Country Music Project

Matt Shu

2022-05-05

Prereqs

```
# Needed to overcome error found below with Homebrew TBB vs bundled TBB:
# https://github.com/RcppCore/RcppParallel/issues/182
# remotes::install_github("RcppCore/RcppParallel")
library(igraph)
library(tidyverse)
library(stm)
library(RSQLite)
library(RecordLinkage)
library(stringdist)
library(devtools)
library(tm)
# devtools::install_github("mikajoh/tidystm", dependencies = TRUE)
library(tidystm)
library(car)
library(xtable)
```

```
conn <- dbConnect(RSQLite::SQLite(), "files/22-04-21-playback-fm-top-country.db")
cleaned_df <- dbGetQuery(conn, 'SELECT * FROM cleaned')
dbDisconnect(conn)
```

```
names(cleaned_df)
```

```
##  [1] "artist_id"          "track_id"           "year"
##  [4] "artist"             "track"              "rank"
##  [7] "link"               "lyrics"             "artist_appearances"
## [10] "mb_id"              "type"               "area.name"
## [13] "gender"             "life_span.begin"    "life_span.ended"
## [16] "song_id"            "cleaned_lyrics"     "lyrics_alnum"
```

Preprocessing (and STM exploration)

```
cleaned_df <- cleaned_df %>%
  filter(gender != "non-binary") %>%
  as.data.frame()
```

```
docs_df <- cleaned_df %>%
   dplyr::select(track_id, lyrics_alnum) %>%
   filter(!is.na(lyrics_alnum))  %>%
   as.data.frame()
```

```
# Dataframe containing (sample) documents' metadata of interest
meta_df <- cleaned_df %>%
   dplyr::select(track_id, rank, artist, track, year, gender, artist_appearances) %>%
   # the objects need to be class "data frame"
   as.data.frame()
```

```
processed_docs_1 <- textProcessor(documents = docs_df$lyrics_alnum,
                                  metadata = meta_df,
                                  lowercase = TRUE,
                                  removestopwords = TRUE,
                                  removenumbers = TRUE,
                                  removepunctuation = TRUE,
                                  ucp = TRUE,
                                  stem = TRUE,
                                  striphtml = TRUE,
                                  wordLengths = c(3, Inf),
                                  language = "en")
```

```
meta <- processed_docs_1$meta
vocab <- processed_docs_1$vocab
docs <- processed_docs_1$documents
keep <- !is.na(meta$artist) & !is.na(meta$rank) & !is.na(meta$gender)
meta <- meta[keep,]
docs <- docs[keep]
```

```
prepped_data <- prepDocuments(docs,
                             vocab,
                             meta,
                             lower.thresh = 2)
```

Old code for removing unusual mismatch with no words despite past filters

```
length(docs_df$lyrics_alnum) # original documents
length(prepped_data$meta$track_id) # off from the preceding count
dif <- setdiff(docs_df$track_id, # original vector of documents
               prepped_data$meta$track_id) # list of documents after prepDocuments
tmp <- docs_df
tmp2 <- tmp[!tmp$track_id %in% dif,]
tmp_doc <- tmp2 %>%
  select(track_id, lyrics_alnum)
length(tmp_doc$track_id)
length(prepped_data$meta$track_id)

# View the track ids that were removed for some reason (often other language)
tmp3 <- tmp[tmp$track_id %in% dif,]
tmp3
```

See Cleaned Sample!

```
head(cleaned_df)
```

```
##   artist_id track_id year    artist                      track rank
## 1         1        0 1944 Red Foley          Smoke On The Water    1
## 2         1      506 1951 Red Foley                 Hobo Boogie   55
## 3         1      587 1953 Red Foley                    Midnight   14
## 4         1      386 1950 Red Foley     Cincinnati Dancing Pig   13
## 5         1      374 1950 Red Foley Chattanoogie Shoe Shine Boy    1
## 6         1      620 1953 Red Foley                   Hot Toddy   47
##                                                                link
## 1          /charts/country/video/1944/red-foley-smoke-on-the-water
## 2                 /charts/country/video/1951/red-foley-hobo-boogie
## 3                    /charts/country/video/1953/red-foley-midnight
## 4      /charts/country/video/1950/red-foley-cincinnati-dancing-pig
## 5 /charts/country/video/1950/red-foley-chattanoogie-shoe-shine-boy
## 6                   /charts/country/video/1953/red-foley-hot-toddy
##
## 1
## 2
## 3
## 4
## 5 Chattanoogie Shoe Shine Boy LyricsHave you ever passed the corner of Forth and Grand? Where a littl
## 6
##   artist_appearances                                mb_id   type     area.name
## 1                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
## 2                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
## 3                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
## 4                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
## 5                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
## 6                 33 aff932c2-ec30-4ee9-9125-5f761aae61a4 Person United States
##   gender life_span.begin life_span.ended song_id
## 1   male      1910-06-17            true   14519
## 2   male      1910-06-17            true   11892
## 3   male      1910-06-17            true   13445
## 4   male      1910-06-17            true   10833
## 5   male      1910-06-17            true   10810
## 6   male      1910-06-17            true   11966
##
## 1
## 2
## 3
## 4
## 5 Have you ever passed the corner of Forth and Grand? Where a little ball o' rhythm has a shoe-shine
## 6
##
## 1
## 2
## 3
## 4
## 5 Have you ever passed the corner of Forth and Grand  Where a little ball o  rhythm has a shoe shine
## 6
```

Find K

```
k_seq = seq(4, 15, 1)
```

```
## You can "watch" the algorithm model topics in the console
searched = searchK(prepped_data$documents,
                   prepped_data$vocab,
                   K = k_seq,
                   data = prepped_data$meta,
                   seed = 183654)
saveRDS(searched, file = "files/22-04-29-searchK.RData")
```

