Final Project

Patrick May

2023-11-18

```
# Load data
isdata <- read.csv("~/Datasets/ISData5.csv", header = TRUE, sep = ',', row.names = NULL, strings
AsFactors = FALSE, quote = "", encoding = "UTF-8")
```

Classifications of observations

Of the 94 columns in the dataset, many are lexical in nature. Here is a further breakdown of what they measure, according to the text complexity package (https://github.com/tsproisl/textcomplexity/tree/master)

Nearly all variables are paired: `{...}` Val and `{...}` Stdev, corresponding to `{...}` 's value and standard deviation, respectively. For these classifications below, we list them without either suffix.

Measures of Sample-Size/Vocab Size

- `typeTokenRatio`
- `brunetsW`
- `cttr`
- `dugastsU`
- `dugastsK`
- `guiraudsR`
- `herdansC`
- `maasAsq`
- `summersS`
- `tuldavasLn`

Measures of Frequency Spectrum

- `entropy`
- `evenness`
- `herdansVm`
- `jarvisEven`
- `hd.d`
- `simpsonsD`
- `yulesK`

Probabilistic Model Parameters

- `orlovsZ`

Whole Text Measures

- `avgTokLen`
- `log10TextLen`
- `mtld`

Dispersion Measurements

- evenDisp
- giniDisp

Sentence Measures

- avgToksSent
- avgWordsPerSent
- punctPerTok
- punctPerSent

Part Of Speech Measures

- lexDensity
- lexRarity

Dependency-Based Measures

- avgDepDist
- closeCentrality
- depsPerWord
- longestShortest
- outdegreeCentralization

Cleaning NAs

```
colSums(is.na(isdata))
```

```
##                  author                   title                 pubdate
##                       0                       0                       0
##              isexemplar                   dept1                   dept2
##                       0                       0                       0
##                  dept3.                   wordc                    figc
##                       0                       0                       0
##                   pagec                    len1                    len2
##                       0                       0                       0
##                    len3                    len4                    len5
##                       0                       0                       0
##                    len6                    len7                    len8
##                       0                       0                       0
##                    len9                   len10                   len11
##                       0                       0                       0
##                   len12                   len13                   len14
##                       0                       0                       0
##                   len15            log10TextLen             punctPerTok
##                       0                     221                     221
##          typeTokenRatio                    mtld             lexDensity
##                     221                     221                     221
##              lexRarity           avgToksSentVal        avgToksSentStdev
##                     221                     221                     221
##       typeTokenRatioVal      typeTokenRatioStdev             guiradsRVal
##                     221                     221                     221
##           guiradsRStdev             herdansCVal          herdansCStdev
##                     221                     221                     221
##            dugastsKVal           dugastsKStdev              maasAsqVal
##                     221                     221                     221
##           maasAsqStdev             dugastsUVal          dugastsUStdev
##                     221                     221                     221
##          tuldavasLnVal          tuldavasLnStdev            brunetsWVal
##                     221                     221                     221
##          brunetsWStdev                 cttrVal              cttrStdev
##                     221                     221                     221
##             summerSVal            summerSStdev             sichelsVal
##                     221                     221                     221
##           sichelsStdev              micheaMVal           micheaMStdev
##                     221                     221                     221
##             honoreHVal            honoreHStdev             entropyVal
##                     221                     221                     221
##           entropyStdev             evennessVal          evennessStdev
##                     221                     221                     221
##          jarvisEvenVal          jarvisEvenStdev                yuleKVal
##                     221                     221                     221
##             yuleKStdev             simpsonDVal          simpsonDStdev
##                     221                     221                     221
##            herdanVmVal            herdanVmStdev                hd.dVal
##                     221                     221                     221
##              hd.dStdev             avgTokLenVal         avgTokLenStdev
##                     221                     221                     221
##             orlovZVal              orlovZStdev            giniDispVal
##                     221                     221                     221
```

```
##             giniDispStdev               evenDispVal             evenDispStdev
##                       221                       221                       221
##              punctPerSent       avgWordsPerSentVal     avgWordsPerSentStdev
##                       221                       221                       221
##             avgDepDistVal          avgDepDistStdev       closeCentralityVal
##                       221                       221                       221
##      closeCentralityStdev  outdegCentralizationVal outdegCentralizationStdev
##                       221                       221                       221
##        longestShortestVal     longestShortestStdev            depsPerWordVal
##                       221                       221                       221
##          depsPerWordStdev
##                       221
```

```
# hey! 222 missing values! much much much better.

# drop all nulls... as these would be likely large outliers
isdata.noNA <- na.omit(isdata)
#isdata.noNA
colSums(is.na(isdata.noNA))
```

```
##                  author                   title                 pubdate
##                       0                       0                       0
##              isexemplar                   dept1                   dept2
##                       0                       0                       0
##                  dept3.                   wordc                    figc
##                       0                       0                       0
##                   pagec                    len1                    len2
##                       0                       0                       0
##                    len3                    len4                    len5
##                       0                       0                       0
##                    len6                    len7                    len8
##                       0                       0                       0
##                    len9                   len10                   len11
##                       0                       0                       0
##                   len12                   len13                   len14
##                       0                       0                       0
##                   len15            log10TextLen             punctPerTok
##                       0                       0                       0
##          typeTokenRatio                    mtld             lexDensity
##                       0                       0                       0
##              lexRarity           avgToksSentVal        avgToksSentStdev
##                       0                       0                       0
##       typeTokenRatioVal      typeTokenRatioStdev             guiradsRVal
##                       0                       0                       0
##           guiradsRStdev             herdansCVal           herdansCStdev
##                       0                       0                       0
##             dugastsKVal           dugastsKStdev              maasAsqVal
##                       0                       0                       0
##            maasAsqStdev             dugastsUVal           dugastsUStdev
##                       0                       0                       0
##           tuldavasLnVal          tuldavasLnStdev             brunetsWVal
##                       0                       0                       0
##           brunetsWStdev                  cttrVal                cttrStdev
##                       0                       0                       0
##              summerSVal             summerSStdev              sichelsVal
##                       0                       0                       0
##            sichelsStdev              micheaMVal             micheaMStdev
##                       0                       0                       0
##              honoreHVal             honoreHStdev              entropyVal
##                       0                       0                       0
##            entropyStdev             evennessVal           evennessStdev
##                       0                       0                       0
##           jarvisEvenVal          jarvisEvenStdev                yuleKVal
##                       0                       0                       0
##              yuleKStdev              simpsonDVal           simpsonDStdev
##                       0                       0                       0
##             herdanVmVal             herdanVmStdev                 hd.dVal
##                       0                       0                       0
##               hd.dStdev             avgTokLenVal          avgTokLenStdev
##                       0                       0                       0
##              orlovZVal              orlovZStdev             giniDispVal
##                       0                       0                       0
```

