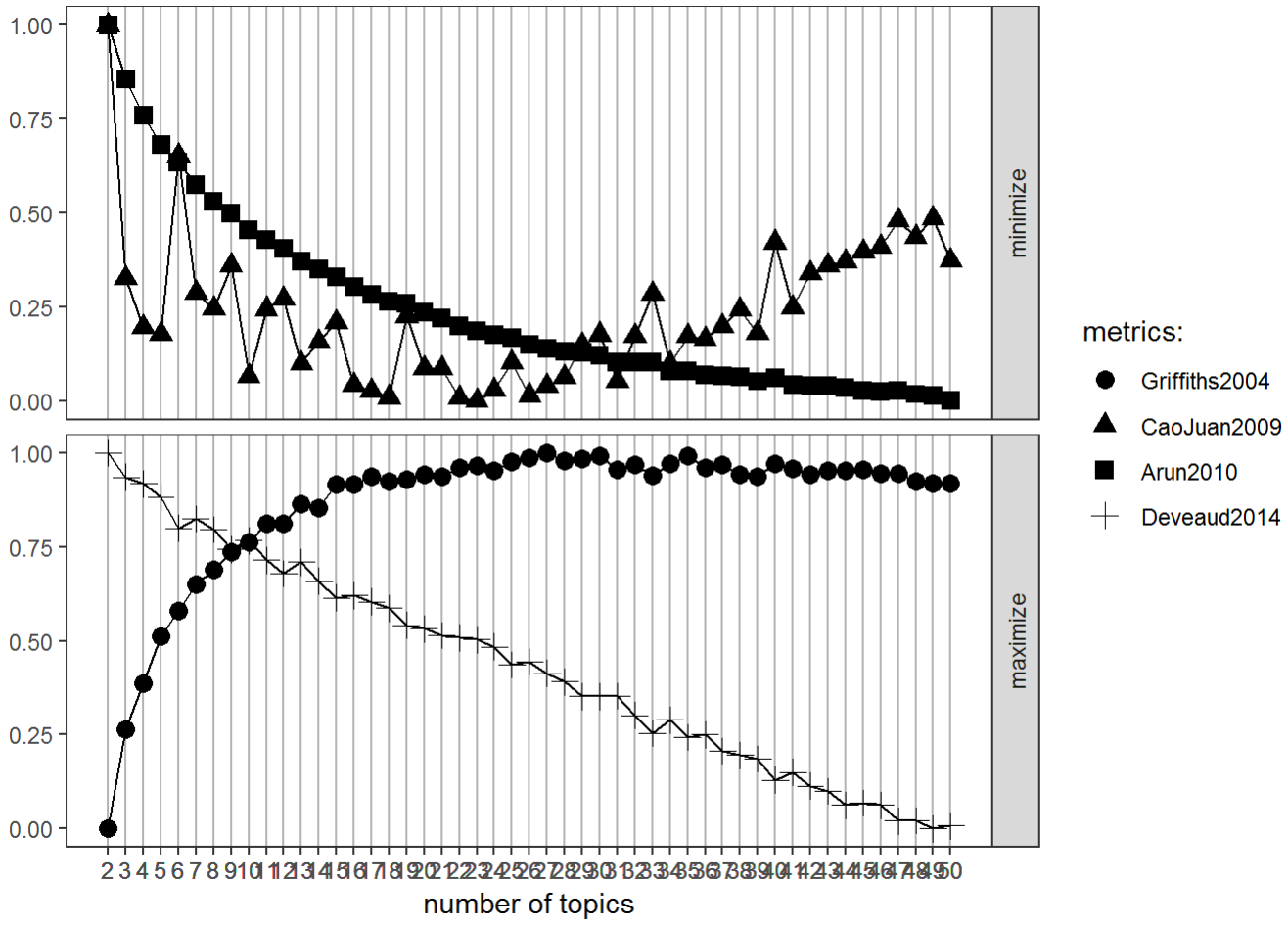
```
## [1] 34
```

```
system.time({
  tunes <- FindTopicsNumber(
    dtm = dtm,
    topics = c(2:50),
    metrics = c("Griffiths2004", "CaoJuan2009", "Arun2010", "Deveaud2014"),
    method = "Gibbs",
    control = list(seed = 12345),
    mc.cores = 4L,
    verbose = TRUE
  )
})
```

```
## fit models... done.
## calculate metrics:
##   Griffiths2004... done.
##   CaoJuan2009... done.
##   Arun2010... done.
##   Deveaud2014... done.
```

```
##    user  system elapsed
##    2.97    0.31  143.52
```

```
FindTopicsNumber_plot(tunes)
```



```
folds <- 5
splitfolds <- sample(1:folds, 23, replace = TRUE)
candidate_k <- c(2:50) # candidates for how many topics


# we parallelize by the different number of topics.  A processor is allocated a value
# of k, and does the cross-validation serially.  This is because it is assumed there
# are more candidate values of k than there are cross-validation folds, hence it
# will be more efficient to parallelise
library(doParallel)
```

```
## Warning: package 'doParallel' was built under R version 3.5.2
```

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

```
##
## Attaching package: 'foreach'
```

```
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
```

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

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

```
cluster <- makeCluster(detectCores(logical = TRUE) - 1) # leave one CPU spare...
registerDoParallel(cluster)

clusterEvalQ(cluster, {
  library(topicmodels)
})
```

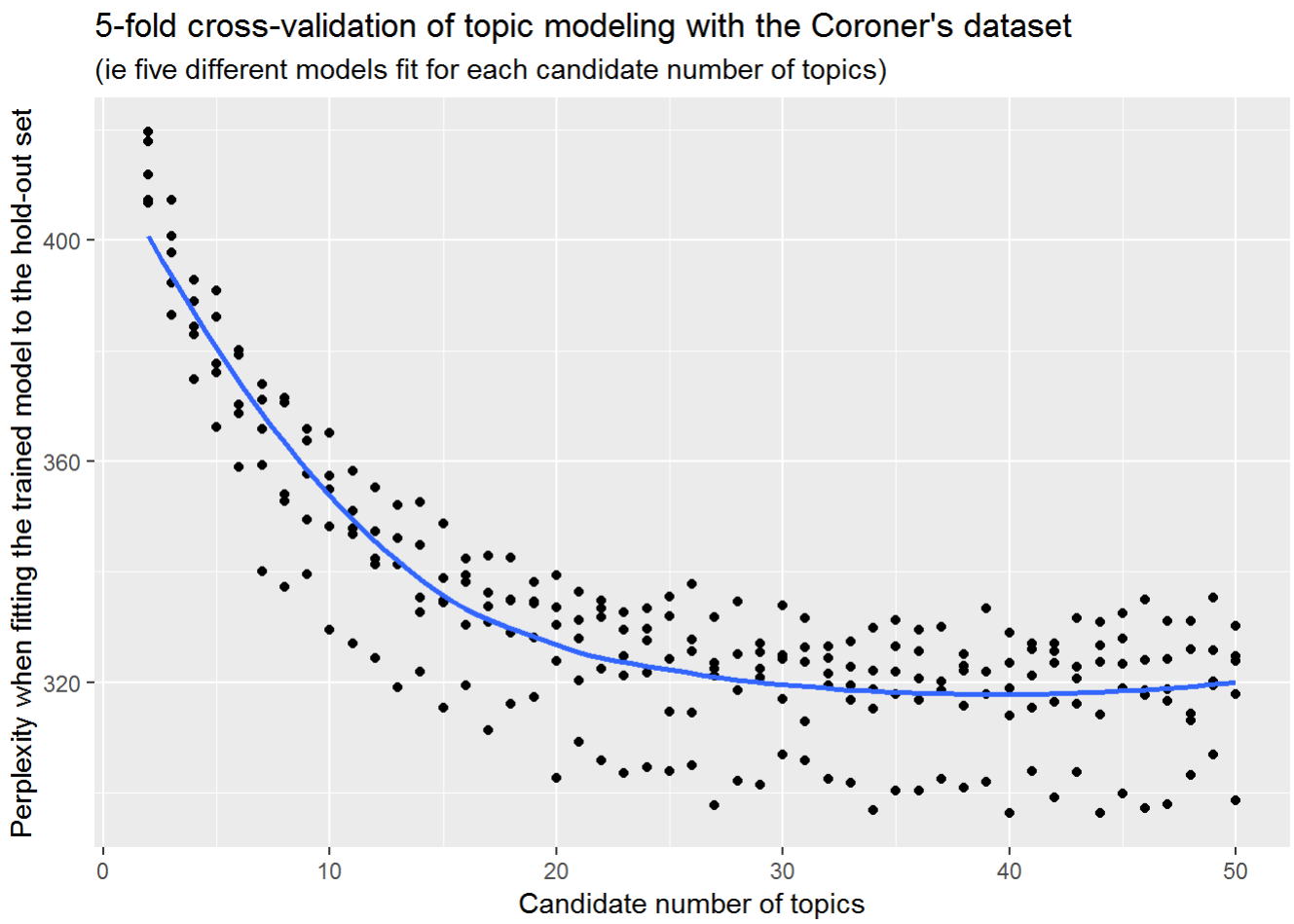
```
## [[1]]
## [1] "topicmodels" "stats"       "graphics"    "grDevices"   "utils"
## [6] "datasets"    "methods"     "base"
##
## [[2]]
## [1] "topicmodels" "stats"       "graphics"    "grDevices"   "utils"
## [6] "datasets"    "methods"     "base"
##
## [[3]]
## [1] "topicmodels" "stats"       "graphics"    "grDevices"   "utils"
## [6] "datasets"    "methods"     "base"
```

```
system.time({
  results <- foreach(j = 1:length(candidate_k), .combine = rbind) %dopar%{
    k <- candidate_k[j]
    results_1k <- matrix(0, nrow = folds, ncol = 2)
    colnames(results_1k) <- c("k", "perplexity")
    for(i in 1:folds){
      train_set <- dtm[splitfolds != i , ]
      valid_set <- dtm[splitfolds == i, ]

      fitted <- LDA(train_set, k = k, method = "Gibbs",
                    control = list(burnin = burnin, iter = iter, keep = keep) )
      results_1k[i,] <- c(k, perplexity(fitted, newdata = valid_set))
    }
    return(results_1k)
  }
})
```

```
##    user  system elapsed
##    0.08    0.02 3046.65
```

```
results_df <- as.data.frame(results)

ggplot(results_df, aes(x = k, y = perplexity)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  ggtitle("5-fold cross-validation of topic modeling with the Coroner's dataset",
          "(ie five different models fit for each candidate number of topics)") +
  labs(x = "Candidate number of topics", y = "Perplexity when fitting the trained model to the h
old-out set")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
##############################################################################################
```

```
#LDA model with 20 topics selected
K <- 20
seqk[which.max(hm_many)] <- K

lda_15 = LDA(dtm, k = K, method = 'Gibbs',
            control = list(nstart = 5, seed = list(1505,99,36,56,88), best = TRUE,
                           thin = 500, burnin = 4000, iter = 2000))

tmResult <- posterior(lda_15)
tmResult1 <- data.frame(t(tmResult$terms))

attributes(tmResult)
```

```
## $names
## [1] "terms"  "topics"
```

```
nTerms(dtm)
```

```
## [1] 678
```

```
# topics are probability distribtions over the entire vocabulary
beta <- tmResult$terms   # get beta from results
dim(beta)
```

```
## [1]  20 678
```

```
rowSums(beta)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
##  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

```
nDocs(dtm)
```

```
## [1] 437
```

```
theta <- tmResult$topics
dim(theta)
```

```
## [1] 437  20
```

```
rowSums(theta)[1:10]
```

```
## 2000-03406 2011-04686 2000-05204 2000-05618 2003-07410 2003-05944
##          1          1          1          1          1          1
## 2003-07423 2003-00694 2001-05221 2017-09303
##          1          1          1          1
```

