STA302: Spadina Project

Junwei Chen

2024-10-03

This project critically evaluates the use of I's and i's on a simulated dataset whereby the key novel is The Firefly. The findings suggests that the as the number of words increase, the average number of times I occurs in a sentence, also increase.

Table of contents

Introduction

The current project aims to reproduce *Jane Eyre* by Charlotte Brontë example from the Chapter 13 whereby the name of the novel is changed to *The Firefly* of France Angellotti. In the domain of linguistics, the study of specific words occurrence in of common interest to scholars which allows them to comprehensively evaluate the quality of the literary work. In this report, the similar approach is adopted whereby the novel is changed with initial model modifications to unearth the linguistic characteristics of the Firefly while evaluating the occurrence of vowel term I or i.[1]

Data Management

Packages

```
# Importing important packages
library(boot)
library(broom.mixed)
library(collapse)
library(dataverse)
library(gutenbergr)
library(janitor)
library(knitr)
library(marginaleffects)
library(modelsummary)
library(rstanarm)
library(tidyr)
library(scales)
library(readr)
library(tidyverse)
library(stringr)
library(ggplot2)
```

In this report, several R packages are used whereby (Canty and Ripley 2022), (Bolker and Robinson 2022), (Kuriwaki, Beasley, and Leeper 2023), (Xie 2023) and(Wickham et al. 2019) are used to clean, tidy, sort, and present the data in an appropriate manner. The other libraries such as (Arel-Bundock 2024), (Goodrich et al. 2024), (Wickham 2023), and (Wickham 2016) are used to compute the models and present the findings using appropriate graphs.

[1]Project files: https://github.com/JunweiChen1012/Spadina-Paper.git

Data source and cleaning

The dataset is downloaded using (Johnston and Robinson 2023) library where the information on texts associated with the selected novel are considered. The downloaded data is stored in a .csv file whereby the initial column is an integer and the next column is a text column which stores the respective lines from the main work. The final observation count includes 2 column with 6770 observations in total.

```
# The Firefly of France Angellotti
gutenberg_id_of_firefly <- 3676

firefly <-
  gutenberg_download(
    gutenberg_id = gutenberg_id_of_firefly,
    mirror = "https://gutenberg.pglaf.org/"
  )

firefly
```

```
# A tibble: 6,770 x 2
   gutenberg_id text
          <int> <chr>
 1         3676 "THE FIREFLY OF FRANCE"
 2         3676 ""
 3         3676 "by Marion Polk Angellotti"
 4         3676 ""
 5         3676 ""
 6         3676 ""
 7         3676 ""
 8         3676 "TO"
 9         3676 ""
10         3676 "THE MEMORY OF"
# i 6,760 more rows
```

```
write.csv(firefly, "firefly.csv")
```

```
firefly <- read_csv(
  "firefly.csv",
  col_types = cols(
    gutenberg_id = col_integer(),
    text = col_character()
```

```
  )
)
```

```
New names:
* `` -> `...1`
```

```
firefly
```

```
# A tibble: 6,770 x 3
    ...1 gutenberg_id text
   <dbl>        <int> <chr>
 1     1         3676 THE FIREFLY OF FRANCE
 2     2         3676 <NA>
 3     3         3676 by Marion Polk Angellotti
 4     4         3676 <NA>
 5     5         3676 <NA>
 6     6         3676 <NA>
 7     7         3676 <NA>
 8     8         3676 TO
 9     9         3676 <NA>
10    10         3676 THE MEMORY OF
# i 6,760 more rows
```

In the next section, several empty lines are removed, chapter number, and conclusion are removed including the extra use E's that are not related to the analysis. Thus, final cleaning of the data renders 4 variables and 270 observations. The initial column inlcudes text, next column comprise chapter, then the third chapter includes overall count of I's in the text selected, and the final column includes the overall word count.

```
firefly_reduced <-
  firefly |>
  filter(!is.na(text)) |> # Remove empty lines
  mutate(chapter = if_else(str_detect(text, "CHAPTER") == TRUE,
                           text,
                           NA_character_)) |> # Find start of chapter
  fill(chapter, .direction = "down") |>
  mutate(chapter_line = row_number(),
         .by = chapter) |> # Add line number to each chapter
  filter(!is.na(chapter),
         chapter_line %in% c(2:11)) |> # Remove "CHAPTER I" etc
```

```
  select(text, chapter) |>
  mutate(
    chapter = str_remove(chapter, "CHAPTER "),
    chapter = str_remove(chapter, "-CONCLUSION"),
    chapter = as.integer(as.roman(chapter))
  ) |> # Change chapters to integers
  mutate(count_e = str_count(text, "i|I"),
         word_count = str_count(text, "\\w+")
         # From: https://stackoverflow.com/a/38058033
         )
```

```
firefly_reduced |>
  select(chapter, word_count, count_e, text) |>
  head()
```

```
# A tibble: 6 x 4
  chapter word_count count_e text
    <int>      <int>   <int> <chr>
1       1          3       1 ALARUMS AND EXCURSIONS
2       1         16       3 The restaurant of the Hotel St. Ives seems, as I l~
3       1         14       2 spot to have served as stage wings for a melodrama~
4       1         13       4 a melodrama did begin there. No other word fits th~
5       1         12       5 of the Middle Ages, which, I believe, reeked with ~
6       1          9       4 cutthroats, pistols and poisoned daggers, offered ~
```