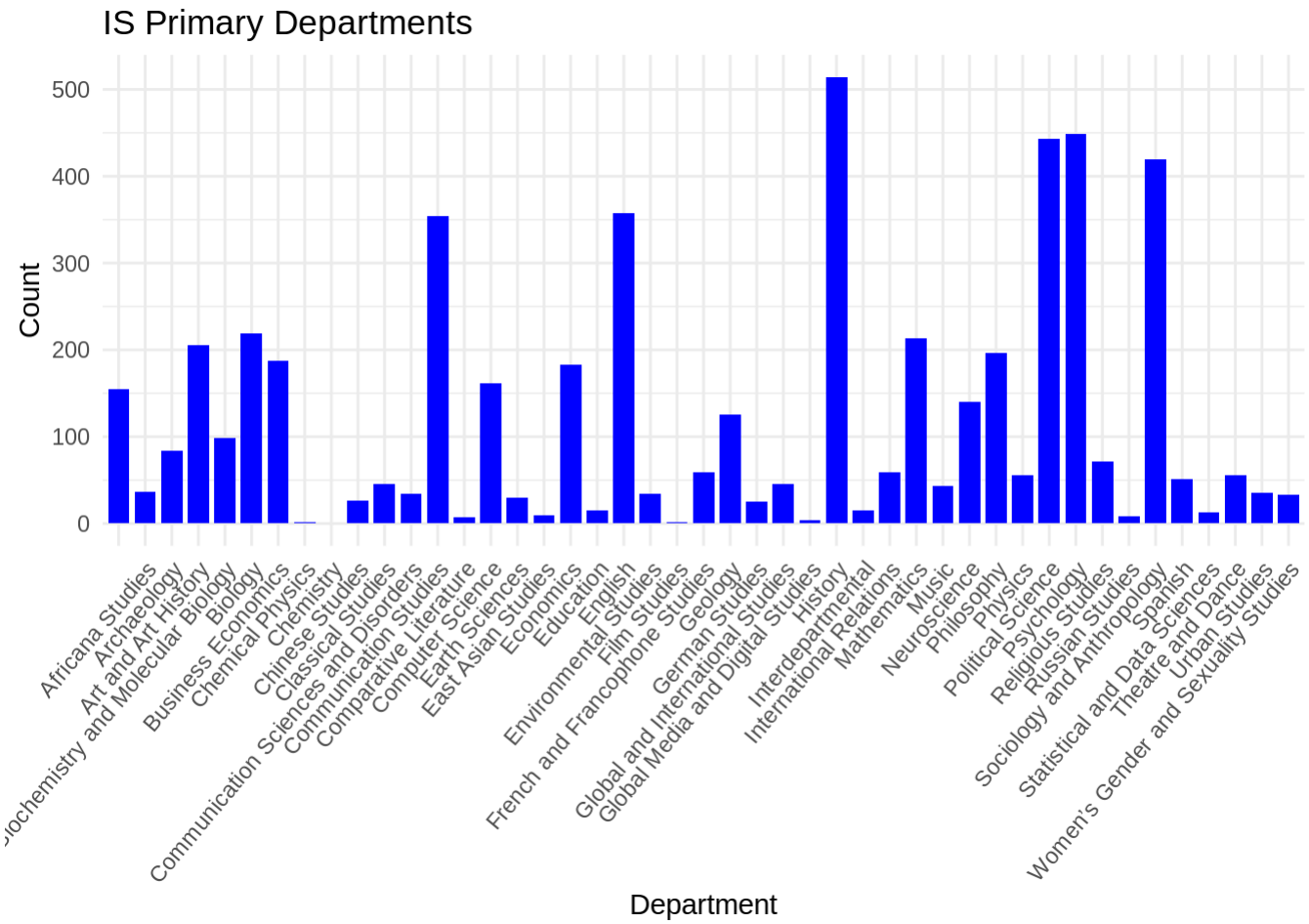
```
##             giniDispStdev               evenDispVal             evenDispStdev
##                         0                         0                         0
##              punctPerSent        avgWordsPerSentVal      avgWordsPerSentStdev
##                         0                         0                         0
##             avgDepDistVal           avgDepDistStdev        closeCentralityVal
##                         0                         0                         0
##      closeCentralityStdev   outdegCentralizationVal outdegCentralizationStdev
##                         0                         0                         0
##        longestShortestVal      longestShortestStdev            depsPerWordVal
##                         0                         0                         0
##          depsPerWordStdev
##                         0
```

```
# something broke in the formatting for JUST this person, because they listed themselves as 7 ma
jors. OK
troublerows <- isdata[isdata$punctPerTok > 1,]
isdata.noNA <- isdata.noNA[isdata.noNA$punctPerTok <= 1,]
```