```
terms(lda_15, 10)
```

```
##       Topic 1    Topic 2     Topic 3   Topic 4   Topic 5     Topic 6
##  [1,] "bag"      "unrespons" "deced"   "father"  "bed"       "femal"
##  [2,] "trash"    "male"      "famili"  "area"    "histori"   "fire"
##  [3,] "plastic"  "prior"     "resid"   "domest"  "face"      "arriv"
##  [4,] "scene"    "respond"   "death"   "due"     "sleep"     "stab"
##  [5,] "newborn"  "play"      "sister"  "histori" "unrespons" "accord"
##  [6,] "place"    "suspect"   "bedroom" "fell"    "lie"       "polic"
##  [7,] "dumpster" "state"     "dead"    "ill"     "said"      "work"
##  [8,] "cord"     "cpr"       "brother" "cheek"   "blood"     "determin"
##  [9,] "wrap"     "approx"    "appar"   "place"   "injuri"    "pleas"
## [10,] "locat"    "foul"      "suicid"  "throat"  "night"     "offic"
##       Topic 7  Topic 8    Topic 9 Topic 10  Topic 11    Topic 12
##  [1,] "head"   "admit"    "room"  "medic"   "hemorrhag" "left"
##  [2,] "injuri" "shaken"   "death" "parent"  "fractur"   "bathtub"
##  [3,] "blunt"  "charg"    "emerg" "known"   "subdur"    "return"
##  [4,] "forc"   "abus"     "arriv" "histori" "retin"     "drown"
##  [5,] "floor"  "taken"    "short" "note"    "diagnos"   "water"
##  [6,] "aunt"   "due"      "vomit" "center"  "hematoma"  "unattend"
##  [7,] "neck"   "head"     "mark"  "death"   "brain"     "minut"
##  [8,] "follow" "foster"   "start" "problem" "injuri"    "bath"
##  [9,] "hand"   "syndrom"  "eye"   "staff"   "skull"     "tub"
## [10,] "cri"    "transfer" "taken" "recent"  "rib"       "grandmoth"
##       Topic 13  Topic 14  Topic 15  Topic 16  Topic 17  Topic 18
##  [1,] "infant"  "care"    "multipl" "home"    "shot"    "bruis"
##  [2,] "discov"  "toddler" "vehicl"  "possibl" "wound"   "boyfriend"
##  [3,] "remov"   "home"    "die"     "appear"  "gun"     "bodi"
##  [4,] "live"    "report"  "seat"    "homicid" "gunshot" "abus"
##  [5,] "morn"    "seen"    "three"   "unknown" "back"    "abdomen"
##  [6,] "posit"   "child"   "car"     "death"   "arm"     "sexual"
##  [7,] "present" "fall"    "drove"   "victim"  "fire"    "batter"
##  [8,] "woke"    "injuri"  "back"    "breath"  "shoot"   "notic"
##  [9,] "earli"   "form"    "auto"    "appar"   "weapon"  "numer"
## [10,] "batter"  "choke"   "condit"  "dead"    "suspect" "assault"
##       Topic 19   Topic 20
##  [1,] "deliv"    "arrest"
##  [2,] "fetus"    "full"
##  [3,] "week"     "resuscit"
##  [4,] "gestat"   "cardiac"
##  [5,] "birth"    "term"
##  [6,] "bleed"    "brought"
##  [7,] "prematur" "fell"
##  [8,] "matern"   "picu"
##  [9,] "pregnant" "remain"
## [10,] "abdomin"  "becam"
```

```
topics(lda_15)
```

```
## 2000-03406 2011-04686 2000-05204 2000-05618 2003-07410 2003-05944
##          4         19         17         17         18         17
## 2003-07423 2003-00694 2001-05221 2017-09303 2007-08130 2003-05135
##         17          3          1          2         17          8
## 2000-00954 2003-05136 2003-00807 2000-07890 2002-07287 2004-07398
##         19          8          1          4         12          1
## 2016-08617 2003-08083 2003-06000 2003-03348 2001-04351 2005-00261
##         18          2          7          1          8          2
## 2003-05445 2006-03426 2002-03022 2007-06033 2001-01504 2003-05704
##         18          1         19         17          1         12
## 2009-07076 2011-04160 2007-05728 2011-00472 2000-03623 2003-04093
##         17         14          3         19         11         17
## 2002-00163 2001-03957 2004-06698 2018-00677 2007-02412 2004-09674
##         19          5         20         17         16         13
## 2007-02413 2001-09493 2009-04953 2001-06360 2003-09997 2000-06743
##         17         17          7         18          1         15
## 2001-09269 2000-02569 2011-01579 2013-08954 2004-08525 2005-04925
##          1          1         18          7          1         16
## 2000-05723 2012-00889 2002-05949 2004-09691 2011-03524 2016-04992
##          2         19          3         17         17         17
## 2001-03814 2016-07938 2006-01951 2002-09276 2002-04475 2002-07146
##          7          2          1          1         18         18
## 2000-00925 2004-07209 2004-02801 2011-01524 2002-01470 2002-04073
##          1         12         17          6         16         17
## 2010-04531 2005-06226 2001-08035 2004-00319 2005-03143 2002-01471
##          5         17         19          2         17          6
## 2001-03810 2007-02887 2016-06687 2003-07485 2005-04332 2006-08897
##          6         18          2          8          8         17
## 2000-01321 2005-03844 2014-00007 2007-03024 2000-07144 2005-06337
##          8         16         19          9          2         13
## 2010-00638 2002-02275 2009-00390 2003-03415 2004-05451 2005-00674
##          6          3         17         17         11         11
## 2012-03734 2005-06478 2007-03182 2010-07377 2015-07369 2005-07452
##         17         19         13         19         19         12
## 2015-07302 2003-09988 2001-08247 2005-07727 2000-04070 2006-00268
##          1          7          1          9          8          7
## 2001-07273 2000-08650 2000-00498 2004-01084 2008-06598 2000-00322
##         10          4         12         12          1         17
## 2017-01958 2004-08807 2003-00134 2001-09398 2005-06756 2000-04901
##          4          5          1         16         11          8
## 2014-01699 2014-03736 2006-09627 2003-09180 2002-06327 2001-06302
##          6          7         11         11          8          1
## 2012-00822 2014-04505 2008-00086 2008-03230 2007-01636 2010-03069
##         13          5          4         17          8         19
## 2005-03609 2001-07196 2010-07204 2005-07693 2003-09681 2006-07396
##         14         16         17          1          2         17
## 2007-03296 2001-01256 2003-01378 2003-04368 2012-08167 2010-02454
##          3          2          5         17          1         17
## 2007-02362 2008-08638 2005-07317 2007-06808 2005-04647 2001-05784
##         17         10          8         17         18         17
## 2000-06218 2000-03104 2005-05514 2009-00711 2003-03606 2014-00161
##         11         17         14         17         17         11
## 2001-08246 2004-07023 2001-01080 2002-09265 2007-05790 2004-04100
```

```
##         19          5          4          1          5          4
## 2002-08223 2001-00491 2004-04101 2008-07702 2004-03462 2002-05130
##          2         17          4         14          2          1
## 2004-09701 2001-00504 2001-07386 2004-08304 2002-01046 2000-00214
##          6          1          2          2         18          5
## 2007-06231 2003-05936 2000-08897 2001-05042 2012-07367 2000-03734
##         18         11          7          6         11          4
## 2011-00412 2009-07195 2002-00699 2012-06988 2006-06824 2004-03134
##         13         17          6         12         12         11
## 2005-08764 2006-08188 2013-05972 2005-04835 2012-03647 2001-03032
##          5         18          9         19         11         14
## 2002-08615 2001-07446 2011-03868 2007-07612 2012-06543 2003-04807
##         15          1         13          2         11          8
## 2002-08026 2005-04474 2001-08503 2006-01009 2001-03829 2006-09172
##         18         17         16          1          5         11
## 2006-07288 2007-04640 2001-06422 2013-06891 2011-06080 2015-01508
##          5         17          8         10         11         11
## 2014-06126 2015-05103 2015-02837 2009-06610 2007-07383 2006-07280
##         19          1         19         17         12          9
## 2002-06331 2005-07806 2009-07246 2011-06093 2005-00086 2005-00348
##         12         19         19         10          1         16
## 2003-00979 2008-03414 2000-05474 2008-03460 2002-03869 2008-04869
##          1         14         12          9         18         20
## 2010-01418 2001-05518 2006-09689 2011-07056 2003-03102 2008-07710
##          7         12         19          7          1          7
## 2000-03665 2010-06620 2004-07197 2010-04829 2008-08217 2000-02351
##          8         18          1         12         19         12
## 2011-05311 2009-02076 2006-07262 2011-00236 2012-05144 2016-05232
##          5          1         11          5          5         20
## 2011-04465 2005-04601 2001-07050 2005-04946 2002-01809 2005-00307
##         11         20          6         11          2         18
## 2000-04633 2002-05441 2002-02241 2005-09774 2003-00450 2008-06611
##          1         15          2         12         13         11
## 2009-08421 2009-08426 2005-01672 2005-07796 2006-02812 2014-02073
##         19          9          4         19          1         18
## 2007-02013 2008-07364 2006-09739 2002-04390 2006-04266 2009-02495
##         19         12         11          1         12         19
## 2006-06442 2008-03505 2014-05515 2003-03919 2008-05255 2009-05459
##         11         19          7         18          4          9
## 2001-02316 2012-05548 2004-02591 2009-07805 2008-07965 2015-00618
##          3         11          5          7         14          4
## 2006-05445 2008-05170 2002-04289 2016-07644 2006-04999 2002-06265
##         18         15         15          6          5          4
## 2003-00584 2011-02474 2007-05511 2006-04928 2003-06406 2012-01101
##         17         11         19          7          2         12
## 2015-03848 2013-01623 2004-06492 2001-07738 2010-01531 2013-00509
##          3         11         10         19          1         11
## 2013-07150 2016-01134 2002-03584 2013-06647 2001-03885 2010-07634
##         18         17         17          5         18         11
## 2009-05699 2003-07938 2003-06937 2004-08594 2002-09479 2017-01562
##          5          4          8         18         12         19
## 2008-02963 2006-04422 2004-00225 2004-00059 2003-06062 2006-07869
##         11         19          6          5          1         12
## 2016-03333 2009-02405 2013-04228 2009-07304 2001-02311 2006-01444
```

```
##         15         13          7         11         19         19
## 2010-02014 2010-05324 2013-00536 2005-07951 2006-00761 2007-02583
##          9         19         11          4          1         15
## 2002-02000 2002-02434 2004-03220 2001-08569 2008-05126 2008-01564
##          9         12         11         17         14         11
## 2001-05959 2006-03201 2010-03970 2001-02143 2016-01807 2013-06529
##         18         11         17         19          2         20
## 2003-01219 2002-08049 2004-03788 2010-02944 2003-07096 2009-07772
##          1          2          8          1         12          5
## 2012-01200 2013-03038 2011-08189 2011-07080 2009-00725 2011-03465
##          3          5         19         14          3         14
## 2001-05191 2005-05995 2006-08854 2009-07082 2003-03762 2001-07035
##          8         18         11         19          8          7
## 2002-01331 2010-01568 2009-00724 2016-04448 2008-02044 2008-07777
##          2         20          3          6         11          7
## 2010-06121 2006-08809 2011-05460 2015-08740 2009-00726 2010-07474
##         18          7         14          2          3         18
## 2001-00546 2011-03324 2009-02765 2007-02577 2001-03029 2009-00727
##         10         20         11         15         17          3
## 2010-03900 2013-06900 2016-07904 2002-02527 2009-07966 2015-02622
##          5         19         15          5          7         11
## 2000-06539 2005-03041 2008-06804 2009-00555 2015-04438 2010-07692
##          1         16         11          5         11          9
## 2013-03431 2003-01986 2015-02244 2001-01756 2008-06337 2000-00690
##         10          7         10          2         20          2
## 2011-04319 2010-07178 2014-03509 2002-00552 2002-01220 2014-03508
##         10          6          6         12          1          6
## 2010-02293 2011-05355 2013-08152 2004-09814 2014-02690 2004-00702
##         12         14         14         19         10         11
## 2014-03507 2013-06348 2004-02477 2015-04061 2013-01593 2006-07757
##          6         15          2          7         10         11
## 2011-01225 2014-06816 2015-00044 2012-02107 2006-08062 2016-07166
##         11         11         10         10         11         18
## 2005-05181 2007-06627 2011-07620 2003-04492 2010-05961 2000-02110
##          6         11         11         11          5         12
## 2008-01599 2001-04821 2012-05030 2014-08278 2009-08510 2011-06658
##         17          1          5         15         11         13
## 2013-06258 2017-07153 2012-06884 2016-02369 2017-06374
##          6         10          6         11         11
```