Findings and Discussion

The findings from the first histogram as shown in Figure 1 suggests that the overall number of I's in the cleaned dataset comprise around 3 to 4 number of I's as compared to the other characters. The variance suggested from the blue line indicates that the variation in the occurrence of I's is greater than the average value of the vowel occurence in the cleaned dataset.

Figure 2 indicates that the relationship between the number of I's in the dataset and the number of words are linear and correlated.

Figure 3 suggests that the as the number of words increased, the likelihood of I's occurrence also increased in the work which indicates that the prevous works of the authors tend to extensively use I's when the scope of work expanded. These findings also highlights the significance of the use of vocabulory during the time of this novel which offers critical insights on the writing pattern for linguistics enthusiasts and researchers.

```r
#Histogram with Mean and Variance Lines
mean_e <- mean(firefly_reduced$count_e)
variance_e <- var(firefly_reduced$count_e)

firefly_reduced |>
  ggplot(aes(x = count_e)) +
  geom_histogram() +
  geom_vline(xintercept = mean_e,
             linetype = "dashed",
             color = "#C64191") +
  geom_vline(xintercept = variance_e,
             linetype = "dashed",
             color = "#0ABAB5") +
  theme_minimal() +
  labs(
    y = "Count",
    x = "Figure 1: Number of I's per line for first ten lines"
  )
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

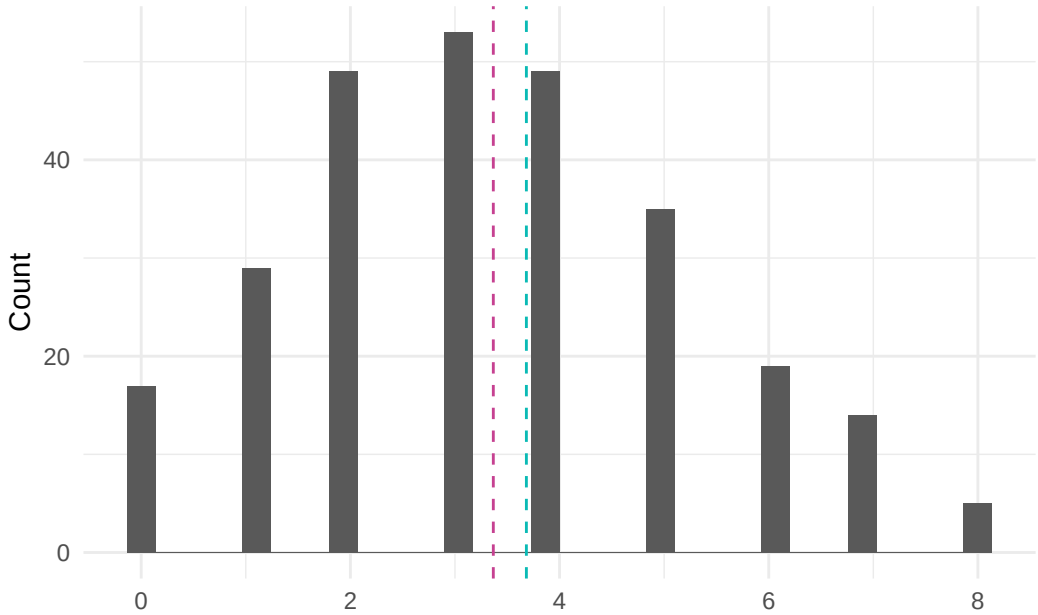
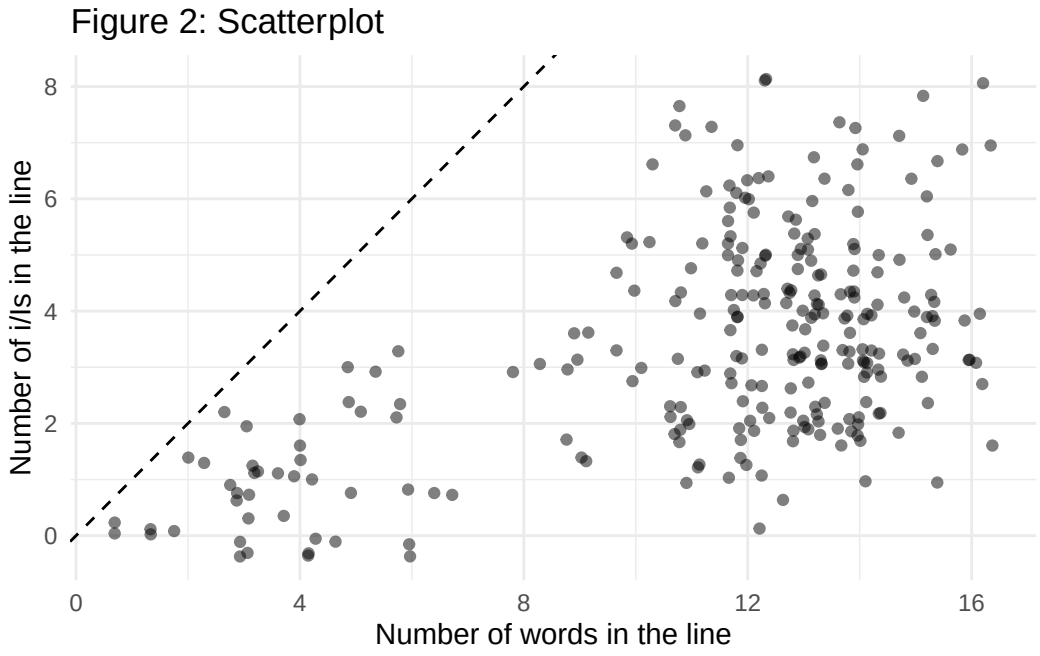


Figure 1: Number of I's per line for first ten lines