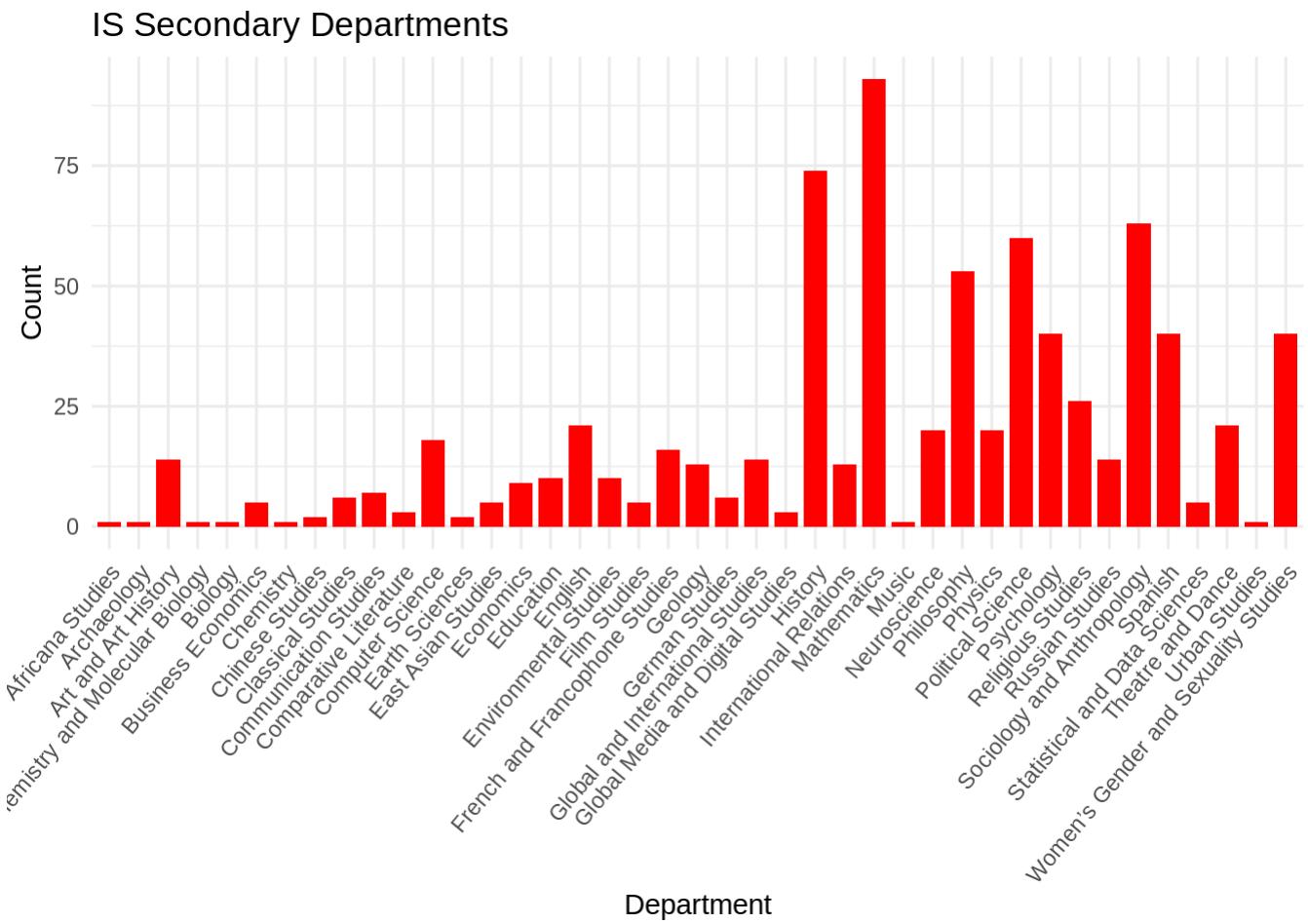
Departmental Breakdown

Since this is a new dataset, some EDA is worth doing for the end paper:

```
# fixing weirdness with showing NA/empty string stuffs
isdata.noNA$new_dept2 <- ifelse(!isdata.noNA$dept2 == "", isdata.noNA$dept2, NA)

# primary
ggplot(isdata.noNA, aes(x = factor(dept1))) +
  geom_bar(fill = "blue", width=0.8) +
  labs(title = "IS Primary Departments",
       x = "Department",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 50, hjust = 1))
```




```
# secondary
ggplot(na.omit(isdata.noNA), aes(x = factor(new_dept2))) +
  geom_bar(fill = "red", width=0.8) +
  labs(title = "IS Secondary Departments",
       x = "Department",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 50, hjust = 1))
```



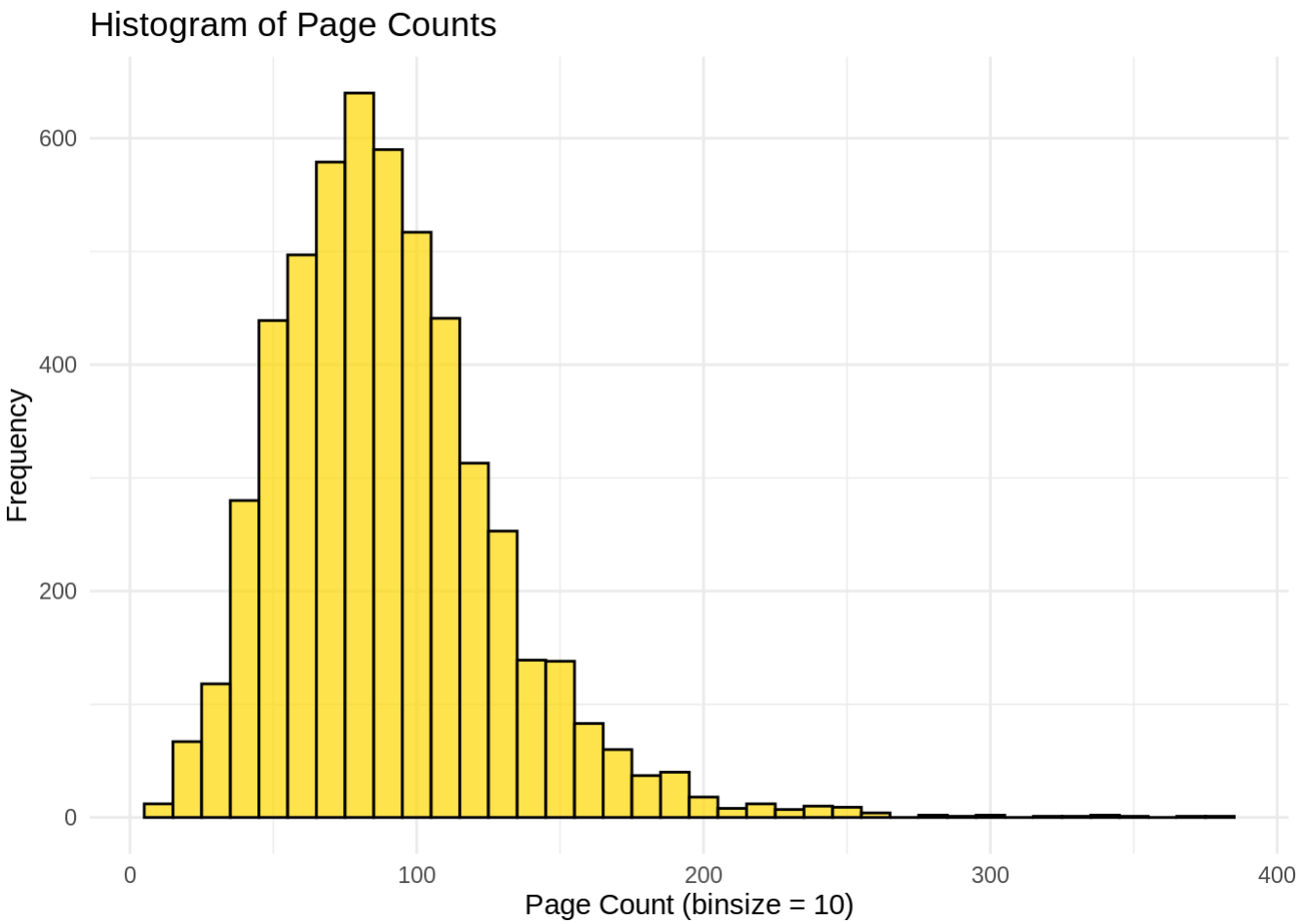

```
sorteddept1 <- table(isdata.noNA$dept1)
sorteddept1[order(sorteddept1, decreasing = TRUE)]
```

```
##
##                             History                          Psychology
##                                 514                                 449
##                   Political Science           Sociology and Anthropology
##                                 443                                 420
##                             English                Communication Studies
##                                 357                                 354
##                             Biology                         Mathematics
##                                 219                                 213
##                 Art and Art History                          Philosophy
##                                 206                                 196
##                  Business Economics                           Economics
##                                 187                                 183
##                    Computer Science
##                                 162                                 155
##                         Neuroscience                            Geology
##                                 140                                 126
##   Biochemistry and Molecular Biology                         Archaeology
##                                  98                                  84
##                   Religious Studies       French and Francophone Studies
##                                  71                                  59
##              International Relations                             Physics
##                                  59                                  56
##                    Theatre and Dance                             Spanish
##                                  56                                  51
##                   Classical Studies     Global and International Studies
##                                  46                                  46
##                               Music                     Africana Studies
##                                  43                                  37
##                       Urban Studies Communication Sciences and Disorders
##                                  35                                  34
##                Environmental Studies Women’s Gender and Sexuality Studies
##                                  34                                  33
##                     Earth Sciences                      Chinese Studies
##                                  30                                  26
##                     German Studies                           Education
##                                  25                                  15
##                   Interdepartmental        Statistical and Data Sciences
##                                  15                                  13
##                  East Asian Studies                      Russian Studies
##                                   9                                   8
##               Comparative Literature     Global Media and Digital Studies
##                                   7                                   4
##                    Chemical Physics                         Film Studies
##                                   2                                   2
##                           Chemistry
##                                   1
```