```
x<-data.frame(topics(lda_15))

x1 <- cbind(deathdat, x)
x1 <- cbind(deathdat, theta)
write.csv(x1, "F:/GSU/x1.csv", row.names = FALSE)


top10terms_15 = as.matrix(terms(lda_15,10))
top10terms_15
```

```
##       Topic 1    Topic 2     Topic 3   Topic 4   Topic 5     Topic 6
##  [1,] "bag"      "unrespons" "deced"   "father"  "bed"       "femal"
##  [2,] "trash"    "male"      "famili"  "area"    "histori"   "fire"
##  [3,] "plastic"  "prior"     "resid"   "domest"  "face"      "arriv"
##  [4,] "scene"    "respond"   "death"   "due"     "sleep"     "stab"
##  [5,] "newborn"  "play"      "sister"  "histori" "unrespons" "accord"
##  [6,] "place"    "suspect"   "bedroom" "fell"    "lie"       "polic"
##  [7,] "dumpster" "state"     "dead"    "ill"     "said"      "work"
##  [8,] "cord"     "cpr"       "brother" "cheek"   "blood"     "determin"
##  [9,] "wrap"     "approx"    "appar"   "place"   "injuri"    "pleas"
## [10,] "locat"    "foul"      "suicid"  "throat"  "night"     "offic"
##       Topic 7  Topic 8    Topic 9 Topic 10  Topic 11    Topic 12
##  [1,] "head"   "admit"    "room"  "medic"   "hemorrhag" "left"
##  [2,] "injuri" "shaken"   "death" "parent"  "fractur"   "bathtub"
##  [3,] "blunt"  "charg"    "emerg" "known"   "subdur"    "return"
##  [4,] "forc"   "abus"     "arriv" "histori" "retin"     "drown"
##  [5,] "floor"  "taken"    "short" "note"    "diagnos"   "water"
##  [6,] "aunt"   "due"      "vomit" "center"  "hematoma"  "unattend"
##  [7,] "neck"   "head"     "mark"  "death"   "brain"     "minut"
##  [8,] "follow" "foster"   "start" "problem" "injuri"    "bath"
##  [9,] "hand"   "syndrom"  "eye"   "staff"   "skull"     "tub"
## [10,] "cri"    "transfer" "taken" "recent"  "rib"       "grandmoth"
##       Topic 13  Topic 14  Topic 15  Topic 16  Topic 17  Topic 18
##  [1,] "infant"  "care"    "multipl" "home"    "shot"    "bruis"
##  [2,] "discov"  "toddler" "vehicl"  "possibl" "wound"   "boyfriend"
##  [3,] "remov"   "home"    "die"     "appear"  "gun"     "bodi"
##  [4,] "live"    "report"  "seat"    "homicid" "gunshot" "abus"
##  [5,] "morn"    "seen"    "three"   "unknown" "back"    "abdomen"
##  [6,] "posit"   "child"   "car"     "death"   "arm"     "sexual"
##  [7,] "present" "fall"    "drove"   "victim"  "fire"    "batter"
##  [8,] "woke"    "injuri"  "back"    "breath"  "shoot"   "notic"
##  [9,] "earli"   "form"    "auto"    "appar"   "weapon"  "numer"
## [10,] "batter"  "choke"   "condit"  "dead"    "suspect" "assault"
##       Topic 19   Topic 20
##  [1,] "deliv"    "arrest"
##  [2,] "fetus"    "full"
##  [3,] "week"     "resuscit"
##  [4,] "gestat"   "cardiac"
##  [5,] "birth"    "term"
##  [6,] "bleed"    "brought"
##  [7,] "prematur" "fell"
##  [8,] "matern"   "picu"
##  [9,] "pregnant" "remain"
## [10,] "abdomin"  "becam"
```

```
lda.topics_15 = as.matrix(topics(lda_15))

write.csv(lda.topics_15,file = paste('F:/GSU/LDAGibbs',15,'DocsToTopics.csv'))
write.csv(x1,file = paste('F:/GSU/LDAGibbs',15,'DocsToTopics.csv'))
summary(as.factor(lda.topics_15[,1]))
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
## 42 25 12 15 26 19 21 18 10 14 50 25  9 13 11  9 46 27 37  8
```

```
x2 <- data.frame(rownames(lda.topics_15), lda.topics_15[,1])
x2<- cbind(x2, x1)
topicprob_15 = as.matrix(lda_15@gamma)

write.csv(topicprob_15, file = paste('F:/GSU/LDAGibbs', 15, 'DoctToTopicProb.csv'))
head(topicprob_15,1)
```

```
##           [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## [1,] 0.04098361 0.04098361 0.07377049 0.09016393 0.04098361 0.05737705
##           [,7]       [,8]       [,9]      [,10]      [,11]      [,12]
## [1,] 0.04098361 0.04098361 0.04098361 0.05737705 0.04098361 0.05737705
##          [,13]      [,14]      [,15]      [,16]      [,17]      [,18]
## [1,] 0.04098361 0.04098361 0.04098361 0.04098361 0.04098361 0.09016393
##          [,19]      [,20]
## [1,] 0.04098361 0.04098361
```

```
topicprob_15[1:5,]
```

```
##           [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## [1,] 0.04098361 0.04098361 0.07377049 0.09016393 0.04098361 0.05737705
## [2,] 0.05932203 0.04237288 0.05932203 0.04237288 0.05932203 0.05932203
## [3,] 0.04032258 0.05645161 0.04032258 0.05645161 0.04032258 0.08870968
## [4,] 0.03731343 0.05223881 0.03731343 0.06716418 0.03731343 0.05223881
## [5,] 0.04545455 0.04545455 0.04545455 0.04545455 0.04545455 0.04545455
##           [,7]       [,8]       [,9]      [,10]      [,11]      [,12]
## [1,] 0.04098361 0.04098361 0.04098361 0.05737705 0.04098361 0.05737705
## [2,] 0.04237288 0.04237288 0.05932203 0.04237288 0.04237288 0.04237288
## [3,] 0.04032258 0.04032258 0.04032258 0.04032258 0.04032258 0.04032258
## [4,] 0.11194030 0.03731343 0.03731343 0.03731343 0.03731343 0.03731343
## [5,] 0.04545455 0.06363636 0.04545455 0.04545455 0.04545455 0.04545455
##          [,13]      [,14]      [,15]      [,16]      [,17]      [,18]
## [1,] 0.04098361 0.04098361 0.04098361 0.04098361 0.04098361 0.09016393
## [2,] 0.04237288 0.04237288 0.04237288 0.04237288 0.05932203 0.04237288
## [3,] 0.04032258 0.04032258 0.07258065 0.04032258 0.12096774 0.04032258
## [4,] 0.03731343 0.03731343 0.03731343 0.03731343 0.15671642 0.03731343
## [5,] 0.04545455 0.04545455 0.04545455 0.06363636 0.04545455 0.08181818
##          [,19]      [,20]
## [1,] 0.04098361 0.04098361
## [2,] 0.09322034 0.04237288
## [3,] 0.04032258 0.04032258
## [4,] 0.03731343 0.03731343
## [5,] 0.04545455 0.06363636
```

```
lda_15.topics <- topicmodels::topics(lda_15, 1)

lda_15.terms <- as.data.frame(topicmodels::terms(lda_15, 60), stringsAsFactors = FALSE)
lda_15.terms[1:5]
```

```
##        Topic 1   Topic 2     Topic 3        Topic 4     Topic 5
## 1          bag unrespons       deced         father         bed
## 2        trash      male      famili           area     histori
## 3      plastic     prior       resid         domest        face
## 4        scene   respond       death            due       sleep
## 5      newborn      play      sister        histori   unrespons
## 6        place   suspect     bedroom           fell         lie
## 7     dumpster     state        dead            ill        said
## 8         cord       cpr     brother          cheek       blood
## 9         wrap    approx       appar          place      injuri
## 10       locat      foul      suicid         throat       night
## 11       still       det        sibl            use        last
## 12       umbil    observ      believ           life       mouth
## 13       appar    breath     station         privat       check
## 14    placenta      crib        door            saw        nose
## 15      attach      note    neighbor          carri        good
## 16         put    extrem        life          femal       small
## 17        neck      drug        take           push      appear
## 18     obvious       sid     contact        asphyxi        dcfs
## 19 preliminari      para      intent           bite       fever
## 20      suffoc      stop        twin           drop      health
## 21         box     front         hit          evacu         ill
## 22        buri     indic       local       investig      pillow
## 23       grave      past      number           jail      recent
## 24       insid     reviv       deliv methamphetamin      diaper
## 25        born    condit       insid          notic       prone
## 26      closet      long   intervent          scene        note
## 27        dump     scene      murder          state         red
## 28    circumst  significe      poison          unabl       state
## 29      deceas   concern        even            air        cold
## 30        note      etoh       incid          chest        next
## 31     partial     femal      person        comatos        side
## 32    asphyxia      miss       state          fetus      provid
## 33     blanket  placenta        call           four     confirm
## 34         cut     torso     chronic           mani         leg
## 35        rear      food      declin          pleas       supin
## 36      reveal     heart      strike           poor        crib
## 37        belt       lip      struck          refer       event
## 38       evalu   pattern     violenc          resid       handl
## 39      examin       ran      beaten           side         lot
## 40    fullterm       yet       chest           auto      releas
## 41   stillborn      bite         cri         breath respiratori
## 42       unabl      girl        hear           cold     buttock
## 43       initi      like       medic          drive        fail
## 44        kill       med         mom           full   grandmoth
## 45        knee       nap        near           girl      mental
## 46     sheriff     pleas        note            leg      record
## 47       shook     polic         put           step      visibl
## 48      summon    releas respiratori         suffoc    conflict
## 49        work    summon      ventil          taken       death
## 50    artifici   support        evid       transfer    forehead
## 51     attempt  accident      franci            wet       hotel
## 52     confirm     admit        give           wife        left
```

```
## 53         cri     adult       labor          bruis        limp
## 54      effort    appear       occur      companion       morti
## 55       expir     brain    previous            die        open
## 56        feet   brought       scene          earli       place
## 57      inform   buttock     subject            eye       prior
## 58         lot   chronic     support          indic        sent
## 59         nap    discov       swell          paper      shower
## 60      receiv    doctor        tabl          posit        sibl
```

```
df <- data.frame(term = lda_15@terms)
colorVec = rep(c('red', 'skyblue'), length.out=nrow(df))
plots <- list()  # new empty list
for (i in 1:seqk[which.max(hm_many)]) {
  topic <- i
  df <- data.frame(term = lda_15@terms, p = exp(lda_15@beta[topic,]))
  df <- df[order(df$p, decreasing = TRUE),]
  df <- df[1:25,]
  p1 = wordcloud2(data = data.frame(df$term, df$p), color= colorVec, ellipticity = .6,  size=2,
 minRotation = -pi/6, maxRotation = -pi/6)
  plots[[i]] <- p1  # add each plot into plot list
}
multiplot(plotlist = plots, cols = 4)
```

