Arya Ayati Project 1 R Notebook

This is the accompanying Notebook for my project 1 results/code. My project seeked out whether the sentiment of a school changed over time. My hypothesis was that for a given school, the sentiment should not change between their publications over time. The underlying idea was that a school has a set of thoughts that they follow, and that they wouldn't change that drastically over time since any large shifts would probably sprout a new school of thought.

Step 0 - Initialize the environment

```
packages.used=c("dplyr", "tidyverse", "tm", "wordcloud", "RColorBrewer",
                "tidytext", "Rcpp", "textclean", "ggalt", "ggplot2", "gridExtra")
# check packages that need to be installed.
packages.needed=setdiff(packages.used,
                        intersect(installed.packages()[,1],
                                  packages.used))
# install additional packages
if(length(packages.needed)>0){
  install.packages(packages.needed, dependencies = TRUE,
                   repos='http://cran.us.r-project.org')
}

library(dplyr)
```

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

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

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

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v stringr 1.4.0
## v tidyr   1.1.4     v forcats 0.5.1
## v readr   2.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tm)
```

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

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

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

```
library(wordcloud)
```

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

```
library(RColorBrewer)
library(tidytext)
library(Rcpp)
library(textclean)
library(ggalt)
```

```
## Registered S3 methods overwritten by 'ggalt':
##   method                  from
##   grid.draw.absoluteGrob  ggplot2
##   grobHeight.absoluteGrob ggplot2
##   grobWidth.absoluteGrob  ggplot2
##   grobX.absoluteGrob      ggplot2
##   grobY.absoluteGrob      ggplot2
```

```
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
setwd("M:/Documents/CU Coursework/STAT5243 Applied Data Science/Projects/Fall2021-Project1-AryaAyati")
```

The notebook was prepared with the following environmental settings.

```
print(R.version)
```

```
##                _
## platform       x86_64-w64-mingw32
## arch           x86_64
## os             mingw32
## system         x86_64, mingw32
## status
## major          4
## minor          1.1
## year           2021
## month          08
## day            10
## svn rev        80725
## language       R
## version.string R version 4.1.1 (2021-08-10)
## nickname       Kick Things
```

Step 1 - Read in the philosophy data and check the formatting

```
data.raw = read.csv('data/philosophy_data.csv')
```

```
colnames(data.raw)
```

```
##  [1] "title"                    "author"
##  [3] "school"                   "sentence_spacy"
##  [5] "sentence_str"             "original_publication_date"
##  [7] "corpus_edition_date"      "sentence_length"
##  [9] "sentence_lowered"         "tokenized_txt"
## [11] "lemmatized_str"
```

```
unique(data.raw$author)
```

```
##  [1] "Plato"           "Aristotle"       "Locke"           "Hume"
##  [5] "Berkeley"        "Spinoza"         "Leibniz"         "Descartes"
##  [9] "Malebranche"     "Russell"         "Moore"           "Wittgenstein"
## [13] "Lewis"           "Quine"           "Popper"          "Kripke"
## [17] "Foucault"        "Derrida"         "Deleuze"         "Merleau-Ponty"
## [21] "Husserl"         "Heidegger"       "Kant"            "Fichte"
## [25] "Hegel"           "Marx"            "Lenin"           "Smith"
## [29] "Ricardo"         "Keynes"          "Epictetus"       "Marcus Aurelius"
## [33] "Nietzsche"       "Wollstonecraft"  "Beauvoir"        "Davis"
```

```
unique(data.raw$school)
```

```
##  [1] "plato"           "aristotle"       "empiricism"      "rationalism"
##  [5] "analytic"        "continental"     "phenomenology"   "german_idealism"
##  [9] "communism"       "capitalism"      "stoicism"        "nietzsche"
## [13] "feminism"
```

```
summary(data.raw$original_publication_date)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    -350    1641    1817    1327    1949    1985
```

```
summary(unique(data.raw$title))
```

```
##    Length     Class      Mode
##        59 character character
```

The data contains 59 titles for 13 schools of thought so we can expect enough datapoints to run a regression on most schools.

Step 2 - Text processing

```
sentenceCorpus <- Corpus(VectorSource(data.raw$sentence_lowered))
sentenceCorpus<-tm_map(sentenceCorpus, removeWords, stopwords("english"))
```

```
## Warning in tm_map.SimpleCorpus(sentenceCorpus, removeWords,
## stopwords("english")): transformation drops documents
```

```
sentenceCorpus<-tm_map(sentenceCorpus, removeWords, character(0))
```

```
## Warning in tm_map.SimpleCorpus(sentenceCorpus, removeWords, character(0)):
## transformation drops documents
```