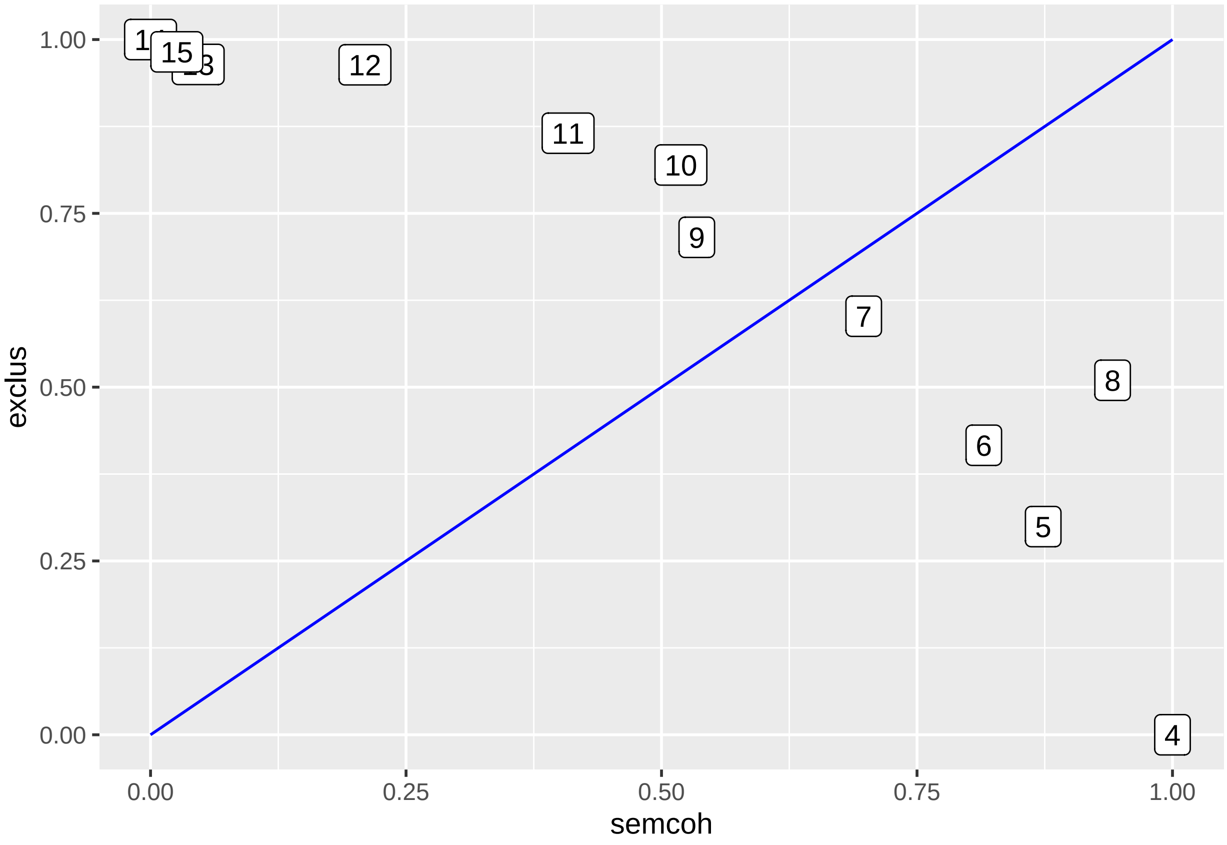
Show K

```
searched <- readRDS("files/22-04-29-searchK.RData")
# Get values from `searchK` output
semcoh <- unlist(searched$results$semcoh)
exclus <- unlist(searched$results$exclus)

# Max/min semantic cohesion
max_sc <- max(semcoh)
min_sc<-min(semcoh)

# Max/min exclusivity
max_ex<-max(exclus)
min_ex<-min(exclus)

# Min-max normalization is (value - min)/(max - min)
x_vals <- (semcoh-min_sc)/(max_sc-min_sc)
y_vals <- (exclus-min_ex)/(max_ex-min_ex)
# add semantic cohesion and exclusivity together weighted evenly
search_plot_df <- tibble(id = k_seq,
                   semcoh = x_vals,
                   exclus = y_vals,
                   combine = x_vals*0.5 + y_vals*0.5)
# Plot
ggplot(search_plot_df, mapping = aes(x = semcoh, y = exclus)) +
  xlim(0,1) +
  ylim(0,1) +
  ggplot2::annotate("segment", x = 0, xend = 1, y = 0, yend = 1, color = "blue") +
  geom_label(aes(label=id))
```



Model Work

```
num_topics <- 7 # Chosen after above search and some playing around
```

```
# 6 topics seems to also work nice, with a strong "Country" category
out_covariates_7 <- stm(prepped_data$documents,
                         prepped_data$vocab,
                         K = num_topics,
                         prevalence = ~ rank + year * gender,
                         max.em.its = 500,
                         data = prepped_data$meta,
                         seed = 592669)
```