```
ap_lda_td <- tidy(lda_15, matrix = "beta")
top_n(ap_lda_td, 10)
```

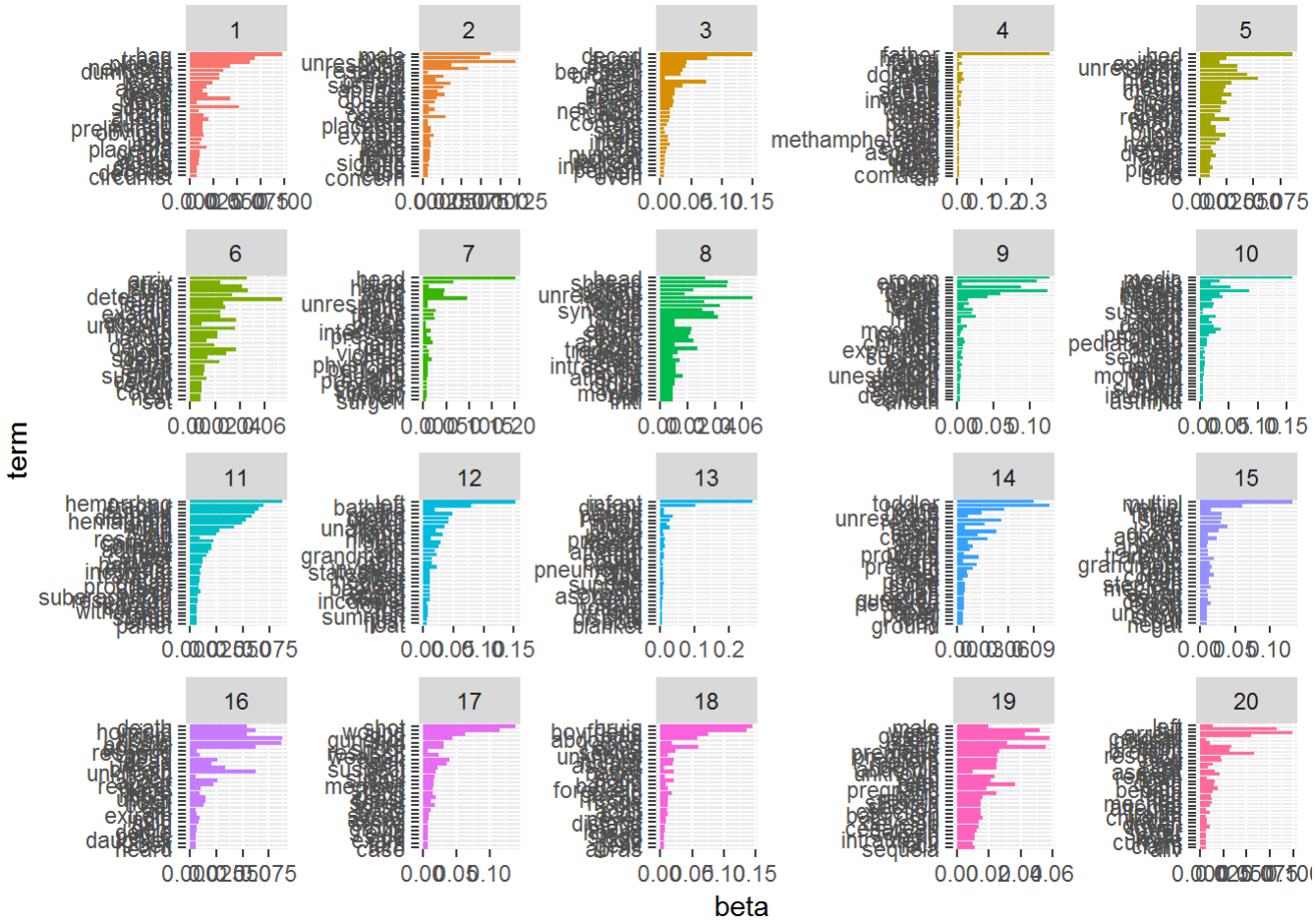
```
## Selecting by beta
```

```
## # A tibble: 10 x 3
##    topic term       beta
##    <int> <chr>     <dbl>
##  1    18 boyfriend 0.136
##  2    18 bruis     0.145
##  3     3 deced     0.150
##  4     4 father    0.374
##  5     7 head      0.202
##  6    13 infant    0.271
##  7    12 left      0.153
##  8    10 medic     0.161
##  9    15 multipl   0.132
## 10    17 shot      0.139
```

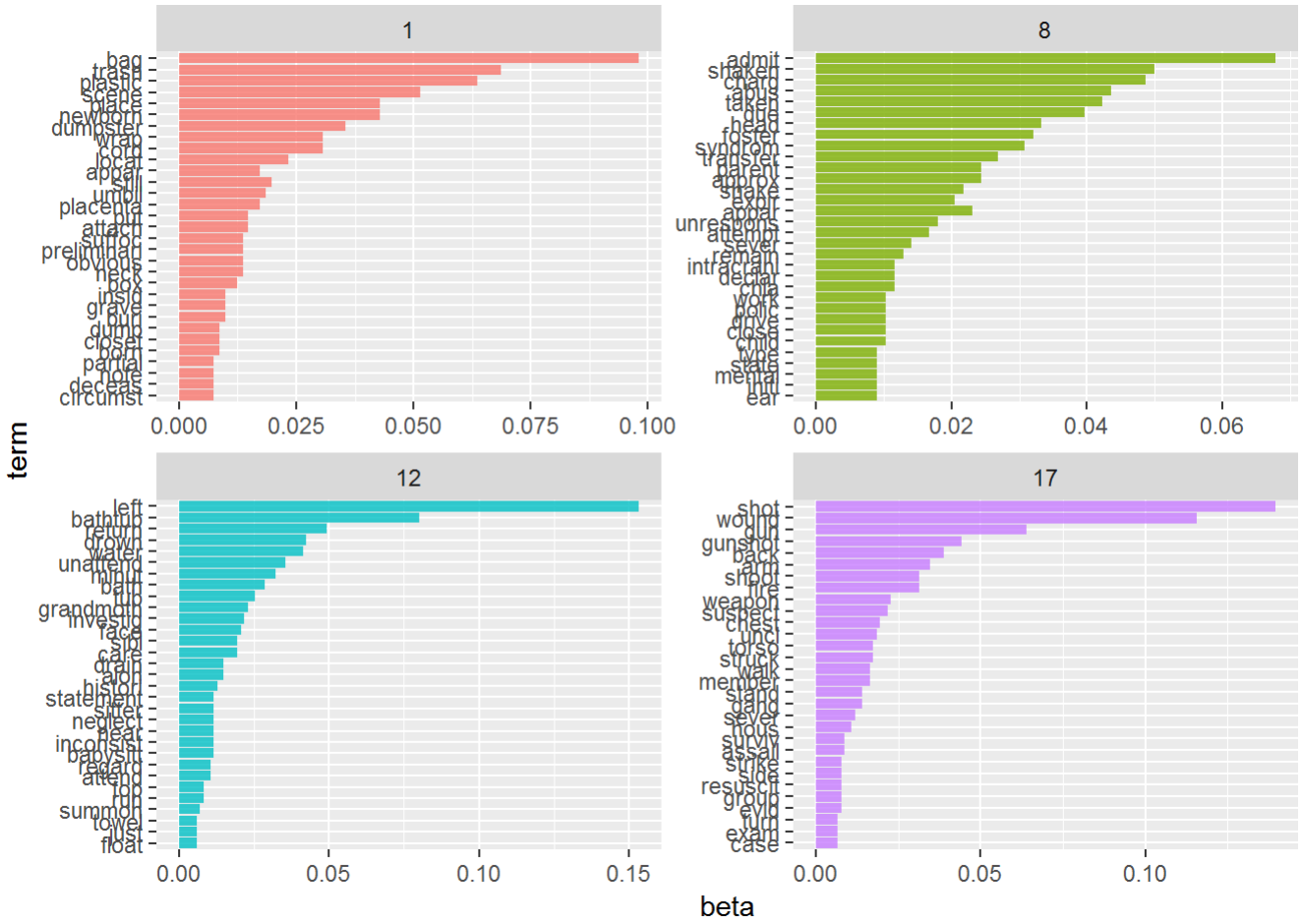
```
top_terms <- ap_lda_td %>%
  group_by(topic) %>%
  top_n(30, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)
top_terms
```

```
## # A tibble: 659 x 3
##    topic term       beta
##    <int> <chr>     <dbl>
##  1     1 bag      0.0979
##  2     1 trash    0.0686
##  3     1 plastic  0.0637
##  4     1 scene    0.0515
##  5     1 newborn  0.0429
##  6     1 place    0.0429
##  7     1 dumpster 0.0356
##  8     1 cord     0.0307
##  9     1 wrap     0.0307
## 10     1 locat    0.0234
## # ... with 649 more rows
```

```
top_terms %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  coord_flip()
```




```
top_terms %>%
 filter(topic==1|topic==8|topic==12|topic==17) %>%
  mutate(term = reorder(term, beta)) %>%
  ggplot(aes(term, beta, fill = factor(topic))) +
  geom_bar(alpha = 0.8, stat = "identity", show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free", ncol = 2) +
  coord_flip()
```




```
top5termsPerTopic <- terms(lda_15, 5)
topicNames <- apply(top5termsPerTopic, 2, paste, collapse=" ")
x3 <- sample(1:437, 10)

exampleIds <- c(x3)
lapply(docs[exampleIds], as.character)
```

```
## $`character(0)`
## [1] "mo male crawl back door lead pool area pool green walga dirti zero visibl water approx m
in infant reisdenti swim pool bruis note forehead report left unattend visit home boyfriend left
unattend coupl intercours recal hear cri check infant crawl consider distanc home open door back
yard swim pool bruis note child forehead scratch seen cheek"
##
## $`character(0)`
## [1] "mo male sinc jusi th shaken syndrom blunt head male alter mental status subdur hematoma
retin hemorrhag injuri occur care sitter"
##
## $`character(0)`
## [1] "yr male parent privat auto full arrest fever sever taken clinic tylenol releas foul play
medhx matern abus histori good health intermitt lowgrad fever symptom coat tongu doctor dx viral
infect sent home ibuprofen night death play toy night vomit sever morn emerg room arriv cardiac
arrest"
##
## $`character(0)`
## [1] "unknown cod ill traumat injuri known medic issu infant care unlicens babysitt woke nap g
asp last breath pass caregiv sought medic treathment emerg room note transvers lacer insid lower
lip petechi hemorrhag seen insid lower eyelid form platero"
##
## $`character(0)`
## [1] "blunt forc er home full arrestp mom father bottl feed stop breathingpron hospitalbabi bo
rn full term full term known medic problem ill injuri difficulti feed seem signific associ cyano
si last well check weight gain littl slow parent increas supplement feed infant fed father arm s
tretch stop breath cpr immedi start return conscious en rout infant cardiac monitor becam bradyc
ard lost puls death short arriv emerg room"
##
## $`character(0)`
## [1] "bht assault unknown object famili violenc femal head chest abdomin due physic abus stepf
ath sexual assault exam wnl"
##
## $`character(0)`
## [1] "gun shot wound loc vermont fetal death matern death mult gun shot wound deced cc involv
gunfir incid sidewalk friend gunman open fire struck multipl initi despit medic intervent expir
henc met demis well discov emerg csection nonviabl weigh g evid gunshot"
##
## $`character(0)`
## [1] "yo male co abdomin pain morn minut parent unrespons bathroom andtran full arrest prior m
edic histori parent batter undernourish live home pregnant boyfriend father toddler brother lace
r insid upper lip numer cutan injuri stage heal pale appear dehydr abdomen de la torr"
##
## $`character(0)`
## [1] "er approx bruis bodi full arrest put lifesupport brain dead bater report fell bed care y
r polic struck head belt taken fire station full arrest staff resuscit place vent admit icu mult
ipl bruis differ retin hemorrhag note abus notifi unstabl ct head deterior declar brain dead dr
s"
##
## $`character(0)`
## [1] "blunt head neck brought children transfer anoth higher care initi brought decreas moveme
nt yellow color skin bruse dayold male transfer children long beach st franci medic center highe
r care decreas movement jaundic complexion observ facial lacer left shoulder polic pleas cooper
biddl prior postmortem examin"
```