```
sentenceCorpus<-tm_map(sentenceCorpus, removePunctuation)
```

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

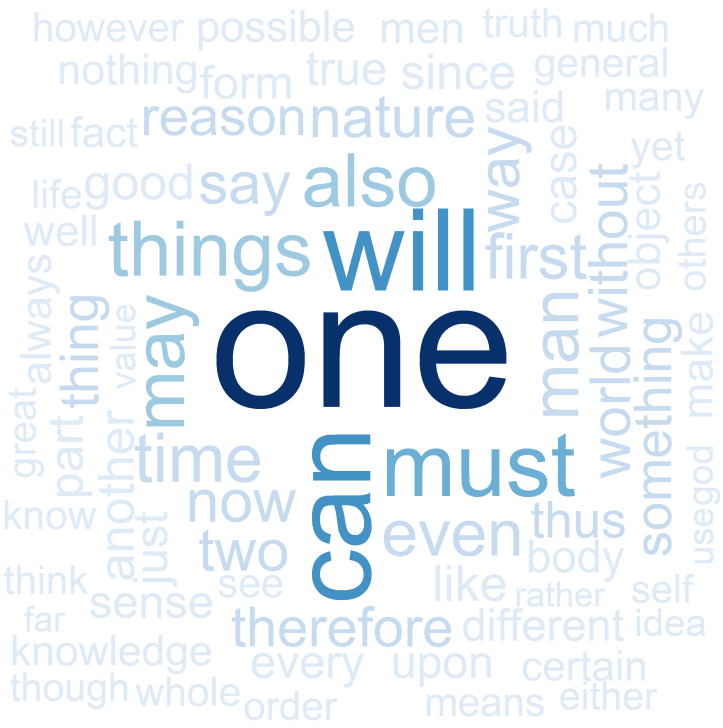
```
sentenceCorpus<-tm_map(sentenceCorpus, stripWhitespace)
```

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

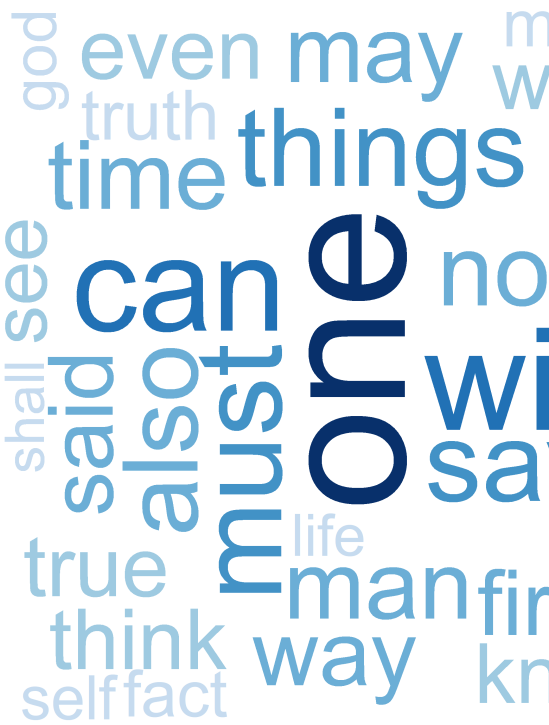
```
tdm <- TermDocumentMatrix(sentenceCorpus)
tdm = removeSparseTerms(tdm, 0.99)
tdm.tidy = tidytext::tidy(tdm)
tdm.overall=summarise(group_by(tdm.tidy, term), sum(count))
```

Step 3 - Overall wordcloud inspection

Generating a wordcloud of the Document Term Matrix cleaned of stopwords and other non-word characters



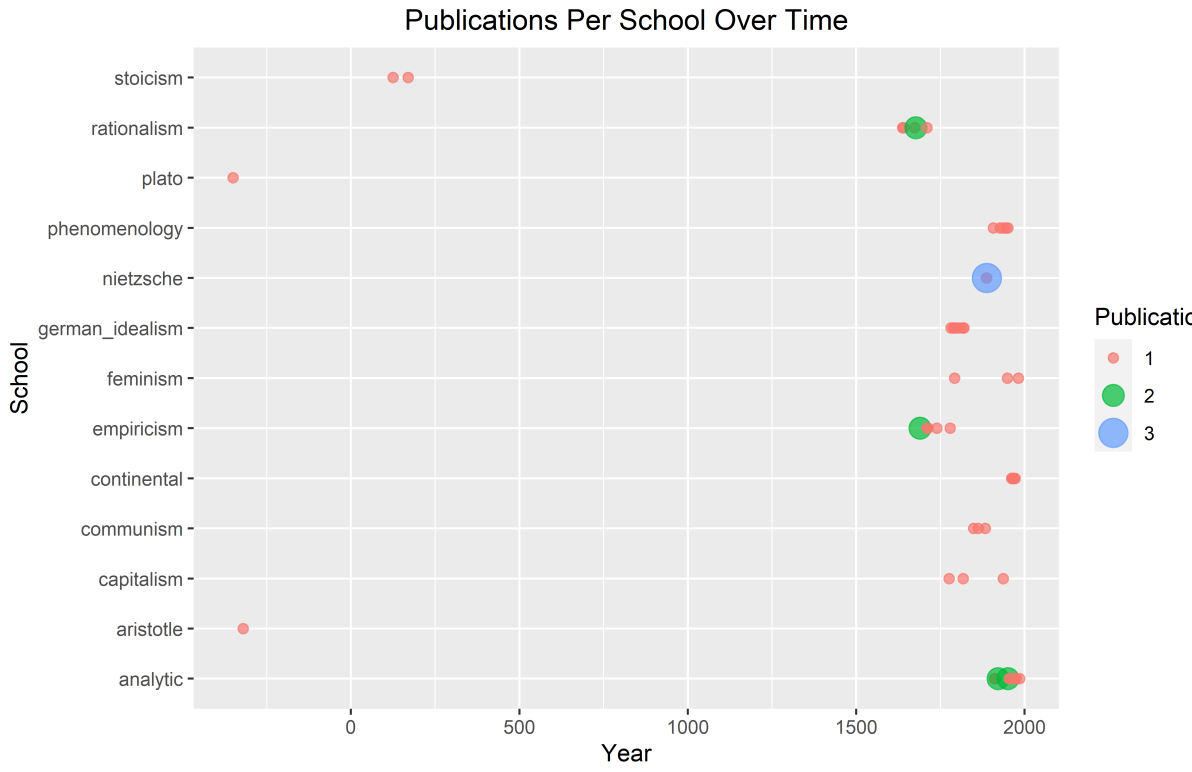
yielded the following:
Overall, the terms seem normal for philosophical texts and confirm that we have appropriate data. Generating



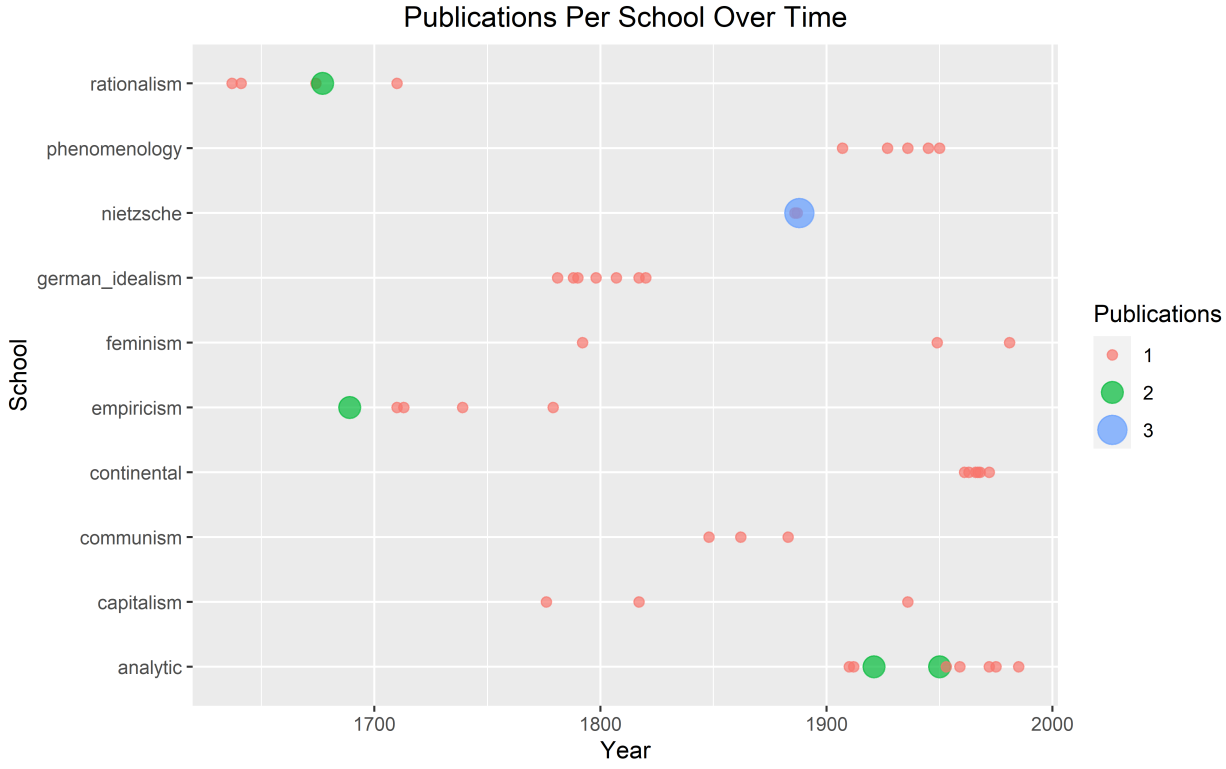
another wordcloud using TF-IDF yielded the following:
The TF-IDF wordcloud is pretty similar to the original which leads me to believe that something went wrong along the way. Luckily, this was primarily an excercise to explore the format of the data and is not relevant to the hypothesis testing.

Step 4 - Grouping publications by year to analyze regressability

Plotting the publications over time to see if there are enough datapoints between the schools to make the analysis worthwhile yields:



Filtering the schools of thought to those with 3 or more publications - so that a regression is nontrivial - yielded the following plot:



From here, most of the remaining schools should have enough datapoints to attempt to fit linear models to them and determine if nonzero coefficients are significant or not.

Step 5 - Loop through the remaining schools and perform regressions to test hypothesis

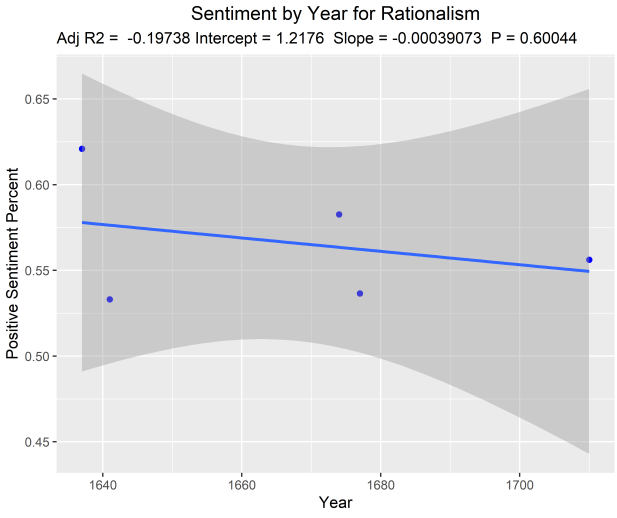
```
#Sentiment Analysis per school using helper function:
PerSchoolSentimentAnalysis <- function(sch, data.gttSenti) {
  schooldata.raw = data.gttSenti[data.gttSenti$school==sch,]