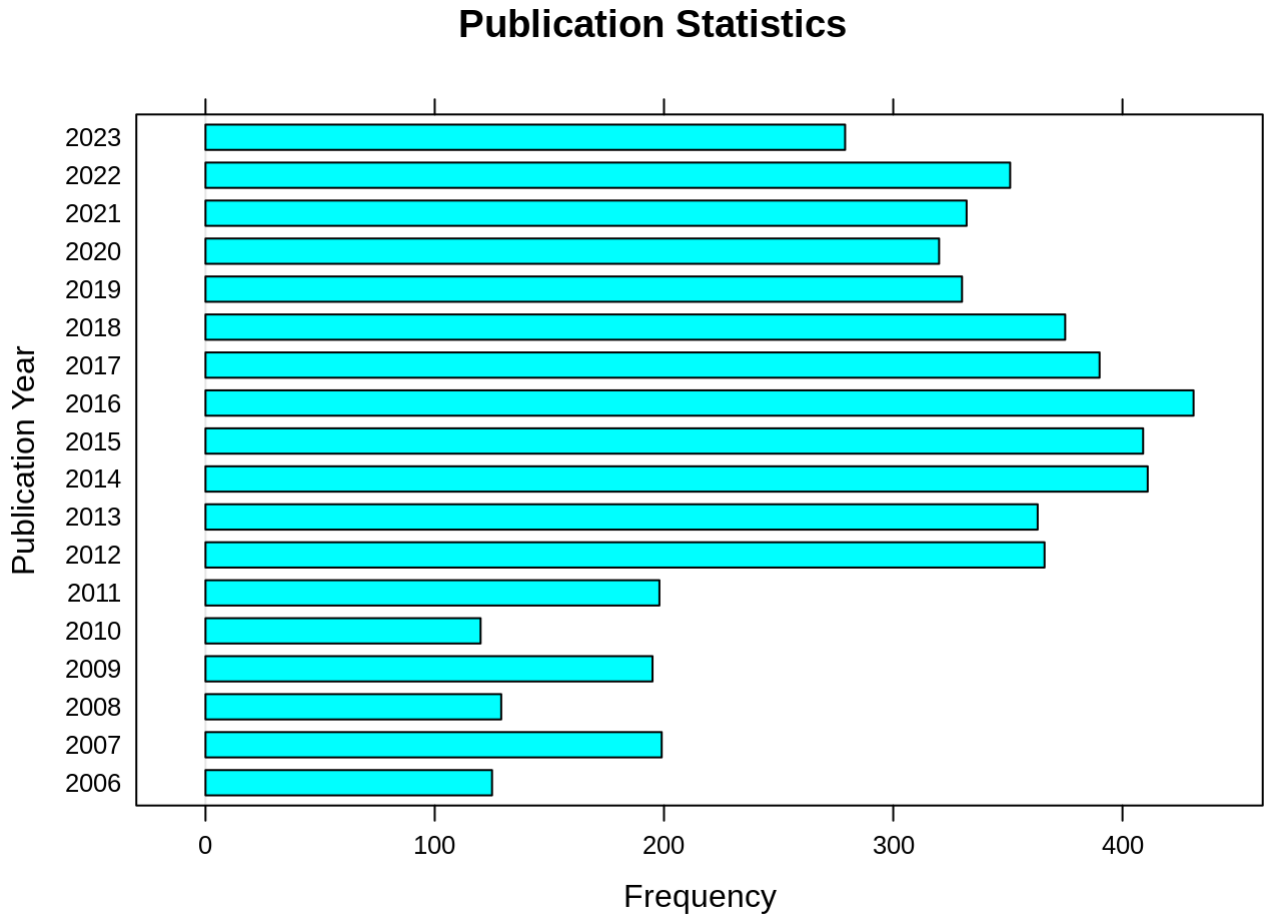
```
sorteddept2 <- table(isdata.noNA$dept2)
sorteddept2[order(sorteddept2, decreasing = TRUE)]
```

```
##
##                                                                 Mathematics
##                                4565                                      93
##                             History                 Sociology and Anthropology
##                                  74                                      63
##                  Political Science                                Philosophy
##                                  60                                      53
##                          Psychology                                 Spanish
##                                  40                                      40
## Women’s Gender and Sexuality Studies                       Religious Studies
##                                  40                                      26
##                             English                       Theatre and Dance
##                                  21                                      21
##                       Neuroscience                                 Physics
##                                  20                                      20
##                    Computer Science           French and Francophone Studies
##                                  18                                      16
##                Art and Art History         Global and International Studies
##                                  14                                      14
##                     Russian Studies                                 Geology
##                                  14                                      13
##             International Relations                               Education
##                                  13                                      10
##               Environmental Studies                               Economics
##                                  10                                       9
##               Communication Studies                       Classical Studies
##                                   7                                       6
##                     German Studies                      Business Economics
##                                   6                                       5
##                  East Asian Studies                            Film Studies
##                                   5                                       5
##       Statistical and Data Sciences                  Comparative Literature
##                                   5                                       3
##    Global Media and Digital Studies                        Chinese Studies
##                                   3                                       2
##                     Earth Sciences                       Africana Studies
##                                   2                                       1
##                         Archaeology   Biochemistry and Molecular Biology
##                                   1                                       1
##                            Biology                                 Chemistry
##                                   1                                       1
##                              Music                             Urban Studies
##                                   1                                       1
```

```
ggplot(isdata.noNA, aes(x = pagec)) +
  geom_histogram(binwidth = 10, fill = "gold", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Page Counts",
       x = "Page Count (binsize = 10)",
       y = "Frequency") +
  theme_minimal()
```