```
library("reshape2")
library("ggplot2")
N <- length(exampleIds)
attr(lda_15, "alpha")
```

```
## [1] 2.5
```

```
tmResult <- posterior(lda_15)
theta <- tmResult$topics
beta <- tmResult$terms
topicNames <- apply(terms(lda_15, 5), 2, paste, collapse = " ")  # reset topicnames

# get topic proportions form example documents
topicProportionExamples <- theta[exampleIds,]
colnames(topicProportionExamples) <- topicNames
vizDataFrame <- melt(cbind(data.frame(topicProportionExamples), document = factor(1:N)), variabl
e.name = "topic", id.vars = "document")

ggplot(data = vizDataFrame, aes(topic, value, fill = document), ylab = "proportion") +
  geom_bar(stat="identity") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  coord_flip() +
  facet_wrap(~ document, ncol = N)
```

```
topicNames <- apply(lda::top.topic.words(beta, 5, by.score = T), 2, paste, collapse = " ")

countsOfPrimaryTopics <- rep(0, 20)
names(countsOfPrimaryTopics) <- topicNames
for (i in 1:nDocs(dtm)) {
  topicsPerDoc <- theta[i, ] # select topic distribution for document i (this will be used in SO
M)
  primaryTopic <- order(topicsPerDoc, decreasing = TRUE)[1]
  countsOfPrimaryTopics[primaryTopic] <- countsOfPrimaryTopics[primaryTopic] + 1
}
sort(countsOfPrimaryTopics, decreasing = TRUE)
```

```
## hemorrhag fractur subdur retin diagnos
##                                     50
##            shot wound gun gunshot back
##                                     46
##        bag trash plastic scene newborn
##                                     42
##          deliv fetus week gestat birth
##                                     37
##      boyfriend bruis bodi abdomen abus
##                                     27
##            bed histori face sleep said
##                                     26
##      unrespons male prior respond play
##                                     25
##        left bathtub return drown water
##                                     25
##           head injuri blunt forc floor
##                                     21
##           femal fire arriv stab accord
##                                     19
##            admit shaken charg due abus
##                                     18
##             father area domest due ill
##                                     15
##      medic parent known histori center
##                                     14
##          care toddler home report seen
##                                     13
##      deced famili resid sister bedroom
##                                     12
##          multipl vehicl die seat three
##                                     11
##           room emerg death arriv short
##                                     10
##          infant discov remov live morn
##                                      9
##    possibl home homicid appear unknown
##                                      9
##      arrest full cardiac resuscit term
##                                      8
```

```
# What are the most probable topics in the entire collection?
topicProportions <- colSums(theta) / nDocs(dtm)  # mean probablities over all paragraphs
names(topicProportions) <- topicNames     # assign the topic names we created before
sort(topicProportions, decreasing = TRUE) # show summed proportions in decreased order
```

```
## hemorrhag fractur subdur retin diagnos
##                             0.05382588
##            shot wound gun gunshot back
##                             0.05346916
##      boyfriend bruis bodi abdomen abus
##                             0.05123400
##      unrespons male prior respond play
##                             0.05099765
##           head injuri blunt forc floor
##                             0.05072127
##        left bathtub return drown water
##                             0.05055373
##            bed histori face sleep said
##                             0.05053085
##           femal fire arriv stab accord
##                             0.05026214
##          deliv fetus week gestat birth
##                             0.05021336
##        bag trash plastic scene newborn
##                             0.05019840
##      deced famili resid sister bedroom
##                             0.05003359
##          infant discov remov live morn
##                             0.04944998
##      arrest full cardiac resuscit term
##                             0.04930521
##          care toddler home report seen
##                             0.04926240
##      medic parent known histori center
##                             0.04910355
##          multipl vehicl die seat three
##                             0.04884169
##    possibl home homicid appear unknown
##                             0.04867658
##            admit shaken charg due abus
##                             0.04858437
##           room emerg death arriv short
##                             0.04776842
##             father area domest due ill
##                             0.04696777
```

```
# get mean topic proportions per decade
topic_proportion_per_decade <- aggregate(theta, by = list(Year = deathdat$yearofdeath), mean)
# set topic names to aggregated columns
colnames(topic_proportion_per_decade)[2:(K+1)] <- topicNames

# reshape data frame
vizDataFrame <- melt(topic_proportion_per_decade, id.vars = "Year")

# plot topic proportions per deacde as bar plot
require(pals)
```

```
## Loading required package: pals
```

```
## Loading required package: maps
```

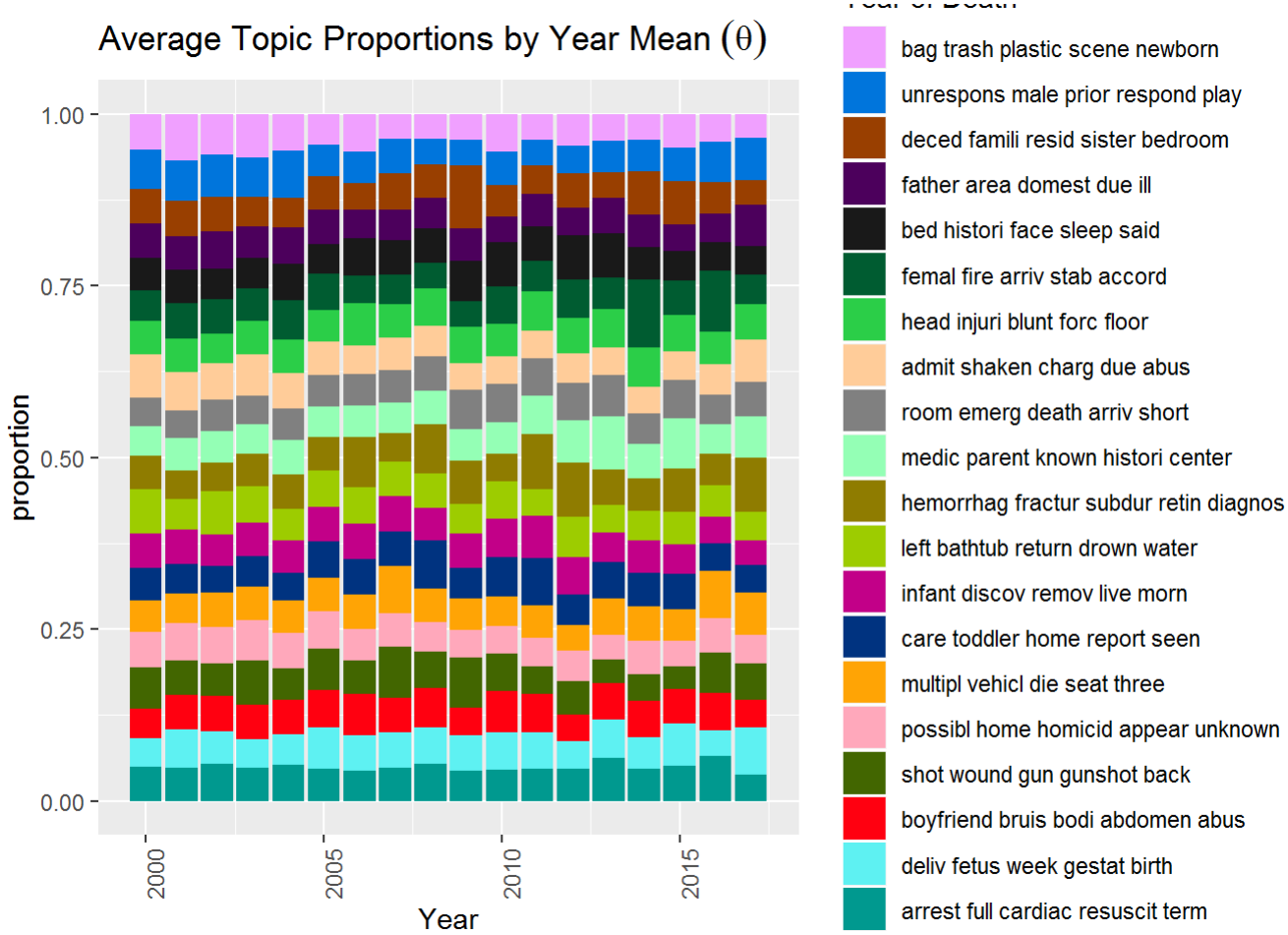
```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
## The following object is masked from 'package:kohonen':
##
##     map
```

```
## The following object is masked from 'package:cluster':
##
##     votes.repub
```

```
ggplot(subset(vizDataFrame, Year < 2018), aes(x=Year, y=value, fill=variable)) +
  geom_bar(stat = "identity") + ylab("proportion") +
  scale_fill_manual(values = paste0(alphabet(20), "FF"), name = "Year of Death") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) + ggtitle("Average Topic Proportions
 by Year Mean" ~(theta))
```




```
# What are the most probable topics in the entire collection?
topicProportions <- colSums(theta) / nDocs(dtm)  # mean probablities over all paragraphs
names(topicProportions) <- topicNames     # assign the topic names we created before
sort(topicProportions, decreasing = TRUE) # show summed proportions in decreased order
```

```
## hemorrhag fractur subdur retin diagnos
##                           0.05382588
##            shot wound gun gunshot back
##                           0.05346916
##      boyfriend bruis bodi abdomen abus
##                           0.05123400
##      unrespons male prior respond play
##                           0.05099765
##           head injuri blunt forc floor
##                           0.05072127
##        left bathtub return drown water
##                           0.05055373
##            bed histori face sleep said
##                           0.05053085
##           femal fire arriv stab accord
##                           0.05026214
##          deliv fetus week gestat birth
##                           0.05021336
##        bag trash plastic scene newborn
##                           0.05019840
##      deced famili resid sister bedroom
##                           0.05003359
##          infant discov remov live morn
##                           0.04944998
##      arrest full cardiac resuscit term
##                           0.04930521
##          care toddler home report seen
##                           0.04926240
##      medic parent known histori center
##                           0.04910355
##          multipl vehicl die seat three
##                           0.04884169
##    possibl home homicid appear unknown
##                           0.04867658
##            admit shaken charg due abus
##                           0.04858437
##           room emerg death arriv short
##                           0.04776842
##             father area domest due ill
##                           0.04696777
```

```
#topicToFilter <- 6  # you can set this manually ...
# ... or have it selected by a term in the topic name (e.g. 'children')
topicToFilter <- grep('gun', topicNames)[1]
topicThreshold <- 0.2
selectedDocumentIndexes <- which(theta[, topicToFilter] >= topicThreshold)
filteredCorpus <- docs[selectedDocumentIndexes]

# show length of filtered corpus
filteredCorpus
```

```
## <<VCorpus>>
## Metadata:  corpus specific: 0, document level (indexed): 2
## Content:  documents: 9
```

```
# The filtered corpus contains 59 documents related to "gun" at least 20% of the time
```

```
topicTerms <- tidyr::gather(lda_15.terms, Topic)
topicTerms <- cbind(topicTerms, Rank = rep(1:30))
topTerms <- dplyr::filter(topicTerms, Rank < 4)
topTerms <- dplyr::mutate(topTerms, Topic = stringr::word(Topic, 2))
topTerms$Topic <- as.numeric(topTerms$Topic)
topicLabel <- data.frame()
for (i in 1:20){
  z <- dplyr::filter(topTerms, Topic == i)
  l <- as.data.frame(paste(z[1,2], z[2,2], z[3,2], sep = " " ), stringsAsFactors = FALSE)
  topicLabel <- rbind(topicLabel, l)

}
colnames(topicLabel) <- c("Label")
topicLabel
```

```
##                       Label
## 1         bag trash plastic
## 2      unrespons male prior
## 3        deced famili resid
## 4        father area domest
## 5          bed histori face
## 6          femal fire arriv
## 7         head injuri blunt
## 8        admit shaken charg
## 9          room death emerg
## 10       medic parent known
## 11 hemorrhag fractur subdur
## 12      left bathtub return
## 13      infant discov remov
## 14        care toddler home
## 15       multipl vehicl die
## 16      home possibl appear
## 17           shot wound gun
## 18     bruis boyfriend bodi
## 19         deliv fetus week
## 20     arrest full resuscit
```