  years = unique(schooldata.raw$original_publication_date)
  schooldata.Senti = data.frame(matrix(ncol = 5, nrow = 0))
  colnames(schooldata.Senti) <- c("year", "negative", "positive", "sentiment", "netSenti")

  for (i in years){
    #per publication sentiment
    schooldata.temptxt = schooldata.raw %>%
      filter(original_publication_date == i)
    schooldata.yrTokens = data_frame(tokens = schooldata.temptxt$tokenized_txt) %>%
      unnest_tokens(word, tokens)
  #reference https://cran.r-project.org/web/packages/tidytext/vignettes/tidytext.html
    schooldata.tmpSenti = schooldata.yrTokens %>%
      inner_join(get_sentiments("bing")) %>%
      count(sentiment) %>%
      spread(sentiment, n, fill = 0) %>%
      mutate(sentiment = positive / (positive + negative), netSenti = positive-negative)
    schooldata.yrSenti = cbind(data_frame(year = i), schooldata.tmpSenti)
    schooldata.Senti = rbind(schooldata.Senti, schooldata.yrSenti)
  }
  percLM = lm(sentiment~year, data = schooldata.Senti)
  netLM = lm(netSenti~year, data = schooldata.Senti)

  percPlot = ggplot(data = schooldata.Senti, aes(x=year, y=sentiment)) +
    geom_point(color='blue') +
    geom_smooth(method = "lm", se = TRUE, formula = y~x)+
    labs(subtitle = paste("Adj R2 = ",signif(summary(percLM)$adj.r.squared, 5),
                          "Intercept =",signif(percLM$coef[[1]],5 ),
                          " Slope =",signif(percLM$coef[[2]], 5),
                          " P =",signif(summary(percLM)$coef[2,4], 5)))+
    ggtitle(paste("Sentiment by Year for", str_to_title(sch)))+
    xlab("Year")+
    ylab("Positive Sentiment Percent")+
    theme(plot.title = element_text(hjust = 0.5))
  netPlot = ggplot(data = schooldata.Senti, aes(x=year, y=netSenti)) +
    geom_point(color='blue') +
    geom_smooth(method = "lm", se = TRUE, formula = y~x)+
    labs(subtitle = paste("Adj R2 = ",signif(summary(netLM)$adj.r.squared, 5),
                          "Intercept =",signif(netLM$coef[[1]],5 ),
                          " Slope =",signif(netLM$coef[[2]], 5),
                          " P =",signif(summary(netLM)$coef[2,4], 5)))+
    ggtitle(paste("Sentiment by Year for", str_to_title(sch)))+
    xlab("Year")+
    ylab("Net Positive Sentiment")+
```

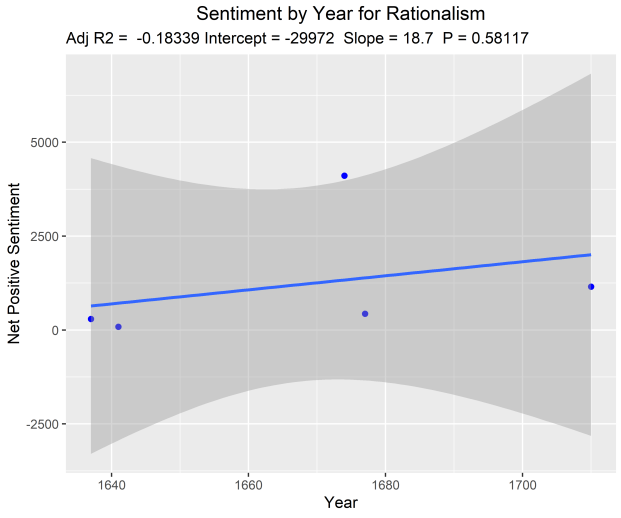
```
    theme(plot.title = element_text(hjust = 0.5))

  png(paste("figs/", sch, "_plots.png", sep=""), units="in", width=12, height=5, res=600)
  grid.arrange(percPlot, netPlot, ncol=2)
  dev.off()

  #Insert plots here in markdown

  #save regression summaries for later
  schooldata.yrlm = cbind(data_frame(school = sch),
                          data_frame(list(percLM)),
                          data_frame(list(netLM)))
  colnames(schooldata.yrlm) <- c("school", "percentLM", "netLM")
  return(schooldata.yrlm)
}
```