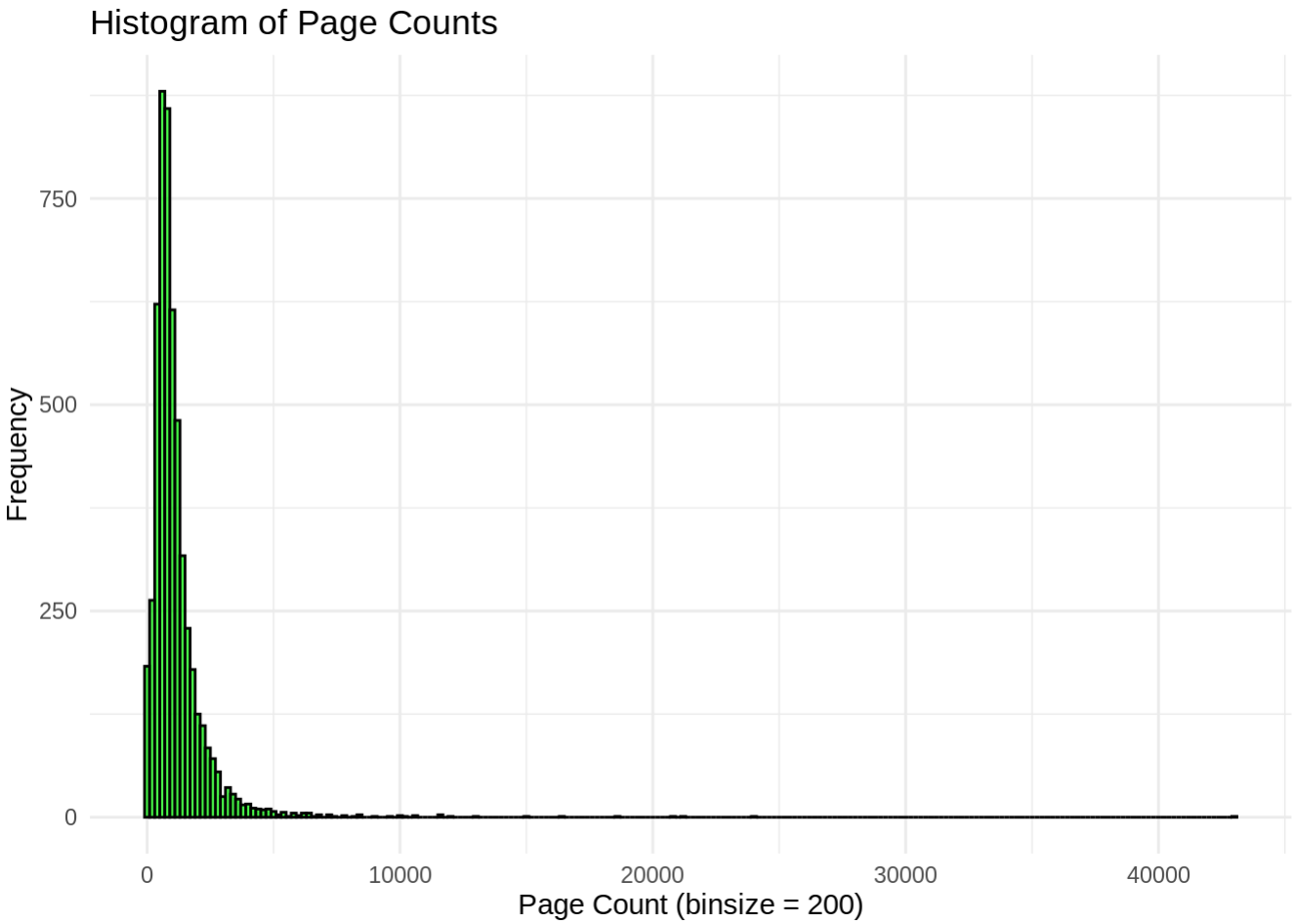



```
barchart(table(isdata.noNA$pubdate), ylab = "Publication Year", xlab = "Frequency", main = "Publ
ication Statistics")
```