```
topicLabel$topic <- rep_len(1:20, length.out=20)

theta <- as.data.frame(topicmodels::posterior(lda_15)$topics)
head(theta[1:5])
```

```
##                     1          2          3          4          5
## 2000-03406 0.04098361 0.04098361 0.07377049 0.09016393 0.04098361
## 2011-04686 0.05932203 0.04237288 0.05932203 0.04237288 0.05932203
## 2000-05204 0.04032258 0.05645161 0.04032258 0.05645161 0.04032258
## 2000-05618 0.03731343 0.05223881 0.03731343 0.06716418 0.03731343
## 2003-07410 0.04545455 0.04545455 0.04545455 0.04545455 0.04545455
## 2003-05944 0.04237288 0.04237288 0.04237288 0.04237288 0.04237288
```

```
dev.off()
```

```
## null device
##           1
```

```
topicProbabilities <- as.data.frame(lda_15@gamma)
colnames(topicProbabilities) <- topicLabel$topic
d <- dist(t(topicProbabilities), method="correlation")
fit <- hclust(d=d, method="ward.D2")
plot.new()
plot(fit, hang=-1)
groups <- cutree(fit, k=5)
rect.hclust(fit, k=5, border="red")
```

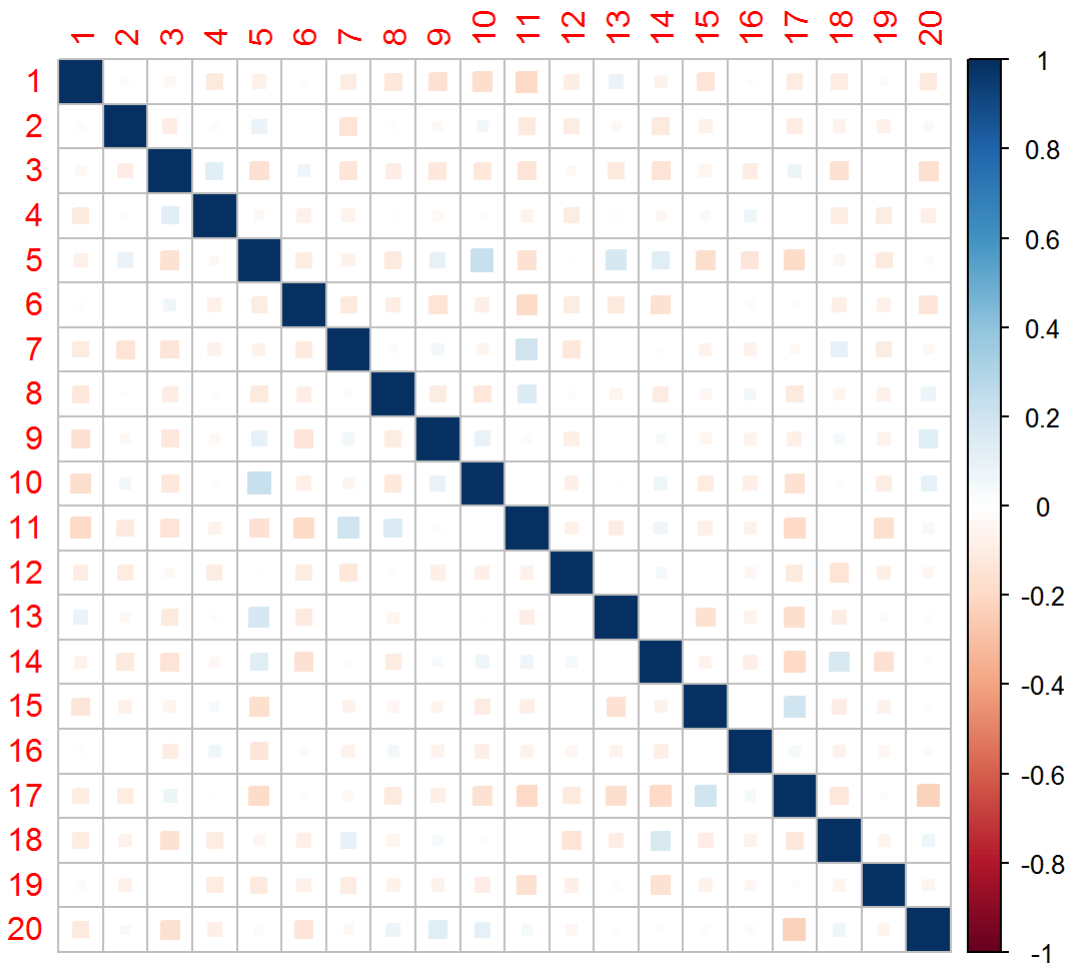
```
theta <- as.data.frame(topicmodels::posterior(lda_15)$topics)
head(theta[1:5])
```

```
##                     1          2          3          4          5
## 2000-03406 0.04098361 0.04098361 0.07377049 0.09016393 0.04098361
## 2011-04686 0.05932203 0.04237288 0.05932203 0.04237288 0.05932203
## 2000-05204 0.04032258 0.05645161 0.04032258 0.05645161 0.04032258
## 2000-05618 0.03731343 0.05223881 0.03731343 0.06716418 0.03731343
## 2003-07410 0.04545455 0.04545455 0.04545455 0.04545455 0.04545455
## 2003-05944 0.04237288 0.04237288 0.04237288 0.04237288 0.04237288
```

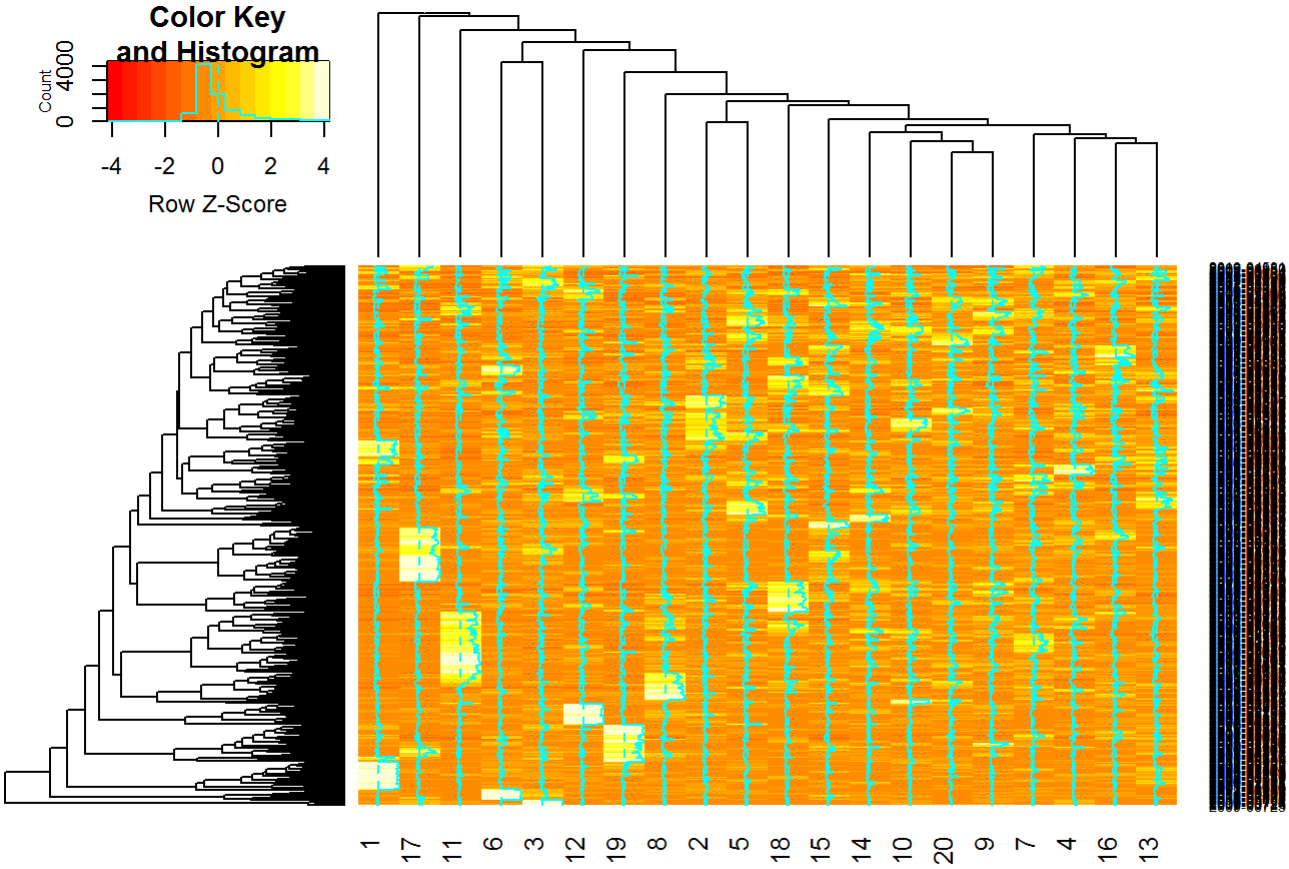
```
x <- as.data.frame(as.character(row.names(theta)), stringsAsFactors = FALSE)
colnames(x) <- c("doc_id")

theta2 <- cbind(x, theta)
theta2 <- dplyr::left_join(theta2, deathdat, by = "doc_id")

## Returns column means grouped by catergory
theta.mean.by <- by(theta2[, 2:21], theta2$doc_id, colMeans)
theta.mean <- do.call("rbind", theta.mean.by)

library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
c <- cor(theta.mean)
corrplot(c, method = "square")
```




```
topics <- topicmodels::posterior(lda_15, dtm)[["topics"]]
heatmap.2(topics, scale = "row")
```




```
post <- topicmodels::posterior(lda_15)


theta.mean.ratios <- theta.mean
for (ii in 1:nrow(theta.mean)) {
  for (jj in 1:ncol(theta.mean)) {
    theta.mean.ratios[ii,jj] <-
      theta.mean[ii,jj] / sum(theta.mean[ii,-jj])
  }
}
topics.by.ratio <- apply(theta.mean.ratios, 1, function(x) sort(x, decreasing = TRUE, index.retu
rn = TRUE)$ix)
topics.most.diagnostic <- topics.by.ratio[1,]
head(topics.most.diagnostic)
```

```
## 2000-00214 2000-00322 2000-00498 2000-00690 2000-00925 2000-00954
##          5         17         12          2          1         19
```