```r
#Scatter Plot with Jitter and Reference Line
firefly_reduced |>
  ggplot(aes(x = word_count, y = count_e)) +
  geom_jitter(alpha = 0.5) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed") +
  theme_minimal() +
  labs(
    x = "Number of words in the line",
    y = "Number of i/Is in the line",
    title = "Figure 2: Scatterplot"
  )
```




```r
firefly_e_counts <-
  stan_glm(
    count_e ~ word_count,
    data = firefly_reduced,
    family = poisson(link = "log"),
    prior = normal(location = 0, scale = 2.5, autoscale = TRUE),
    prior_intercept = normal(location = 0, scale = 2.5, autoscale = TRUE),
    seed = 853
  )
```

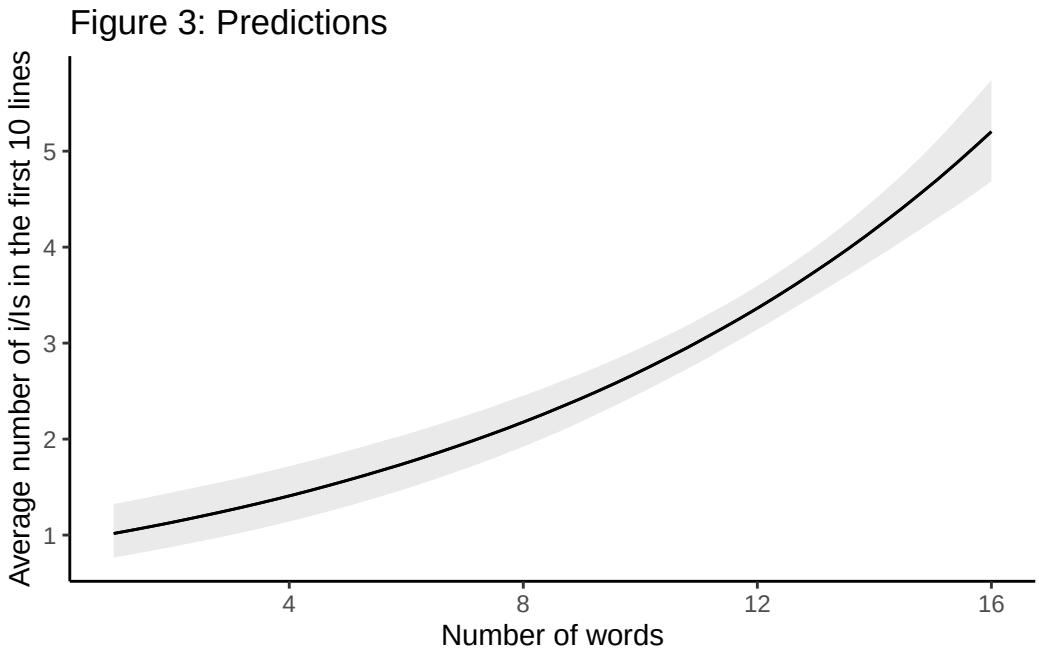
```
SAMPLING FOR MODEL 'count' NOW (CHAIN 1).
Chain 1:
Chain 1: Gradient evaluation took 4.2e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.42 seconds.
Chain 1: Adjust your expectations accordingly!
Chain 1:
Chain 1:
Chain 1: Iteration:    1 / 2000 [  0%]  (Warmup)
Chain 1: Iteration:  200 / 2000 [ 10%]  (Warmup)
Chain 1: Iteration:  400 / 2000 [ 20%]  (Warmup)
Chain 1: Iteration:  600 / 2000 [ 30%]  (Warmup)
Chain 1: Iteration:  800 / 2000 [ 40%]  (Warmup)
Chain 1: Iteration: 1000 / 2000 [ 50%]  (Warmup)
Chain 1: Iteration: 1001 / 2000 [ 50%]  (Sampling)
Chain 1: Iteration: 1200 / 2000 [ 60%]  (Sampling)
Chain 1: Iteration: 1400 / 2000 [ 70%]  (Sampling)
Chain 1: Iteration: 1600 / 2000 [ 80%]  (Sampling)
Chain 1: Iteration: 1800 / 2000 [ 90%]  (Sampling)
Chain 1: Iteration: 2000 / 2000 [100%]  (Sampling)
Chain 1:
Chain 1:  Elapsed Time: 0.074 seconds (Warm-up)
Chain 1:                0.081 seconds (Sampling)
Chain 1:                0.155 seconds (Total)
Chain 1:

SAMPLING FOR MODEL 'count' NOW (CHAIN 2).
Chain 2:
Chain 2: Gradient evaluation took 1.5e-05 seconds
Chain 2: 1000 transitions using 10 leapfrog steps per transition would take 0.15 seconds.
Chain 2: Adjust your expectations accordingly!
Chain 2:
Chain 2:
Chain 2: Iteration:    1 / 2000 [  0%]  (Warmup)
Chain 2: Iteration:  200 / 2000 [ 10%]  (Warmup)
Chain 2: Iteration:  400 / 2000 [ 20%]  (Warmup)
Chain 2: Iteration:  600 / 2000 [ 30%]  (Warmup)
Chain 2: Iteration:  800 / 2000 [ 40%]  (Warmup)
Chain 2: Iteration: 1000 / 2000 [ 50%]  (Warmup)
Chain 2: Iteration: 1001 / 2000 [ 50%]  (Sampling)
Chain 2: Iteration: 1200 / 2000 [ 60%]  (Sampling)
Chain 2: Iteration: 1400 / 2000 [ 70%]  (Sampling)
Chain 2: Iteration: 1600 / 2000 [ 80%]  (Sampling)
Chain 2: Iteration: 1800 / 2000 [ 90%]  (Sampling)
```

```
Chain 2: Iteration: 2000 / 2000 [100%]  (Sampling)
Chain 2:
Chain 2:  Elapsed Time: 0.067 seconds (Warm-up)
Chain 2:                0.083 seconds (Sampling)
Chain 2:                0.15 seconds (Total)
Chain 2:

SAMPLING FOR MODEL 'count' NOW (CHAIN 3).
Chain 3:
Chain 3: Gradient evaluation took 1.5e-05 seconds
Chain 3: 1000 transitions using 10 leapfrog steps per transition would take 0.15 seconds.
Chain 3: Adjust your expectations accordingly!
Chain 3:
Chain 3:
Chain 3: Iteration:    1 / 2000 [  0%]  (Warmup)
Chain 3: Iteration:  200 / 2000 [ 10%]  (Warmup)
Chain 3: Iteration:  400 / 2000 [ 20%]  (Warmup)
Chain 3: Iteration:  600 / 2000 [ 30%]  (Warmup)
Chain 3: Iteration:  800 / 2000 [ 40%]  (Warmup)
Chain 3: Iteration: 1000 / 2000 [ 50%]  (Warmup)
Chain 3: Iteration: 1001 / 2000 [ 50%]  (Sampling)
Chain 3: Iteration: 1200 / 2000 [ 60%]  (Sampling)
Chain 3: Iteration: 1400 / 2000 [ 70%]  (Sampling)
Chain 3: Iteration: 1600 / 2000 [ 80%]  (Sampling)
Chain 3: Iteration: 1800 / 2000 [ 90%]  (Sampling)
Chain 3: Iteration: 2000 / 2000 [100%]  (Sampling)
Chain 3:
Chain 3:  Elapsed Time: 0.07 seconds (Warm-up)
Chain 3:                0.087 seconds (Sampling)
Chain 3:                0.157 seconds (Total)
Chain 3:

SAMPLING FOR MODEL 'count' NOW (CHAIN 4).
Chain 4:
Chain 4: Gradient evaluation took 1.4e-05 seconds
Chain 4: 1000 transitions using 10 leapfrog steps per transition would take 0.14 seconds.
Chain 4: Adjust your expectations accordingly!
Chain 4:
Chain 4:
Chain 4: Iteration:    1 / 2000 [  0%]  (Warmup)
Chain 4: Iteration:  200 / 2000 [ 10%]  (Warmup)
Chain 4: Iteration:  400 / 2000 [ 20%]  (Warmup)
Chain 4: Iteration:  600 / 2000 [ 30%]  (Warmup)
```

```
Chain 4: Iteration:  800 / 2000 [ 40%]  (Warmup)
Chain 4: Iteration: 1000 / 2000 [ 50%]  (Warmup)
Chain 4: Iteration: 1001 / 2000 [ 50%]  (Sampling)
Chain 4: Iteration: 1200 / 2000 [ 60%]  (Sampling)
Chain 4: Iteration: 1400 / 2000 [ 70%]  (Sampling)
Chain 4: Iteration: 1600 / 2000 [ 80%]  (Sampling)
Chain 4: Iteration: 1800 / 2000 [ 90%]  (Sampling)
Chain 4: Iteration: 2000 / 2000 [100%]  (Sampling)
Chain 4:
Chain 4:  Elapsed Time: 0.07 seconds (Warm-up)
Chain 4:                0.081 seconds (Sampling)
Chain 4:                0.151 seconds (Total)
Chain 4:
```

```r
saveRDS(
  firefly_e_counts,
  file = "firefly_e_counts.rds"
)
```

```r
plot_predictions(firefly_e_counts, condition = "word_count") +
  labs(x = "Number of words",
       y = "Average number of i/Is in the first 10 lines",
       title ="Figure 3: Predictions" ) +
  theme_classic()
```



Conclusion

The project implements a general analysis using simulated and cleaned dataset related to a novel, to evaluate the use of vowels like I's in the literary works. The key finding from the analysis suggests that the use of I's or i's increased in the works of scholars and writers as it is a vowel and mainly used in various English words to establish a certain meaning. The core limitation of the analysis is lack of evidence pertaining to the hypotheses testing, for example, researchers can hypothesise wheather a vowel like "I' occur more than 5 times in a sentence comprising 10 words. This will enable linguist researchers to compare the writing pattern and vovvcabolary use of the writers from different time period or contemporary time period.

References

Arel-Bundock, Vincent. 2024. "Marginaleffects: Predictions, Comparisons, Slopes, Marginal Means, and Hypothesis Tests." https://marginaleffects.com/.

Bolker, Ben, and David Robinson. 2022. "Broom.mixed: Tidying Methods for Mixed Models." https://github.com/bbolker/broom.mixed.

Canty, Angelo, and B. D. Ripley. 2022. "Boot: Bootstrap r (s-Plus) Functions."

Goodrich, Ben, Jonah Gabry, Imad Ali, and Sam Brilleman. 2024. "Rstanarm: {Bayesian} Applied Regression Modeling via {Stan}." https://mc-stan.org/rstanarm/.

Johnston, Myfanwy, and David Robinson. 2023. "Gutenbergr: Download and Process Public Domain Works from Project Gutenberg." https://docs.ropensci.org/gutenbergr/.

Kuriwaki, Shiro, Will Beasley, and Thomas J. Leeper. 2023. "Dataverse: R Client for Dataverse 4+ Repositories."

Wickham, Hadley. 2016. "Ggplot2: Elegant Graphics for Data Analysis." https://ggplot2.tidyverse.org.

———. 2023. "Stringr: Simple, Consistent Wrappers for Common String Operations." https://stringr.tidyverse.org.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the {Tidyverse}" 4: 1686. https://doi.org/10.21105/joss.01686.

Xie, Yihui. 2023. "Knitr: A General-Purpose Package for Dynamic Report Generation in r." https://yihui.org/knitr/.