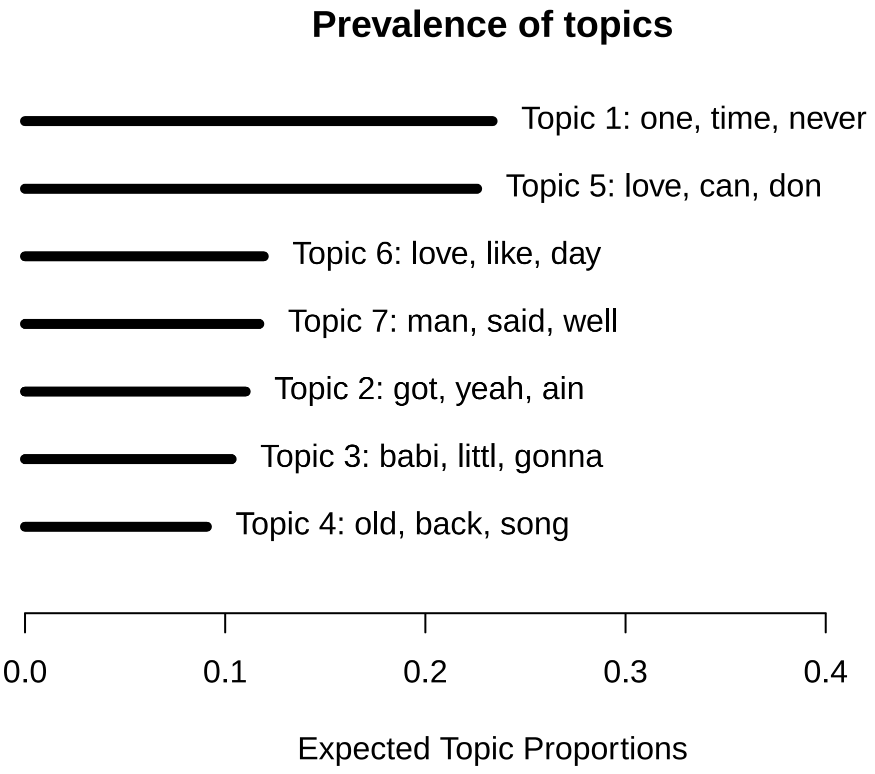
```
terms = labelTopics(out_covariates_7, n = 10)
terms$prob # rows are topics; columns are most probable words (in order)
```

```
##      [,1]   [,2]    [,3]    [,4]      [,5]    [,6]    [,7]   [,8]    [,9]
## [1,] "one"  "time"  "never" "now"     "heart" "still" "say"  "just"  "gone"
## [2,] "got"  "yeah"  "ain"   "like"    "girl"  "good"  "get"  "wanna" "just"
## [3,] "babi" "littl" "gonna" "come"    "night" "get"   "time" "take"  "back"
## [4,] "old"  "back"  "song"  "countri" "roll"  "town"  "road" "ride"  "like"
## [5,] "love" "can"   "don"   "know"    "just"  "want"  "let"  "make"  "feel"
## [6,] "love" "like"  "day"   "night"   "dream" "eye"   "blue" "sweet" "rain"
## [7,] "man"  "said"  "well"  "old"     "daddi" "boy"   "big"  "mama"  "just"
##      [,10]
## [1,] "think"
## [2,] "can"
## [3,] "home"
## [4,] "sing"
## [5,] "need"
## [6,] "light"
## [7,] "got"
```

```
terms$frex # rows are topics; columns are most FREX words (in order)
```

```
##      [,1]      [,2]     [,3]        [,4]    [,5]     [,6]     [,7]
## [1,] "fool"    "goodby" "cri"       "lone"  "miss"   "memori" "lie"
## [2,] "ooh"     "huh"    "boo"       "yeah"  "nothin" "ain"    "whoa"
## [3,] "bye"     "babi"   "bit"       "honey" "shake"  "gonna"  "danc"
## [4,] "countri" "boogi"  "hillbilli" "crank" "cowboy" "cha"    "doo"
## [5,] "hold"    "need"   "want"      "fall"  "love"   "pleas"  "easi"
## [6,] "heaven"  "rain"   "angel"     "sail"  "sea"    "storm"  "sunshin"
## [7,] "mom"     "dad"    "wife"      "hero"  "father" "twenti" "daddi"
##      [,8]       [,9]     [,10]
## [1,] "heartach" "still"  "tear"
## [2,] "lovin"    "gimm"   "nobodi"
## [3,] "step"     "littl"  "batter"
## [4,] "jone"     "tonk"   "santa"
## [5,] "believ"   "feel"   "lose"
## [6,] "rainbow"  "wing"   "sky"
## [7,] "sir"      "famili" "mommi"
```

```
# Parameters modified from: https://milesdwilliams15.github.io/Better-Graphics-for-the-stm-Package-in-R
par(bty="n",lwd=5)
plot(out_covariates_7,
     type = "summary",
     main = "Prevalence of topics")
```



```r
docs_examples_covar <- findThoughts(out_covariates_7,
                                    texts = tmp_doc$track_id,
                                    n = 10,
                                    topics = c(1:num_topics))

for(topic_num in c(1:num_topics)) {
  print(paste("Topic ", topic_num))
  for(track in docs_examples_covar$docs[[topic_num]]) {
    print(cleaned_df$track[cleaned_df$track_id == track])
  }
  print("")
}
```

```
## [1] "Topic  1"
## [1] "Something Old, Something New"
## [1] "All Alone in This World without You"
## [1] "I Forgot To Remember To Forget"
## [1] "Fool Fool Fool"
## [1] "Happy Journey"
## [1] "You're The One"
## [1] "Sweetheart You Done Me Wrong"
## [1] "Hang Your Head In Shame"
## [1] "Things Aren't Funny Anymore"
## [1] "Careless Darlin'"
## [1] ""
## [1] "Topic  2"
## [1] "Desperate Man"
## [1] "Gimmie That Girl"
## [1] "My Bucket's Got a Hole in it"
## [1] "Just The Way"
## [1] "Just the Way"
## [1] "Cool Again"
## [1] "Drinkin' Beer. Talkin' God. Amen."
## [1] "Uh-Huh--Mm"
## [1] "She Ain't Your Ordinary Girl"
## [1] "Uh-Huh-mm"
## [1] ""
## [1] "Topic  3"
## [1] "Swing"
## [1] "Waitin' in School"
## [1] "Waitin' In School"
## [1] "Baby Let's Play House"
## [1] "Trademark"
## [1] "Little Bit of Life"
## [1] "Little Bit Of Life"
## [1] "Penny Arcade"
## [1] "Shine, Shave, Shower (It's Saturday)"
## [1] "Whole Lotta Shakin' Goin' On"
## [1] ""
## [1] "Topic  4"
## [1] "Teenage Boogie"
## [1] "Redneck Yacht Club"
## [1] "Cincinnati Dancing Pig"
## [1] "Long Live"
## [1] "Ragtime Cowboy Joe"
## [1] "Mule Train"
## [1] "She Cranks My Tractor"
## [1] "Smokey Mountain Boogie"
## [1] "The Rhumba Boogie"
## [1] "Hula Rock"
## [1] ""
## [1] "Topic  5"
## [1] "Love Can't Wait"
## [1] "Don't Underestimate My Love For You"
## [1] "Don't Underestimate My Love for You"
## [1] "I Want To Know You Before We Make Love"
## [1] "Count on Me"
## [1] "A Lover's Question"
## [1] "Mr. Lovemaker"
## [1] "Fall into Me"
## [1] "Fall Into Me"
## [1] "It Matters to Me"
## [1] ""
## [1] "Topic  6"
## [1] "Ring Of Fire"
## [1] "My Special Angel"
## [1] "The Red Strokes"
## [1] "Your Name Is Beautiful"
## [1] "Sweet Summer Lovin'"
## [1] "Mockin' Bird Hill"
## [1] "A Fallen Star"
## [1] "Would You Lay With Me (In A Field Of Stone)"
## [1] "Kentucky Waltz"
## [1] "Beautiful Brown Eyes"
## [1] ""
## [1] "Topic  7"
## [1] "What's Your Mama's Name"
## [1] "Life Of A Poor Boy"
## [1] "No Charge"
## [1] "(Margie's At) The Lincoln Park Inn"
## [1] "Poor, Poor Pitiful Me"
## [1] "History Repeats Itself"
## [1] "Deck Of Cards"
## [1] "Po' Folks"
## [1] "Shiftwork"
## [1] "None Of My Business"
## [1] ""
```

```
# Topic 1: Heartbreak Songs
# Topic 2: Cross-Country (Country Rock/Pop)
# Topic 3: Traditionalist Country (Pardi, Hank Williams)
# Topic 4: Bro-Country
# Topic 5: Sex Jams
# Topic 6: Love songs
# Topic 7: Family
topic_labels <- c("Heartbreak", "Cross-Country", "(Neo)-Traditional", "Bro-Country", "Sex Jams", "Romanc
```

```
num_topics <- 7
length(prepped_data$meta$year)
```

```
## [1] 5969
```

```
length(prepped_data$meta$gender)
```

```
## [1] 5969
```