```
x5<-cbind(deathdat, as.matrix(topics(lda_15)))
```

```
library(LDAvis)
library(servr)
library(dplyr)
library(stringi)
library(Rmpfr)

topicmodels2LDAvis <- function(x, ...){
  post <- topicmodels::posterior(x)
  if (ncol(post[["topics"]]) < 3) stop("The model must contain > 2 topics")
  mat <- x@wordassignments
  LDAvis::createJSON(
    phi = post[["terms"]],
    theta = post[["topics"]],
    vocab = colnames(post[["terms"]]),
    doc.length = slam::row_sums(mat, na.rm = TRUE),
    term.frequency = slam::col_sums(mat, na.rm = TRUE)
  )
}

serVis(topicmodels2LDAvis(lda_15))
```

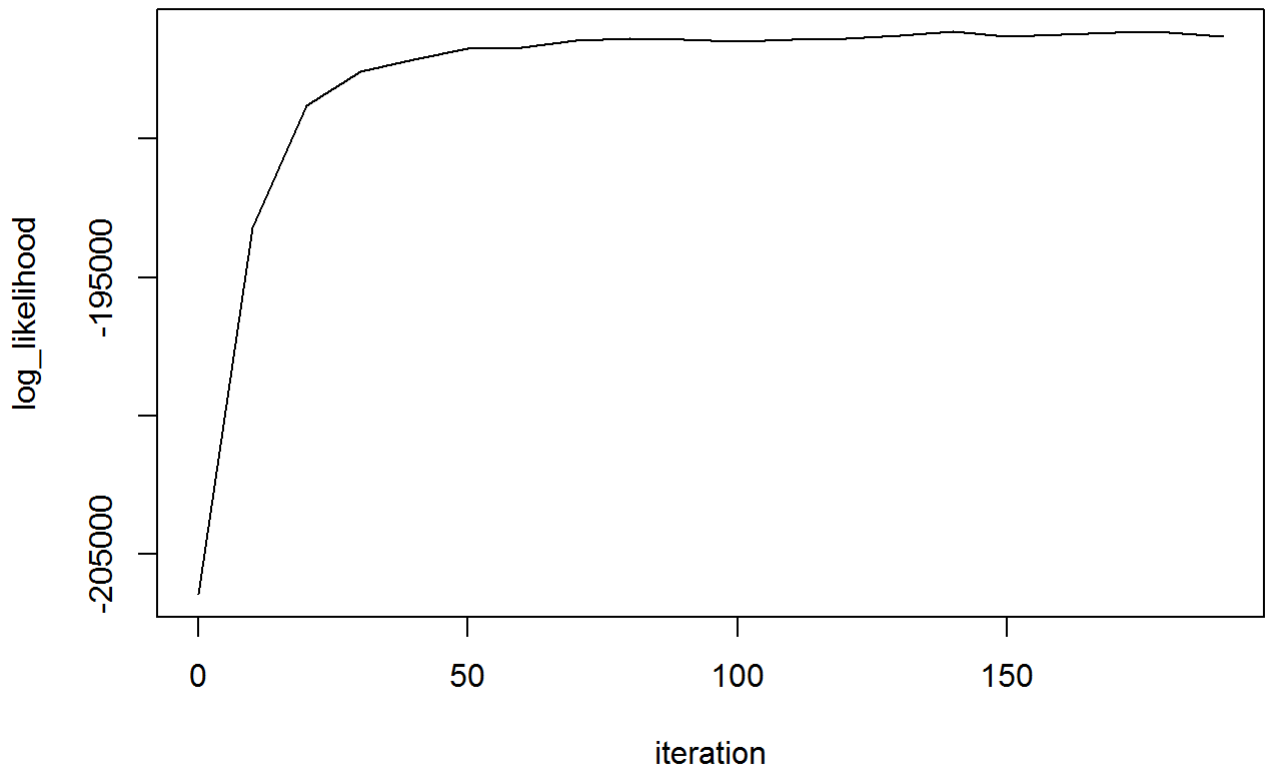
```
##############################################
library(textmineR)
set.seed(12345)


dtm <- CreateDtm(doc_vec = deathdat$text, # character vector of documents
                 doc_names = deathdat$doc_id , # document names
                 ngram_window = c(1, 3), # minimum and maximum n-gram length
                 stopword_vec = c(stopwords::stopwords("en"), # stopwords from tm
                                  myStopWords=s_words), # this is the default value
                 lower = TRUE, # lowercase - this is the default value
                 remove_punctuation = TRUE, # punctuation - this is the default
                 remove_numbers = TRUE, # numbers - this is the default
                 verbose = FALSE, # Turn off status bar for this demo
                 cpus = 2) # default is all available cpus on the system

dtm <- dtm[,colSums(dtm) > 2]

model <- FitLdaModel(dtm = dtm,
                     k = 20,
                     iterations = 200, # I usually recommend at least 500 iterations or more
                     burnin = 180,
                     alpha = 0.1,
                     beta = 0.05,
                     optimize_alpha = TRUE,
                     calc_likelihood = TRUE,
                     calc_coherence = TRUE,
                     calc_r2 = TRUE,
                     cpus = 2)
str(model)
```

```
## List of 9
##  $ phi           : num [1:20, 1:2393] 4.59e-05 5.91e-05 8.67e-05 7.29e-05 6.93e-05 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:20] "t_1" "t_2" "t_3" "t_4" ...
##   .. ..$ : chr [1:2393] "hands_feet" "inches_water" "bruise_noted_forehead" "fire_personnel"
...
##  $ theta         : num [1:437, 1:20] 0.09167 0.00714 0.00455 0.00333 0.01667 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:437] "2000-03406" "2011-04686" "2000-05204" "2000-05618" ...
##   .. ..$ : chr [1:20] "t_1" "t_2" "t_3" "t_4" ...
##  $ gamma         : num [1:20, 1:2393] 0.017 0.0169 0.0168 0.0167 0.0169 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:20] "t_1" "t_2" "t_3" "t_4" ...
##   .. ..$ : chr [1:2393] "hands_feet" "inches_water" "bruise_noted_forehead" "fire_personnel"
...
##  $ data          :Formal class 'dgCMatrix' [package "Matrix"] with 6 slots
##   .. ..@ i       : int [1:19555] 201 232 214 331 268 387 271 353 409 70 ...
##   .. ..@ p       : int [1:2394] 0 2 4 6 9 11 14 16 19 22 ...
##   .. ..@ Dim     : int [1:2] 437 2393
##   .. ..@ Dimnames:List of 2
##   .. .. ..$ : chr [1:437] "2000-03406" "2011-04686" "2000-05204" "2000-05618" ...
##   .. .. ..$ : chr [1:2393] "hands_feet" "inches_water" "bruise_noted_forehead" "fire_personne
l" ...
##   .. ..@ x       : num [1:19555] 2 1 1 2 2 1 1 1 1 2 ...
##   .. ..@ factors : list()
##  $ alpha         : num [1:20] 0.1057 0.0767 0.0518 0.0641 0.0688 ...
##  $ beta          : num [1:2393] 0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05 ...
##  $ log_likelihood:'data.frame':  20 obs. of  2 variables:
##   ..$ iteration     : num [1:20] 0 10 20 30 40 50 60 70 80 90 ...
##   ..$ log_likelihood: num [1:20] -206458 -193205 -188791 -187564 -187127 ...
##  $ coherence     : Named num [1:20] 0.1507 0.075 0.2319 0.2033 0.0851 ...
##   ..- attr(*, "names")= chr [1:20] "t_1" "t_2" "t_3" "t_4" ...
##  $ r2            : num 0.186
##  - attr(*, "class")= chr "lda_topic_model"
```

```
model$r2
```

```
## [1] 0.1863814
```

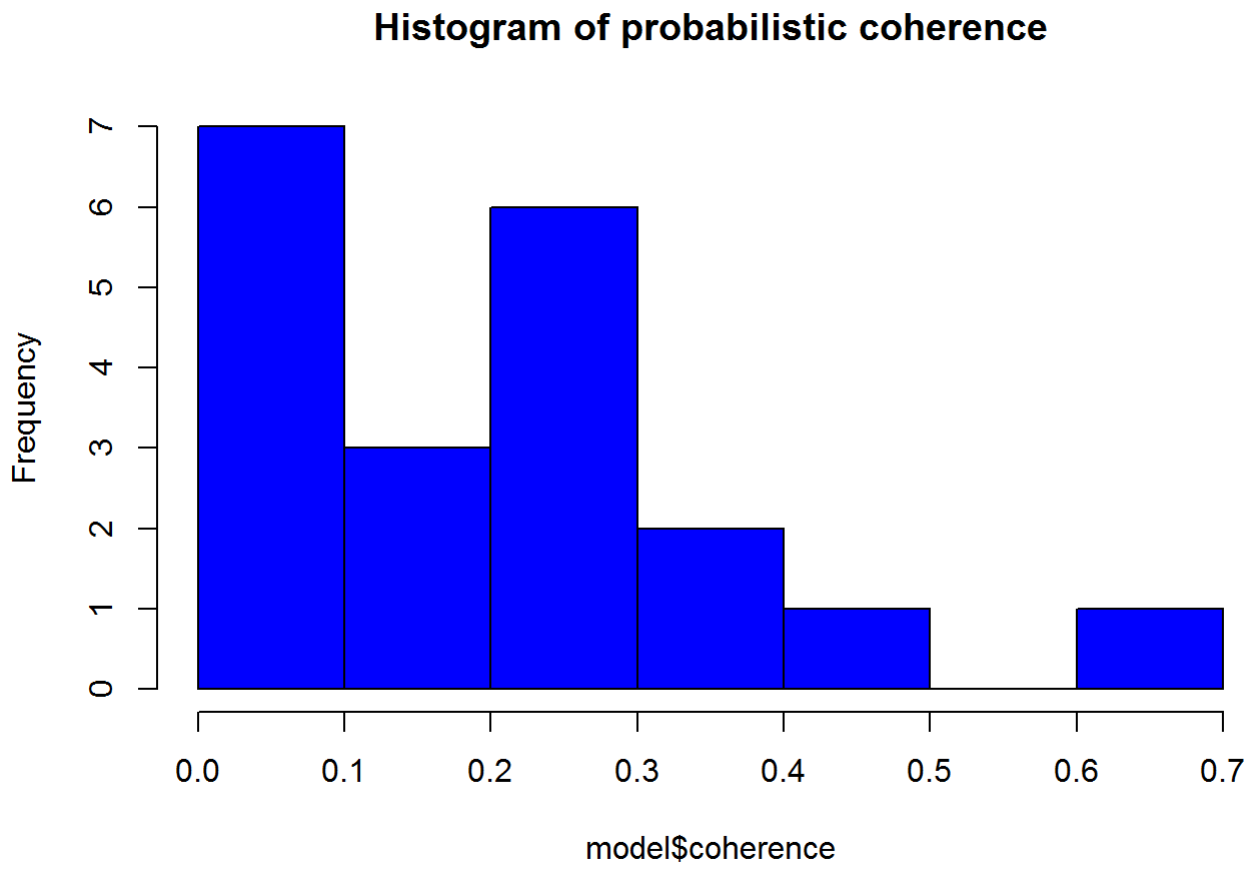
```
plot(model$log_likelihood, type = "l")
```




```
summary(model$coherence)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.01106 0.08512 0.17698 0.20903 0.28571 0.69410
```

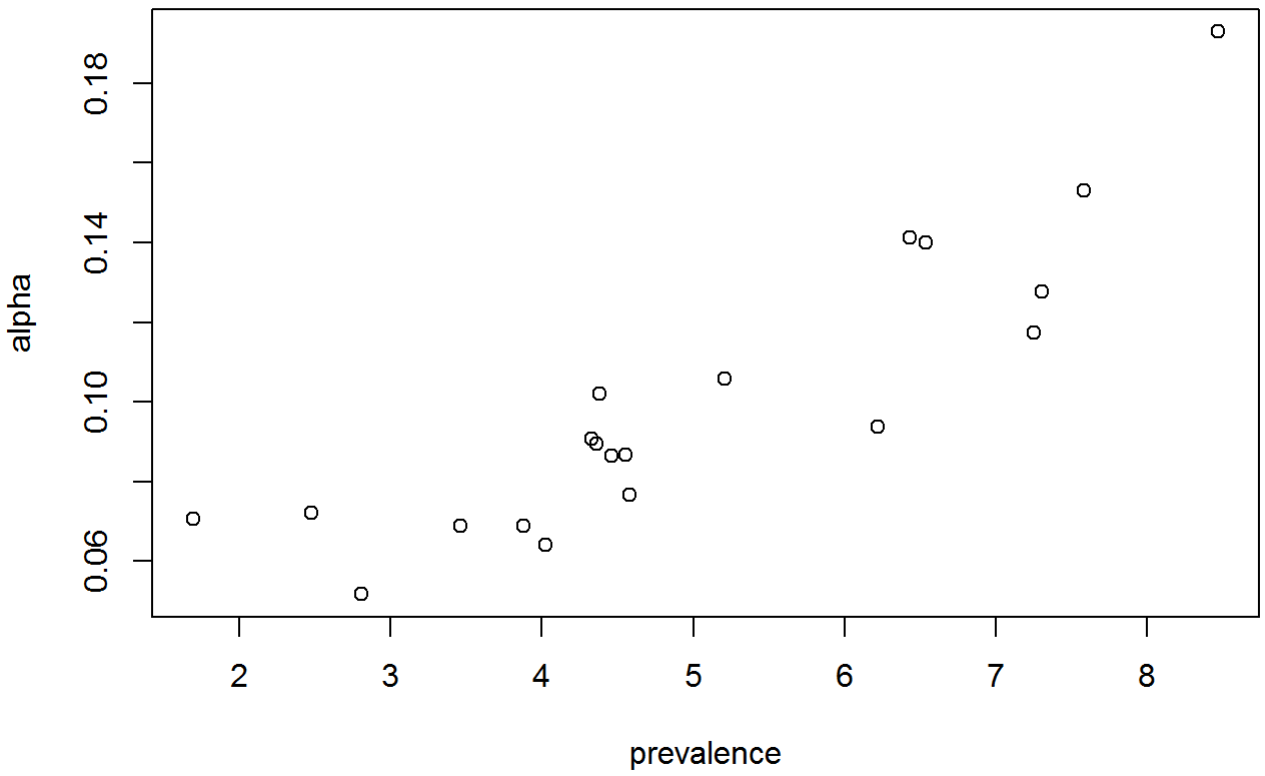
```
hist(model$coherence,
     col= "blue",
     main = "Histogram of probabilistic coherence")
```