```
#filter the dataframe by the schools we want to look at
data.gttSchools = unique(data.activeYearsgtt$school)
data.gttSenti = data.raw %>%
  filter(school %in% data.gttSchools)

#create an empty dataframe to store the linear models in for plotting
data.schoolLM = data.frame(matrix(ncol = 3, nrow = 0))
colnames(data.schoolLM) <- c("school", "percentLM", "netLM")

# loop over the interested schools using the helper function above
for (school in data.gttSchools) {
  data.schoolLM = rbind(data.schoolLM,
                        PerSchoolSentimentAnalysis(school, data.gttSenti))
}
```

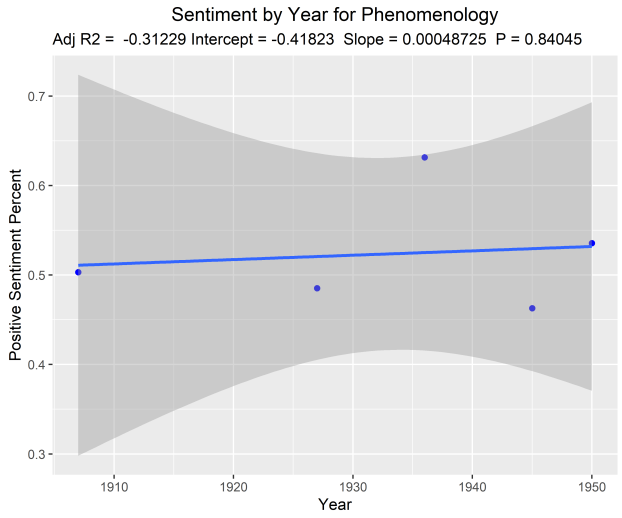
The following plots were generated for each school (in reverse alpha. order):

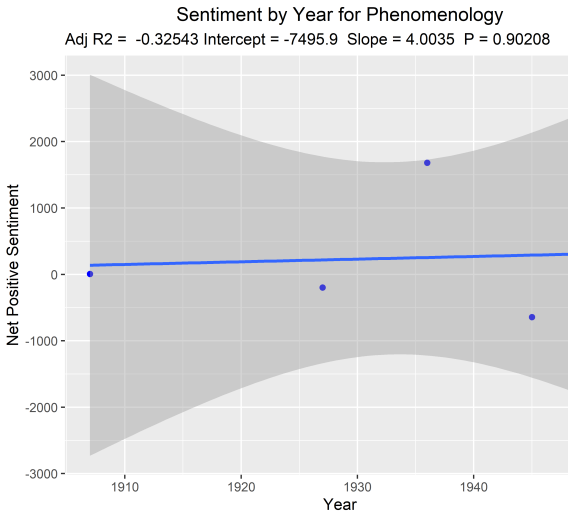
Rationalism:



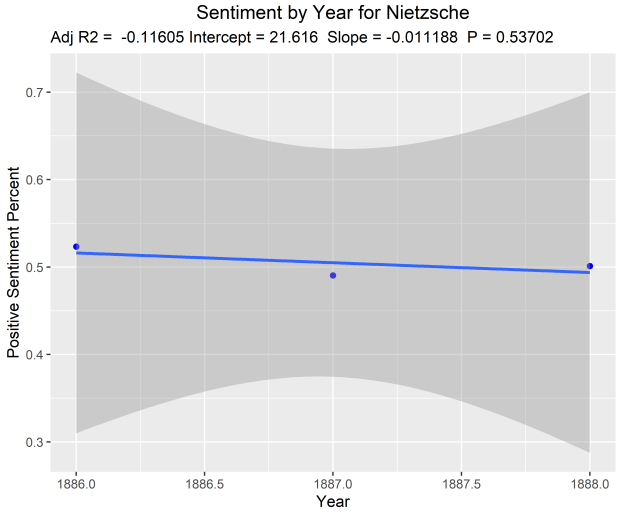


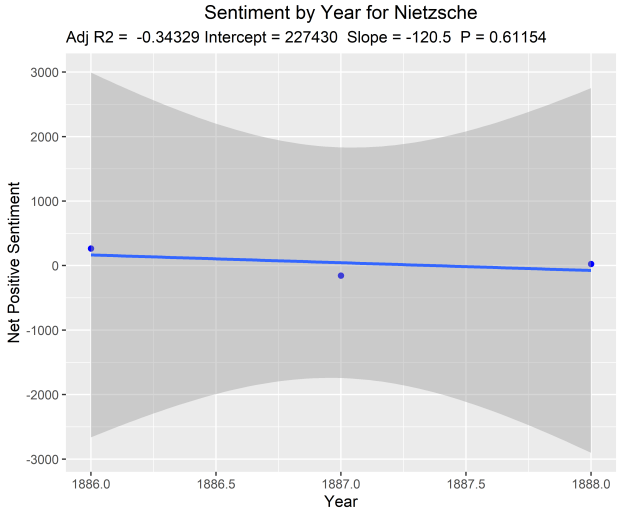
Phenomenology:



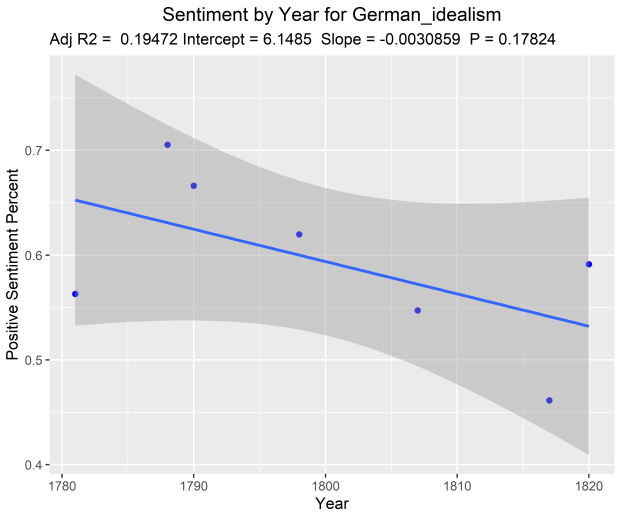


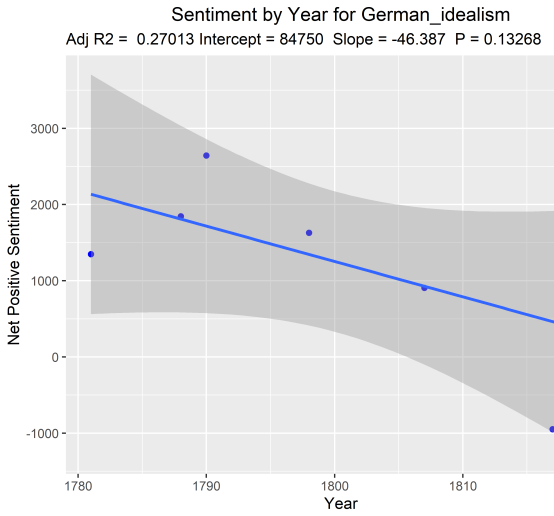
Nietzsche:



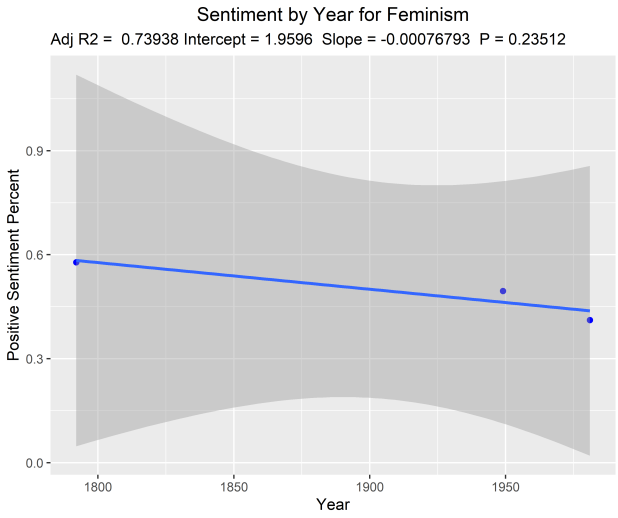


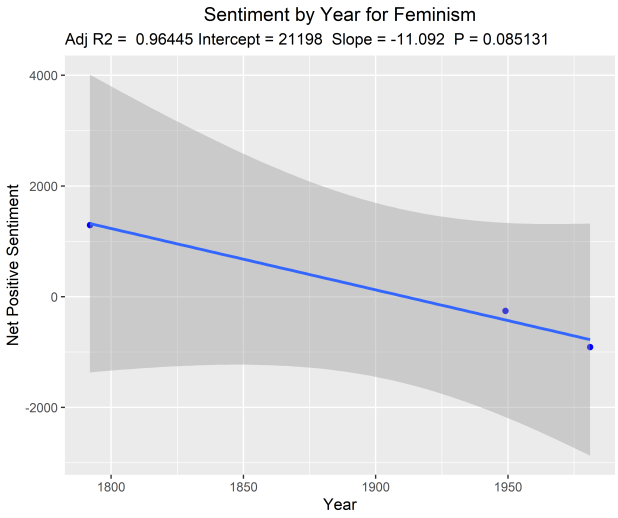
German Idealism:



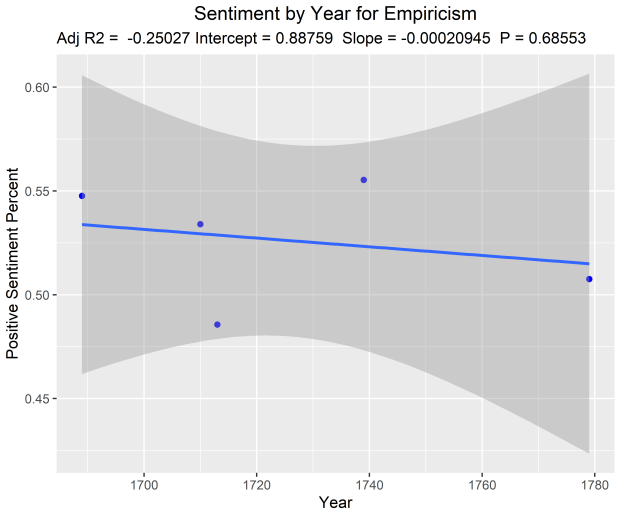