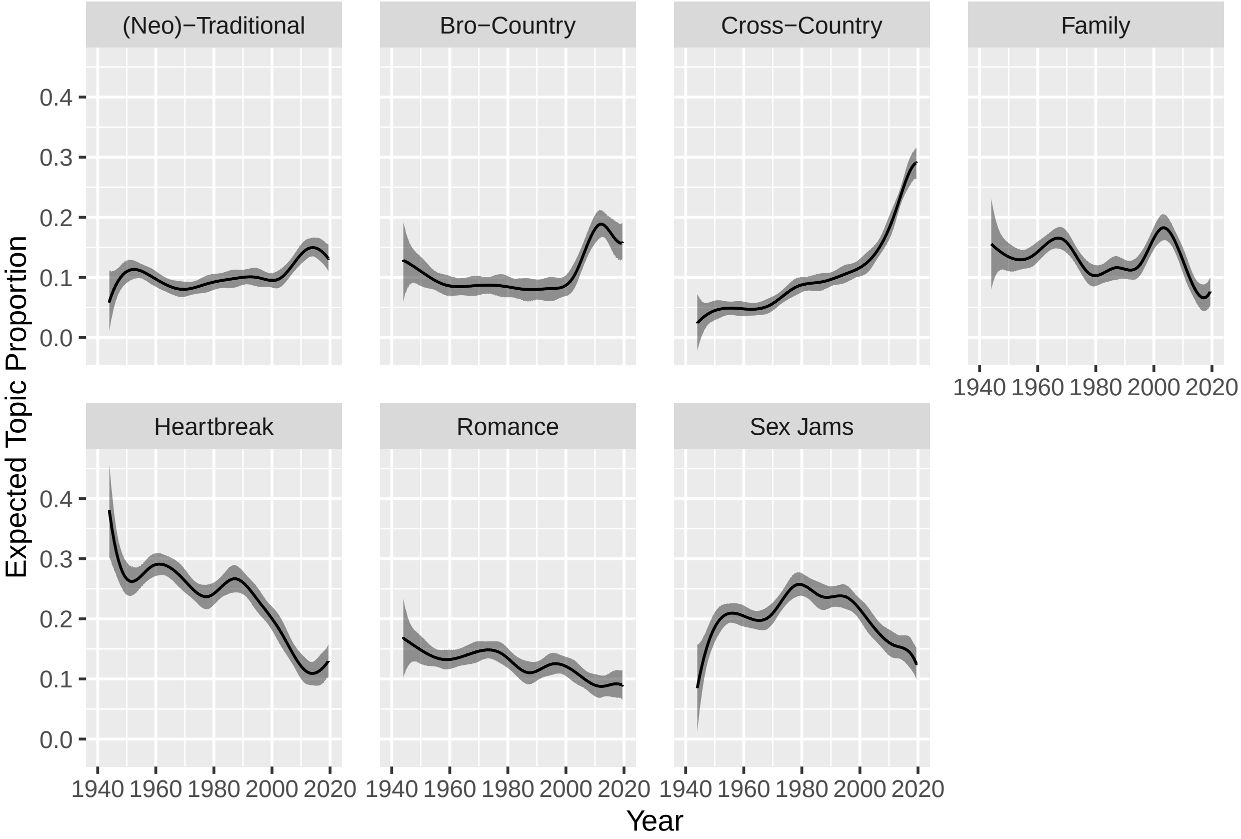
```
eff1 <- estimateEffect(formula = c(1:7) ~ s(year) * gender,
                      # the line above matches the model specification we used
                      stmobj = out_covariates_7,
                      meta = prepped_data$meta,
                      uncertainty = "Global")

# plot.estimateEffect(eff1,
#       covariate = "year",
#       topics = c(1:num_topics),
#       model = out_covariates_7,
#       method = "continuous",
#       xlab = "Year",
#       ylim=c(0, .4),
#       xlim=c(1940, 2020),
#       main = "Effect of Year on Topic Proportion")
```

```
effect <- lapply(c(0, 1), function(i) {
  extract.estimateEffect(eff1,
     covariate = "year",
     topics = c(1:num_topics),
     model = out_covariates_7,
     method = "continuous")
})
effect <- do.call("rbind", effect)
effect <- effect %>% mutate(label = dplyr::recode(topic, "1"=topic_labels[1], "2" = topic_labels[2], "3
## And, for example, plot it with ggplot2 and facet by topic instead.
library(ggplot2)

ggplot(effect, aes(x = covariate.value, y = estimate,
                   ymin = ci.lower, ymax = ci.upper)) +
  facet_wrap(~ label, nrow = 2) +
  geom_ribbon(alpha = .5) +
  geom_line() +
  labs(x = "Year",
       y = "Expected Topic Proportion") +
  scale_x_continuous(breaks=c(1940, 1960, 1980, 2000, 2020),
   labels=waiver(), lim=c(1940,2020)) +
  theme(panel.spacing = unit(1, "lines"))
```

```
## Warning: Removed 4 row(s) containing missing values (geom_path).
```