```
model$top_terms <- GetTopTerms(phi = model$phi, M = 5)
head(t(model$top_terms))
```

```
##     [,1]        [,2]        [,3]               [,4]
## t_1 "boyfriend" "bruises"   "toddler"          "body"
## t_2 "home"      "fire"      "residence"        "family"
## t_3 "domestic"  "delivered" "domestic_violence" "violence"
## t_4 "father"    "shaken"    "syndrome"         "shaken_syndrome"
## t_5 "abuse"     "foster"    "sexual"           "physical"
## t_6 "full"      "arrest"    "full_arrest"      "er"
##     [,5]
## t_1 "abdomen"
## t_2 "grandmother"
## t_3 "friend"
## t_4 "homicide"
## t_5 "apparent"
## t_6 "residence"
```

```
model$prevalence <- colSums(model$theta) / sum(model$theta) * 100
plot(model$prevalence, model$alpha, xlab = "prevalence", ylab = "alpha")
```




```
model$labels <- LabelTopics(assignments = model$theta > 0.05,
                            dtm = dtm,
                            M = 1)

head(model$labels)
```

```
##     label_1
## t_1 "multiple_bruises"
## t_2 "adult_male"
## t_3 "domestic_violence"
## t_4 "shaken_syndrome"
## t_5 "cardiac_arrest"
## t_6 "full_arrest"
```

```
model$summary <- data.frame(topic = rownames(model$phi),
                            label = model$labels,
                            coherence = round(model$coherence, 3),
                            prevalence = round(model$prevalence,3),
                            top_terms = apply(model$top_terms, 2, function(x){
                              paste(x, collapse = ", ")
                            }),
                            stringsAsFactors = FALSE)
model$summary[ order(model$summary$coherence, decreasing = TRUE) , ]
```

```
##      topic             label_1 coherence prevalence
## t_14  t_14            gun_shot     0.694      7.253
## t_8    t_8   subdural_hematoma     0.455      4.378
## t_11  t_11     left_unattended     0.365      4.458
## t_10  t_10      emergency_room     0.322      6.537
## t_20  t_20         plastic_bag     0.298      7.305
## t_15  t_15   prior_examination     0.282      2.475
## t_6    t_6         full_arrest     0.234      4.329
## t_3    t_3   domestic_violence     0.232      2.806
## t_16  t_16    mechanism_injury     0.225      4.553
## t_4    t_4     shaken_syndrome     0.203      4.024
## t_1    t_1    multiple_bruises     0.151      5.207
## t_12  t_12     weeks_gestation     0.145      6.221
## t_9    t_9            gun_shot     0.119      3.459
## t_17  t_17 retinal_hemorrhages     0.100      8.468
## t_5    t_5      cardiac_arrest     0.085      3.881
## t_7    t_7       known_medical     0.085      7.581
## t_2    t_2          adult_male     0.075      4.577
## t_18  t_18           v_station     0.050      1.695
## t_19  t_19      medical_center     0.050      6.430
## t_13  t_13         blunt_force     0.011      4.361
##                                                    top_terms
## t_14                  shot, wound, gun, gun_shot, shot_wound
## t_8        hematoma, brain, subdural, subdural_hematoma, care
## t_11               bathtub, left, water, drowning, unattended
## t_10          room, emergency_room, emergency, death, arrival
## t_20                    bag, trash, plastic, newborn, infant
## t_15               scene, department, work, determined, death
## t_6                  full, arrest, full_arrest, er, residence
## t_3  domestic, delivered, domestic_violence, violence, friend
## t_16     unknown, infant, injury, mechanism, mechanism_injury
## t_4       father, shaken, syndrome, shaken_syndrome, homicide
## t_1                boyfriend, bruises, toddler, body, abdomen
## t_12               fetus, weeks, delivered, multiple, section
## t_9                         father, vehicle, left, car, seat
## t_17             head, father, injury, hemorrhages, fractures
## t_5                 abuse, foster, sexual, physical, apparent
## t_7               infant, father, history, unresponsive, bed
## t_2               home, fire, residence, family, grandmother
## t_18                 father, residence, family, shot, sister
## t_19                 unresponsive, medical, male, dr, prior
## t_13                  head, multiple, bruising, aunt, prior
```

```
########################
names(model)
```

```
##  [1] "phi"            "theta"          "gamma"          "data"
##  [5] "alpha"          "beta"           "log_likelihood" "coherence"
##  [9] "r2"             "top_terms"      "prevalence"     "labels"
## [13] "summary"
```

```
summary(model$coherence)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.01106 0.08512 0.17698 0.20903 0.28571 0.69410
```

```
model$assignments <- model$theta
model$assignments[ model$assignments < 0.05 ] <- 0
model$assignments <- model$assignments / rowSums(model$assignments)
model$assignments[ is.na(model$assignments) ] <- 0

# Get some topic labels using n-grams from the DTM
model$labels <- LabelTopics(assignments = model$assignments, dtm = dtm, M = 2)

model$doc_count <- colSums(model$assignments > 0)

# Create a summary matrix to view topics
model$topic_summary <- data.frame(topic = rownames(model$phi),
top_terms = apply(model$top_terms, 2,
function(x) paste(x, collapse=", ")),labels = apply(model$labels, 1, function(x) paste(x, collap
se=", ")),
coherence = round(model$coherence, 3),prevalence = round(model$prevalence),
doc_count = model$doc_count,
stringsAsFactors=FALSE)

print(model$topic_summary)
```

```
##      topic                                                top_terms
## t_1    t_1               boyfriend, bruises, toddler, body, abdomen
## t_2    t_2               home, fire, residence, family, grandmother
## t_3    t_3 domestic, delivered, domestic_violence, violence, friend
## t_4    t_4      father, shaken, syndrome, shaken_syndrome, homicide
## t_5    t_5                abuse, foster, sexual, physical, apparent
## t_6    t_6                 full, arrest, full_arrest, er, residence
## t_7    t_7               infant, father, history, unresponsive, bed
## t_8    t_8       hematoma, brain, subdural, subdural_hematoma, care
## t_9    t_9                          father, vehicle, left, car, seat
## t_10  t_10          room, emergency_room, emergency, death, arrival
## t_11  t_11               bathtub, left, water, drowning, unattended
## t_12  t_12               fetus, weeks, delivered, multiple, section
## t_13  t_13                    head, multiple, bruising, aunt, prior
## t_14  t_14                   shot, wound, gun, gun_shot, shot_wound
## t_15  t_15               scene, department, work, determined, death
## t_16  t_16     unknown, infant, injury, mechanism, mechanism_injury
## t_17  t_17             head, father, injury, hemorrhages, fractures
## t_18  t_18                  father, residence, family, shot, sister
## t_19  t_19                  unresponsive, medical, male, dr, prior
## t_20  t_20                    bag, trash, plastic, newborn, infant
##                                          labels coherence prevalence
## t_1           boyfriend_care, fall_boyfriend_care     0.151          5
## t_2                     th_floor, washing_machine     0.075          5
## t_3                brought_er, brought_er_friend     0.232          3
## t_4                  shaken_syndrome, due_shaken     0.203          4
## t_5  apparent_hyperthermia, locked_vehicle_degree     0.085          4
## t_6               full_arrest, approx_full_arrest     0.234          4
## t_7          known_medical, known_medical_history     0.085          8
## t_8        subdural_hematoma, retinal_hemorrhages     0.455          4
## t_9         multiple_witnesses, traffic_collision     0.119          3
## t_10          marks_chest, arrival_emergency_room     0.322          7
## t_11          left_unattended, unattended_bathtub     0.365          4
## t_12             fetal_demise, intrauterine_fetal     0.145          6
## t_13         multiple_blunt_force, multiple_blunt     0.011          4
## t_14                         gun_shot, shot_wound     0.694          7
## t_15           fire_department, prior_examination     0.282          2
## t_16              nursing_home, home_unresponsive     0.225          5
## t_17  retinal_hemorrhages, resuscitated_emergency     0.100          8
## t_18                        v_station, shot_death     0.050          2
## t_19                      foul_play, sq_responded     0.050          6
## t_20                    trash_dumpster, full_term     0.298          7
##      doc_count
## t_1        100
## t_2         66
## t_3         41
## t_4         65
## t_5         51
## t_6         76
## t_7        125
## t_8         69
## t_9         59
## t_10       115
```

```
## t_11        52
## t_12        58
## t_13        75
## t_14        62
## t_15        25
## t_16        75
## t_17       113
## t_18        14
## t_19       108
## t_20        63
```

```
tf_mat <- TermDocFreq(dtm = dtm)
head(tf_mat[ order(tf_mat$term_freq, decreasing = TRUE) , ], 10)
```

```
##                       term term_freq doc_freq       idf
## father              father       274      121 1.2841426
## infant              infant       224      121 1.2841426
## head                  head       214      141 1.1311733
## death                death       213      169 0.9500345
## unresponsive unresponsive       173      135 1.1746584
## left                  left       158      107 1.4071044
## home                  home       134      101 1.4648127
## medical            medical       133      110 1.3794528
## shot                  shot       125       59 2.0023958
## multiple          multiple       123       80 1.6979066
```

```
tf_bigrams <- tf_mat[ stringr::str_detect(tf_mat$term, "_") , ]
head(tf_bigrams[ order(tf_bigrams$term_freq, decreasing = TRUE) , ], 10)
```

```
##                                    term term_freq doc_freq      idf
## emergency_room           emergency_room        70       65 1.905546
## retinal_hemorrhages retinal_hemorrhages        55       50 2.167910
## gun_shot                       gun_shot        52       41 2.366361
## shot_wound                   shot_wound        48       38 2.442347
## gun_shot_wound           gun_shot_wound        45       36 2.496414
## full_arrest                 full_arrest        43       38 2.442347
## cardiac_arrest           cardiac_arrest        37       35 2.524585
## known_medical             known_medical        36       33 2.583426
## subdural_hematoma     subdural_hematoma        36       28 2.747729
## full_term                     full_term        36       30 2.678736
```

```
dtm <- dtm[ , colSums(dtm > 0) > 3 ] # alternatively: dtm[ , tf_mat$term_freq > 3 ]

tf_mat <- tf_mat[ tf_mat$term %in% colnames(dtm) , ]

tf_bigrams <- tf_bigrams[ tf_bigrams$term %in% colnames(dtm) , ]
m1<-as.matrix(dtm)
#write.csv(m1, file=paste("F:/Examples/ITMViz-master/ITMViz-master/data/jedit-5.1.0", "DocumentT
ermMatrix.csv", sep="/"))
```