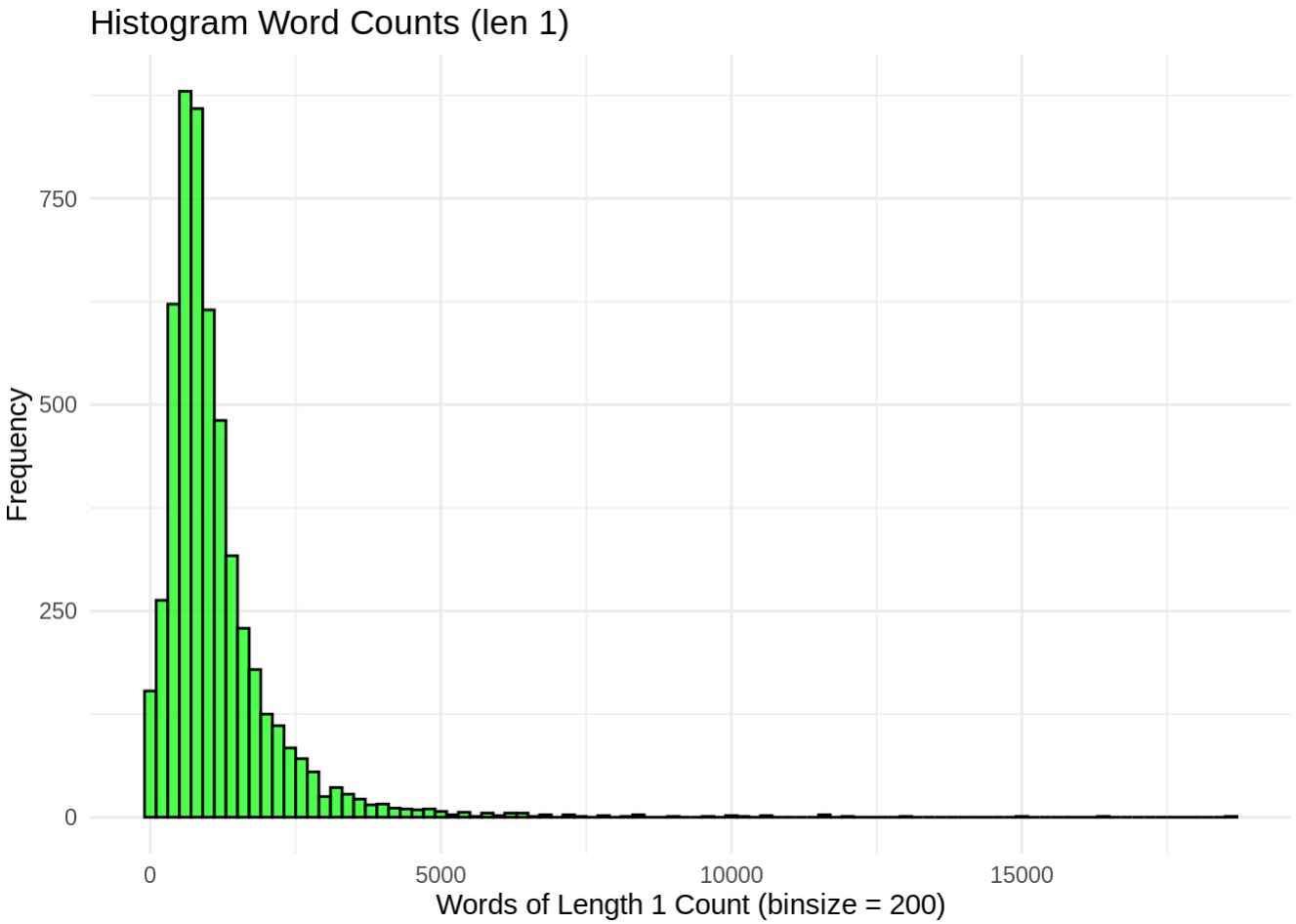


Word Length Frequencies

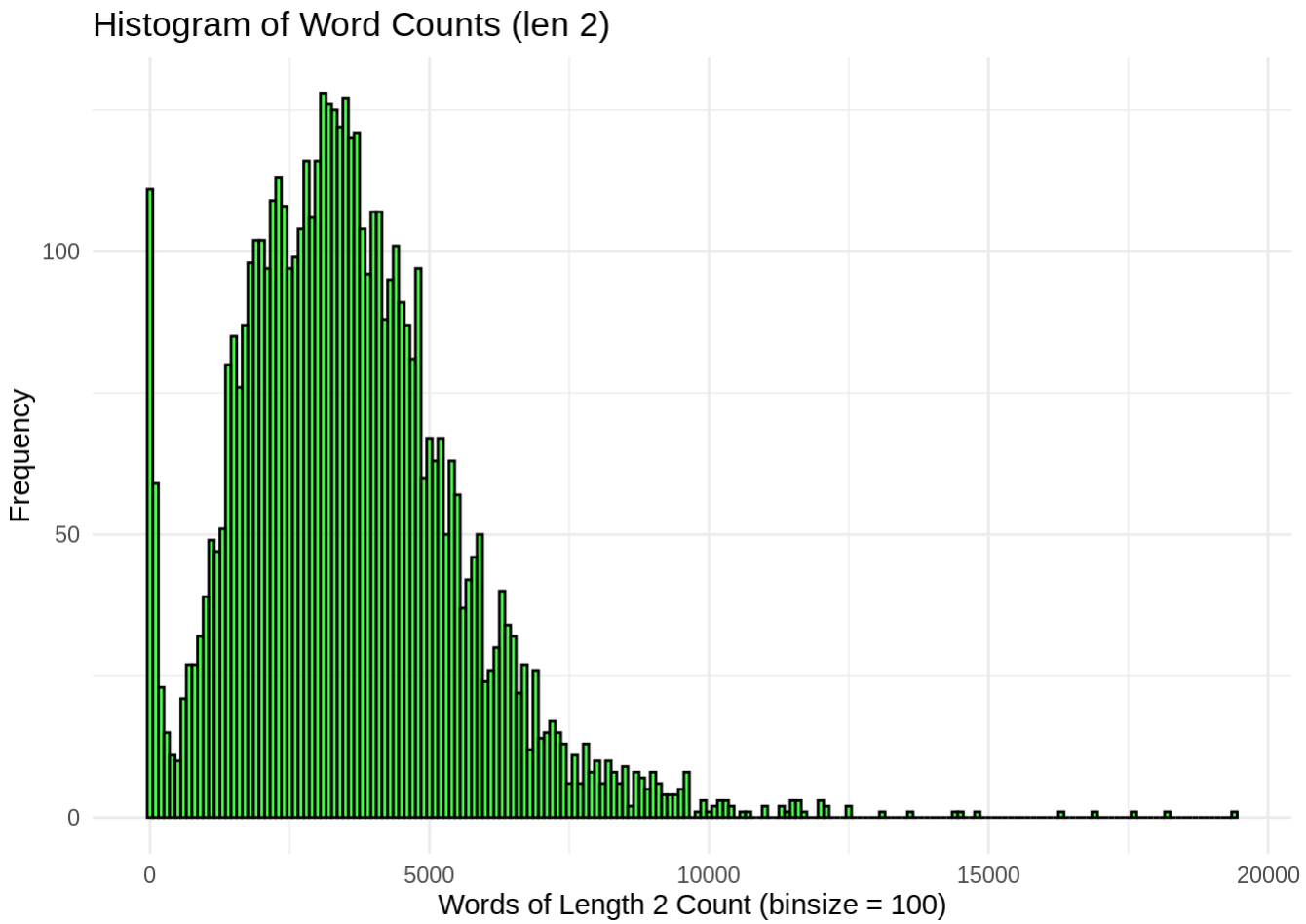
```
ggplot(isdata.noNA, aes(x = len1)) +
  geom_histogram(binwidth = 200, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Page Counts",
       x = "Page Count (binsize = 200)",
       y = "Frequency") +
  theme_minimal()
```