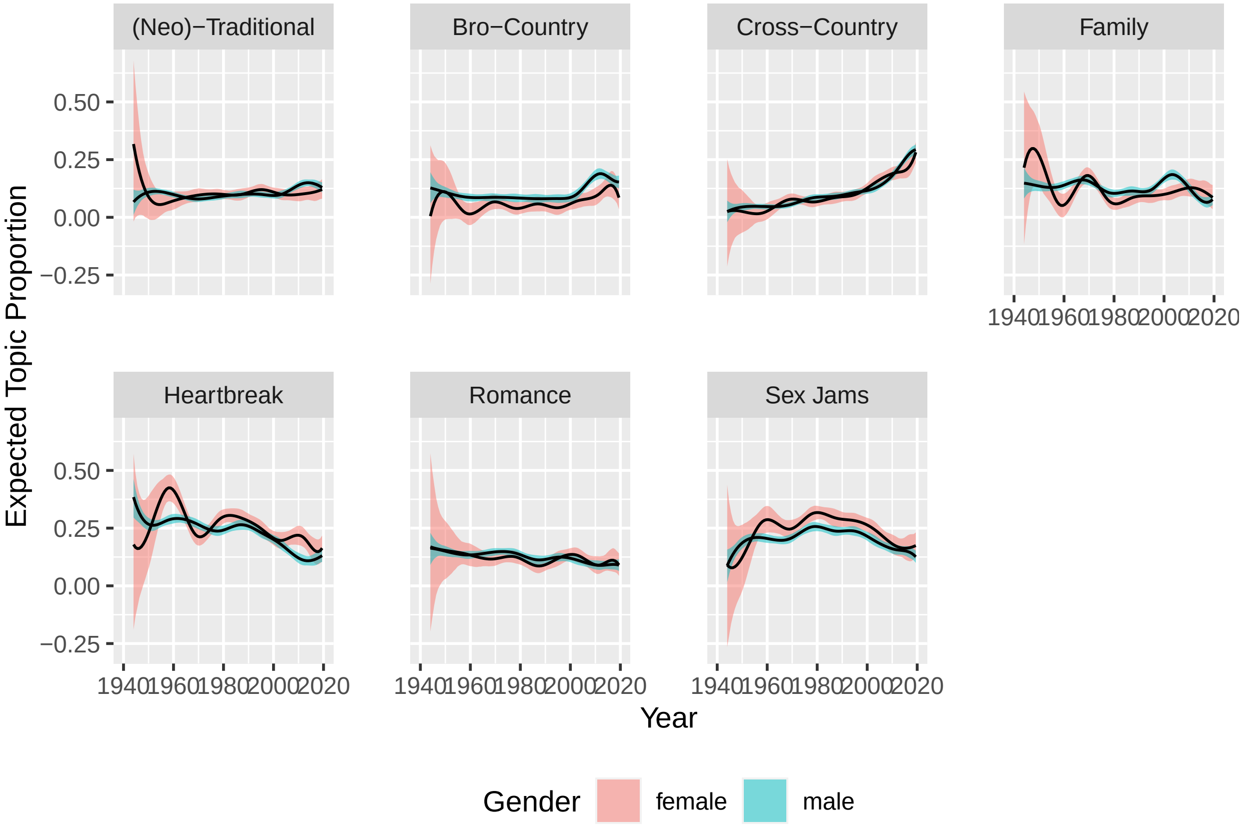


```
# pdf(file = "figures/gender-subgenre-time.pdf", width = 10)
eff <- estimateEffect(formula = c(1:7) ~ s(year) * gender,
                      # the line above matches the model specification we used
                      stmobj = out_covariates_7,
                      meta = prepped_data$meta,
                      uncertainty = "Global")

effect <- lapply(c("male", "female"), function(i) {
  extract.estimateEffect(x = eff,
     covariate = "year",
     topics = c(1:num_topics),
     model = out_covariates_7,
     method = "continuous",
     moderator = "gender",
     moderator.value = i)
})
effect <- do.call("rbind", effect)
effect <- effect %>% mutate(label = dplyr::recode(topic, "1"=topic_labels[1], "2" = topic_labels[2], "3

ggplot(effect, aes(x = covariate.value, y = estimate,
                   ymin = ci.lower, ymax = ci.upper,
                   group = moderator.value,
                   fill = factor(moderator.value))) +
  facet_wrap(~ label, nrow = 2) +
  geom_ribbon(alpha = .5) +
  geom_line() +
  labs(x = "Year",
      y = "Expected Topic Proportion") +
  scale_x_continuous(breaks=c(1940, 1960, 1980, 2000, 2020),
   labels=waiver(), lim=c(1940,2020)) +
  theme(panel.spacing = unit(2, "lines"), legend.direction="horizontal",legend.position="bottom", legen
  labs(fill = "Gender")
```

```
## Warning: Removed 4 row(s) containing missing values (geom_path).
```



```
# dev.off()
```

```
library(huge)
```

```
## Registered S3 methods overwritten by 'huge':
##   method    from
##   plot.sim  lava
##   print.sim lava
```