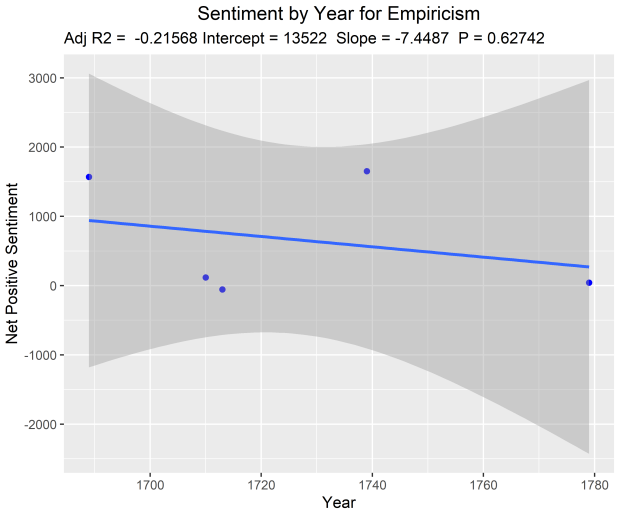
Feminism:



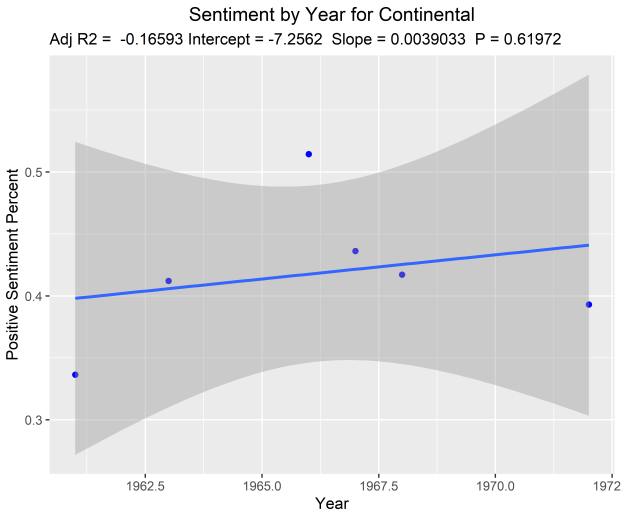


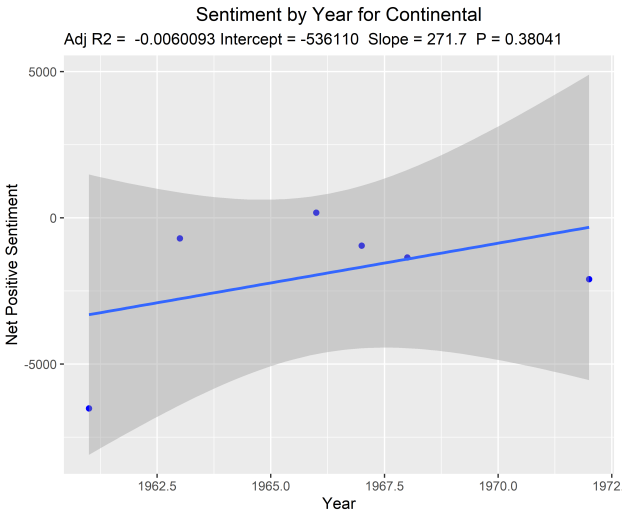
Empiricism:



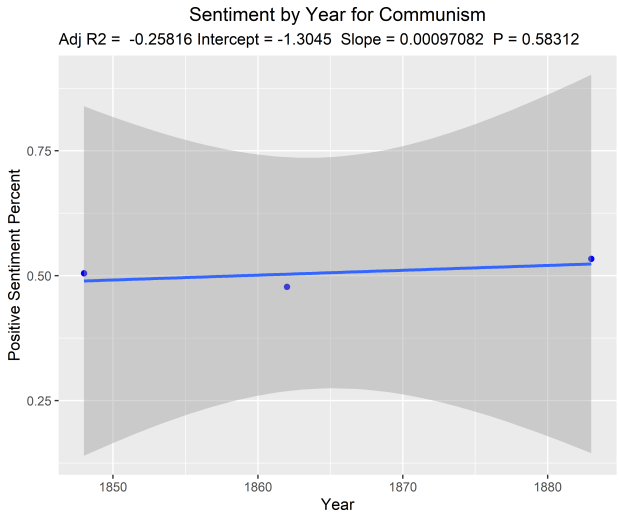


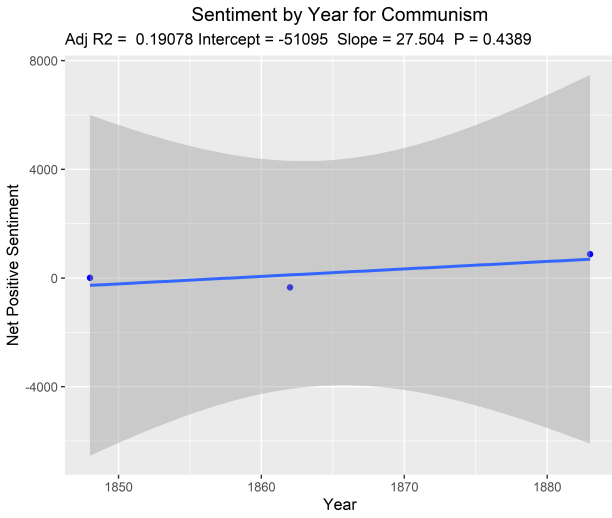
Continental:



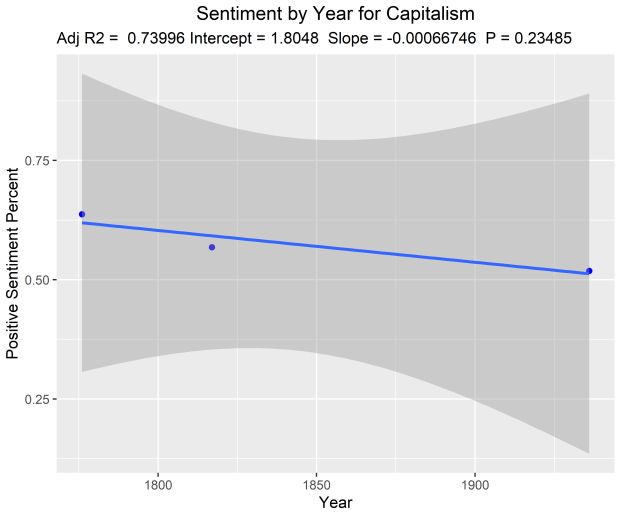


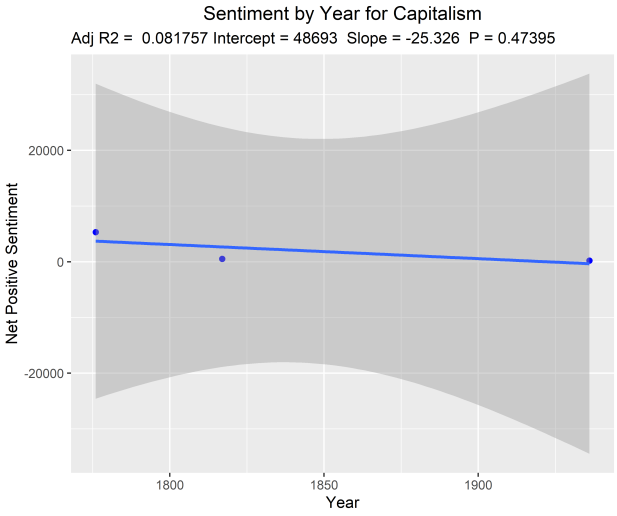
Communism:



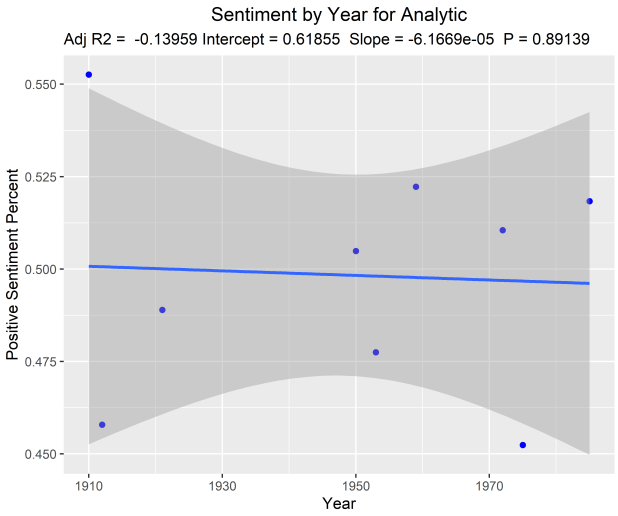


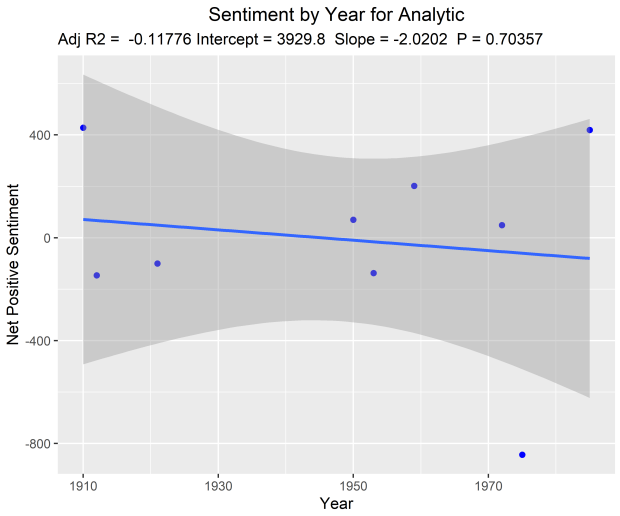
Capitalism:





Analytic:





Results

The schools with the smallest p values were German Idealism and Feminism at P=0.13 and P=0.08 respectively when regressing against the net positive sentiment. Between the two, only Feminism had an adj R2 greater than 0.95 (next highest was Capitalism at 0.74)

Conclusion

Based on the regression results, we can say that Feminism is the only school that could be considered as having a change in sentiment from going net positive in 1792 to net negative by the latest publication 1981. While capitalism's regression was the next closest to explaining the data, it's slope was not far from zero and the sentiment stayed net positive over time.