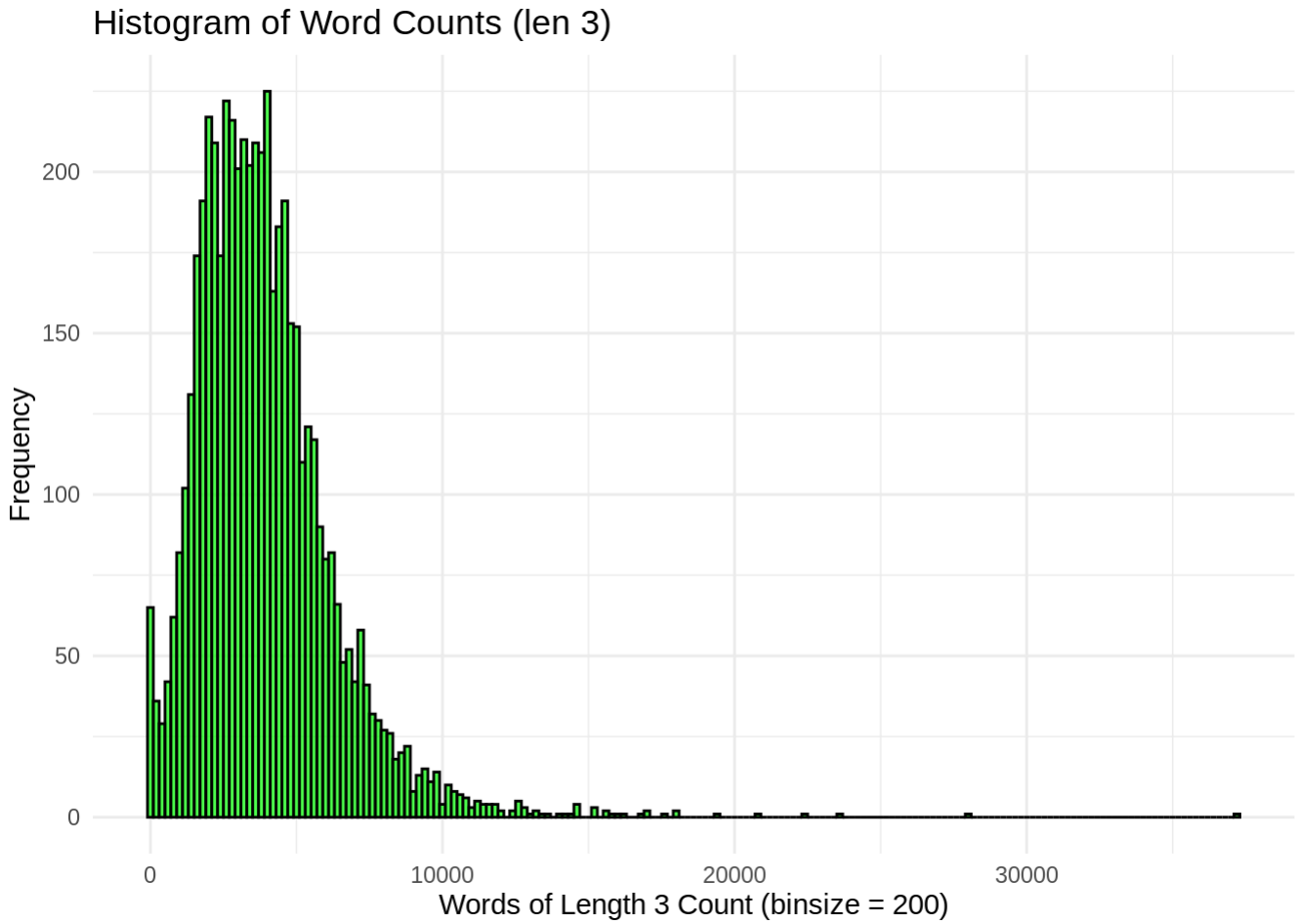

```
# drop entries with more than 20,000 characters of length 1
is.2 <- isdata.noNA[isdata.noNA$len1 < 20000 & isdata.noNA$len1 > 0,]
#is.2

is.rem <- isdata.noNA[isdata.noNA$len1 > 20000 | isdata.noNA$len1 == 0,]
#is.rem

ggplot(is.2, aes(x = len1)) +
  geom_histogram(binwidth = 200, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram Word Counts (len 1)",
       x = "Words of Length 1 Count (binsize = 200)",
       y = "Frequency") +
  theme_minimal()
```