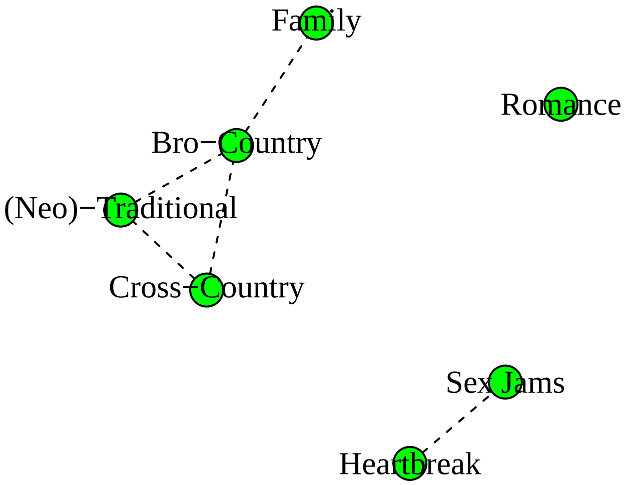
```
topic_corr <- topicCorr(out_covariates_7, method = "huge")
```

```
## Conducting the nonparanormal (npn) transformation via shrunkun ECDF....done.
## Conducting Meinshausen & Buhlmann graph estimation (mb)....done
## Conducting rotation information criterion (ric) selection....done
## Computing the optimal graph....done
```

```
topic_corr
```

```
## $posadj
## 7 x 7 sparse Matrix of class "dgCMatrix"
##
## [1,] . . . . . 1 . .
## [2,] . . 1 1 . . .
## [3,] . 1 . 1 . . .
## [4,] . 1 1 . . . 1
## [5,] 1 . . . . . .
## [6,] . . . . . . .
## [7,] . . . 1 . . .
##
## $poscor
## 7 x 7 sparse Matrix of class "dgCMatrix"
##
## [1,] .          .          .           .           0.02844859 . .
## [2,] .          .          0.140686252 0.052294962 .          . .
## [3,] .          0.14068625 .           0.008587188 .          . .
## [4,] .          0.05229496 0.008587188 .           .          . 0.07118912
## [5,] 0.02844859 .          .           .           .          . .
## [6,] .          .          .           .           .          . .
## [7,] .          .          .           0.071189118 .          . .
##
## $cor
## 7 x 7 Matrix of class "dgeMatrix"
##             [,1]        [,2]         [,3]         [,4]        [,5]       [,6]
## [1,]  0.00000000 -0.33517905 -0.287172416 -0.386253589  0.02844859 -0.1489539
## [2,] -0.33517905  0.00000000  0.140686252  0.052294962  0.00000000 -0.2376631
## [3,] -0.28717242  0.14068625  0.000000000  0.008587188  0.00000000 -0.2051478
## [4,] -0.38625359  0.05229496  0.008587188  0.000000000 -0.41592488 -0.1152605
## [5,]  0.02844859  0.00000000  0.000000000 -0.415924876  0.00000000  0.0000000
## [6,] -0.14895388 -0.23766309 -0.205147781 -0.115260498  0.00000000  0.0000000
## [7,]  0.00000000  0.00000000 -0.128285907  0.071189118 -0.35160217 -0.1581811
##             [,7]
## [1,]  0.00000000
## [2,]  0.00000000
## [3,] -0.12828591
## [4,]  0.07118912
## [5,] -0.35160217
## [6,] -0.15818113
## [7,]  0.00000000
##
## attr(,"class")
## [1] "topicCorr"
```

```
set.seed(5)
plot(topic_corr,
  vlabels = topic_labels, vertex.label.cex = 1, layout =  layout.auto)
```



Topics 3, 2, 4, 7 are all related. This is an interesting finding! This suggests that traditionalist country especially seems related to both country rock/pop songs Topic 2?: Country Rock/Pop Topic 3: Traditionalist Country Topic 4: Bro-Country Topic 7: Family

More on Topic Models

Questions/Interests

- How would I see where individual artists fell in terms of topics?
- In general, seeing prevalence of certain
- Would it be, taking the top x documents for different topics and counting from there? ### More to Do?
- Plot covariate interaction!
  - Particularly interested in tracking gender * year interactions!

Artist Validation

```
head(out_covariates_7$theta) # each row is each document
```

```
##             [,1]        [,2]        [,3]       [,4]        [,5]        [,6]
## [1,] 0.17143003 0.007183287 0.016922249 0.14898950 0.033074431 0.365870054
## [2,] 0.06056009 0.026463971 0.028309853 0.64724253 0.171068445 0.039770943
## [3,] 0.47745747 0.034807344 0.041475635 0.01423934 0.084898921 0.308433502
## [4,] 0.00661786 0.006916791 0.008254175 0.91129141 0.004351277 0.031036096
## [5,] 0.03787950 0.033285558 0.108736954 0.38615651 0.031878444 0.201169853
## [6,] 0.03504111 0.649147616 0.029693331 0.01517530 0.035813788 0.009704519
##             [,7]
## [1,] 0.25653045
## [2,] 0.02658416
## [3,] 0.03868779
## [4,] 0.03153239
## [5,] 0.20089318
## [6,] 0.22542433
```

```
# To find each artists, link the songs to the artists and then take the average for each artists, for e
head(prepped_data$meta) # same order between dataframes
```

```
##   track_id rank    artist                    track year gender
## 1        0    1 Red Foley      Smoke On The Water 1944   male
## 2      506   55 Red Foley             Hobo Boogie 1951   male
## 3      587   14 Red Foley                Midnight 1953   male
## 4      386   13 Red Foley      Cincinnati Dancing Pig 1950   male
## 5      374    1 Red Foley Chattanoogie Shoe Shine Boy 1950   male
## 6      620   47 Red Foley                   Hot Toddy 1953   male
##   artist_appearances
## 1                 33
## 2                 33
## 3                 33
## 4                 33
## 5                 33
## 6                 33
```

```
track_topic_df <- cbind(prepped_data$meta, out_covariates_7$theta)
artist_topic_df <- track_topic_df %>%
  filter(artist_appearances > 1) %>%
  group_by(artist) %>%
  summarize(mean_1=mean(`1`),mean_2=mean(`2`), mean_3=mean(`3`), mean_4=mean(`4`), mean_5=mean(`5`), mea
colnames(artist_topic_df)[2:(1+num_topics)] <- topic_labels
artist_topic_df
```

```
## # A tibble: 584 x 8
##    artist   Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##    <chr>         <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
##  1 Aaron L~     0.0932         0.251             0.0365       0.00825      0.430
##  2 Aaron T~     0.179          0.123             0.0638       0.102        0.213
##  3 Al Dext~     0.269          0.00822           0.143        0.00929      0.0972
##  4 Alabama      0.160          0.0719            0.0875       0.139        0.307
##  5 Alan Ja~     0.193          0.110             0.0780       0.121        0.205
##  6 Andy Gr~     0.259          0.0756            0.118        0.0429       0.244
##  7 Anne Mu~     0.209          0.0910            0.0889       0.0753       0.285
##  8 Ashley ~     0.131          0.417             0.156        0.123        0.112
##  9 Ashton ~     0.208          0.309             0.0981       0.183        0.112
## 10 Autry I~     0.337          0.0114            0.0383       0.0128       0.165
## # ... with 574 more rows, and 2 more variables: Romance <dbl>, Family <dbl>
```

```
for(topic in topic_labels) {
  print(artist_topic_df %>% arrange(desc(.data[[topic]])) %>% slice(1:5))
}
```

```
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Bill Mon~      0.843         0.00494          0.0179        0.00768      0.0736
## 2 Don McLe~      0.748         0.00949          0.0121        0.00140      0.201
## 3 Bill Phi~      0.706         0.00946          0.0183        0.00174      0.228
## 4 George J~      0.633         0.0109           0.0191        0.0212       0.250
## 5 Buck Owe~      0.628         0.00726          0.00977       0.00204      0.270
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Parmalee~     0.0285           0.776           0.0197       0.00351      0.152
## 2 Mitchell~     0.138            0.667           0.0881       0.00484      0.0868
## 3 Lady A        0.0329           0.611           0.111        0.0240       0.201
## 4 Walker H~     0.0544           0.589           0.120        0.148        0.0509
## 5 Ryan Hurd     0.0168           0.523           0.239        0.00822      0.197
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Chase Br~     0.102           0.146             0.461       0.0978      0.123
## 2 Ricky Ne~     0.209           0.186             0.392       0.0317      0.102
## 3 Foster &~     0.0961          0.0986            0.378       0.0160      0.364
## 4 Lari Whi~     0.251           0.147             0.362       0.00552     0.213
## 5 Dierks B~     0.118           0.0525            0.345       0.0422      0.0870
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 The Lost~    0.0123           0.144            0.0667         0.718     0.0166
## 2 Jack Gut~    0.0841           0.0152           0.107          0.604     0.0752
## 3 The Jane~    0.00925          0.301            0.0642         0.537     0.0322
## 4 Delmore ~    0.196            0.0125           0.0377         0.475     0.144
## 5 Morgan W~    0.0395           0.147            0.0936         0.473     0.122
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Boy Howdy     0.133          0.0222            0.0535      0.00188       0.765
## 2 Jimmy Wo~     0.184          0.00577           0.0172      0.000846      0.762
## 3 Zeb Turn~     0.0189         0.00686           0.0205      0.200         0.724
## 4 Bobby G.~     0.107          0.00933           0.0260      0.00339       0.673
## 5 Lila McC~     0.184          0.0755            0.0837      0.00377       0.602
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Margie S~     0.0880         0.00453           0.0140       0.00242     0.229
## 2 Steven T~     0.108          0.0232            0.0282       0.00987     0.179
## 3 Pee Wee ~     0.293          0.0136            0.0333       0.00572     0.0983
## 4 The Brow~     0.170          0.0112            0.0804       0.0350      0.138
## 5 Bobbie G~     0.104          0.00615           0.0146       0.00269     0.368
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
## # A tibble: 5 x 8
##   artist    Heartbreak `Cross-Country` `(Neo)-Traditi~` `Bro-Country` `Sex Jams`
##   <chr>          <dbl>           <dbl>            <dbl>         <dbl>      <dbl>
## 1 Henson C~     0.0896         0.0588            0.0212        0.0157     0.0234
## 2 Mac Wise~     0.0378         0.00967           0.0152        0.0175     0.0415
## 3 Claude G~     0.0835         0.0118            0.0147        0.142      0.0471
## 4 Ferlin H~     0.103          0.0754            0.139         0.0109     0.0434
## 5 Jamey Jo~     0.0261         0.114             0.123         0.0548     0.0669
## # ... with 2 more variables: Romance <dbl>, Family <dbl>
```

Female Artist Popularity + Subgenres

```
gender_year_df <- track_topic_df %>%
  filter(gender != "non-binary") %>%
  filter(gender != "group") %>%
  mutate(year_factor = factor(year), gender = factor(gender)) %>%
  group_by(year_factor, gender) %>%
  filter(n() > 2) %>%
  summarize(mean_1=mean(`1`),mean_2=mean(`2`), mean_3=mean(`3`), mean_4=mean(`4`), mean_5=mean(`5`), mea
  # summarize(gender_total=n(), sum_TTR=ifelse(gender_total != 0, sum(TTR)/gender_total, 0))  %>%
  mutate(year=as.numeric(as.character(year_factor)))
```

```
## `summarise()` has grouped output by 'year_factor'. You can override using the
## `.groups` argument.
```

```
lm(unlist(gender_year_df[,paste0("mean_", 1)]) ~ gender_year_df$gender)
```

```
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", 1)]) ~ gender_year_df$gender)
##
## Coefficients:
##                  (Intercept)     gender_year_df$gendermale
##                      0.26416                      -0.02482
## gender_year_df$genderunknown
##                     -0.04036
```

```
for(topic in c(1:7)) {
  print(paste("Topic:", topic_labels[topic]))
  anc1 <- lm(unlist(gender_year_df[,paste0("mean_", topic)]) ~ gender + year + gender*year, data = gende
  Anova(anc1, type = 3)
  print(summary(anc1))
}
```

```
## [1] "Topic: Heartbreak"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15549 -0.03514 -0.00152  0.03155  0.49111
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.469e+00  8.249e-01   6.630 2.65e-10 ***
## gendermale           1.028e-01  1.056e+00   0.097    0.923
## genderunknown       -6.278e-01  1.064e+00  -0.590    0.556
## year                -2.618e-03  4.149e-04  -6.310 1.56e-09 ***
## gendermale:year     -7.152e-05  5.317e-04  -0.135    0.893
## genderunknown:year  2.898e-04  5.358e-04   0.541    0.589
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06611 on 216 degrees of freedom
## Multiple R-squared:  0.4344, Adjusted R-squared:  0.4214
## F-statistic: 33.19 on 5 and 216 DF,  p-value: < 2.2e-16
##
## [1] "Topic: Cross-Country"
##
```

```
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.095869 -0.029255 -0.006975  0.023159  0.184508
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -6.8005218  0.5334600 -12.748   <2e-16 ***
## gendermale          1.1126438  0.6828451   1.629    0.105
## genderunknown       1.1119278  0.6881455   1.616    0.108
## year                0.0034766  0.0002683  12.956   <2e-16 ***
## gendermale:year    -0.0005549  0.0003438  -1.614    0.108
## genderunknown:year -0.0005543  0.0003465  -1.600    0.111
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04275 on 216 degrees of freedom
## Multiple R-squared:  0.7109, Adjusted R-squared:  0.7042
## F-statistic: 106.2 on 5 and 216 DF,  p-value: < 2.2e-16
##
## [1] "Topic: (Neo)-Traditional"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.076522 -0.020390 -0.004366  0.016186  0.132044
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -1.009e+00  3.996e-01  -2.524   0.0123 *
## gendermale         -5.643e-02  5.115e-01  -0.110   0.9122
## genderunknown      -6.214e-01  5.155e-01  -1.206   0.2293
## year                5.556e-04  2.010e-04   2.764   0.0062 **
## gendermale:year     3.269e-05  2.576e-04   0.127   0.8991
## genderunknown:year  3.180e-04  2.595e-04   1.225   0.2218
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03202 on 216 degrees of freedom
## Multiple R-squared:  0.1907, Adjusted R-squared:  0.172
## F-statistic: 10.18 on 5 and 216 DF,  p-value: 9.02e-09
##
## [1] "Topic: Bro-Country"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.09574 -0.03042 -0.00630  0.02262  0.32938
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -1.5097376  0.5844140  -2.583  0.01044 *
## gendermale          0.1297916  0.7480677   0.174  0.86242
## genderunknown       1.2462214  0.7538745   1.653  0.09977 .
## year                0.0007876  0.0002940   2.679  0.00795 **
## gendermale:year    -0.0000409  0.0003767  -0.109  0.91364
## genderunknown:year -0.0006046  0.0003796  -1.593  0.11264
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04683 on 216 degrees of freedom
## Multiple R-squared:  0.2142, Adjusted R-squared:  0.1961
## F-statistic: 11.78 on 5 and 216 DF,  p-value: 4.396e-10
##
## [1] "Topic: Sex Jams"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.165301 -0.048377 -0.002077  0.044097  0.315556
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         3.2831886  0.8137543   4.035 7.59e-05 ***
## gendermale         -2.6267311  1.0416304  -2.522 0.012399 *
## genderunknown      -3.1787061  1.0497159  -3.028 0.002760 **
## year               -0.0015196  0.0004093  -3.712 0.000261 ***
## gendermale:year     0.0012927  0.0005245   2.465 0.014496 *
## genderunknown:year  0.0015847  0.0005285   2.998 0.003033 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06521 on 216 degrees of freedom
## Multiple R-squared:  0.1582, Adjusted R-squared:  0.1387
## F-statistic: 8.116 on 5 and 216 DF,  p-value: 4.882e-07
##
## [1] "Topic: Romance"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.148130 -0.022859 -0.004323  0.020735  0.225292
##
```

```
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.1328304  0.5862953   1.932   0.0546 .
## gendermale          0.7463771  0.7504759   0.995   0.3211
## genderunknown       1.7776904  0.7563013   2.351   0.0196 *
## year               -0.0005142  0.0002949  -1.743   0.0827 .
## gendermale:year    -0.0003720  0.0003779  -0.984   0.3260
## genderunknown:year -0.0008871  0.0003808  -2.329   0.0208 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04699 on 216 degrees of freedom
## Multiple R-squared:  0.2123, Adjusted R-squared:  0.1941
## F-statistic: 11.64 on 5 and 216 DF,  p-value: 5.668e-10
##
## [1] "Topic: Family"
##
## Call:
## lm(formula = unlist(gender_year_df[, paste0("mean_", topic)]) ~
##     gender + year + gender * year, data = gender_year_df)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.097191 -0.033957 -0.006606  0.030803  0.152602
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.4340772  0.5775876   0.752    0.453
## gendermale          0.5915915  0.7393298   0.800    0.424
## genderunknown       0.2921510  0.7450687   0.392    0.695
## year               -0.0001678  0.0002905  -0.578    0.564
## gendermale:year    -0.0002860  0.0003723  -0.768    0.443
## genderunknown:year -0.0001465  0.0003751  -0.390    0.697
##
## Residual standard error: 0.04629 on 216 degrees of freedom
## Multiple R-squared:  0.08344,    Adjusted R-squared:  0.06222
## F-statistic: 3.933 on 5 and 216 DF,  p-value: 0.001964
```

```
selected_tracks <- c("All Alone in This World without You", "Coat of Many Colors", "Ring Of Fire", "Mama

colnames(track_topic_df)[8:(7+num_topics)] <- topic_labels
selected_track_topics <- track_topic_df %>% filter(track %in% selected_tracks) %>% mutate(across(8:(7+nu
print(xtable(selected_track_topics, type = "latex"), file = "figures/track_topics.tex")
```