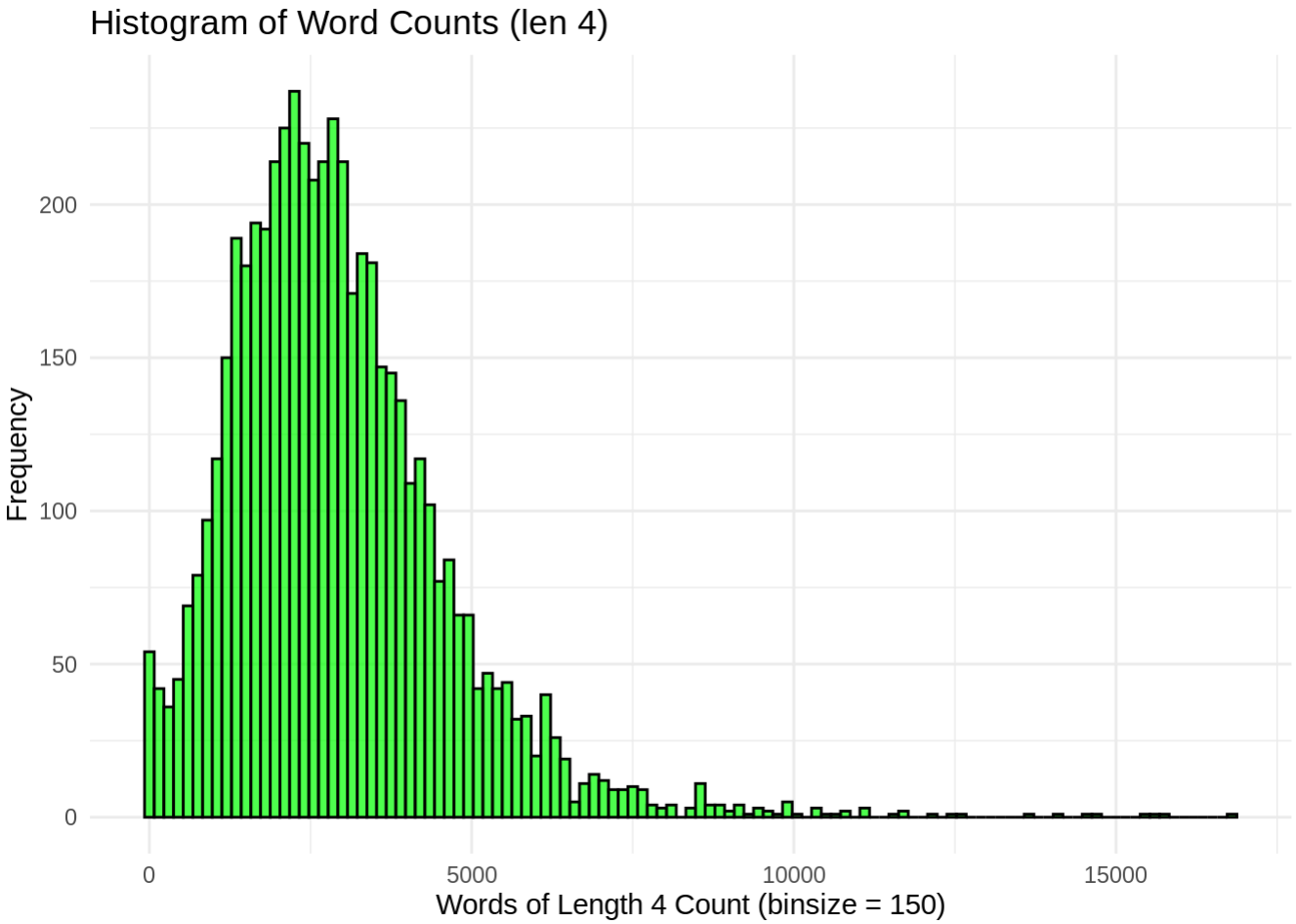

```
ggplot(is.2, aes(x = len2)) +
  geom_histogram(binwidth = 100, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 2)",
       x = "Words of Length 2 Count (binsize = 100)",
       y = "Frequency") +
  theme_minimal()
```



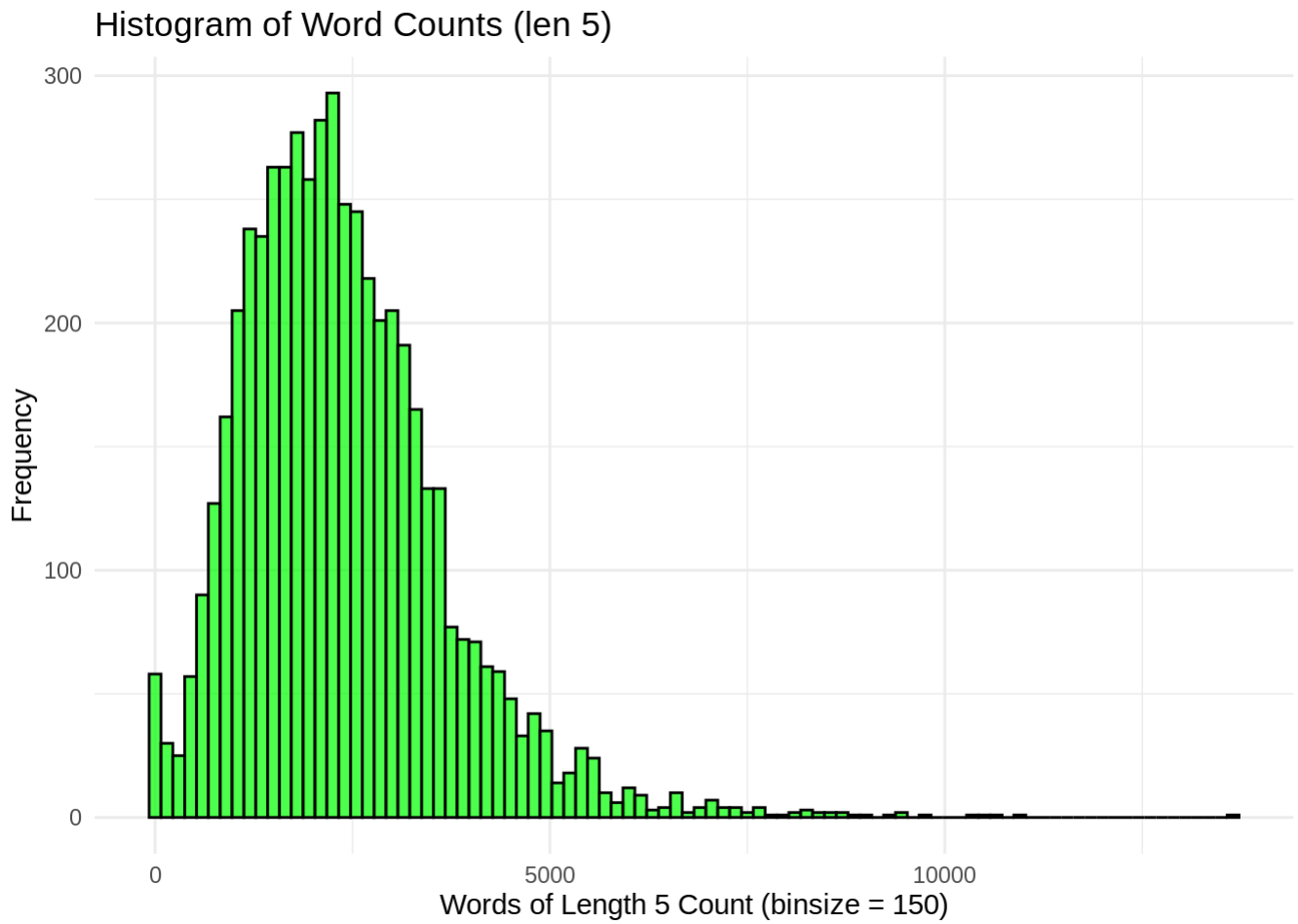

```
ggplot(is.2, aes(x = len3)) +
  geom_histogram(binwidth = 200, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 3)",
       x = "Words of Length 3 Count (binsize = 200)",
       y = "Frequency") +
  theme_minimal()
```



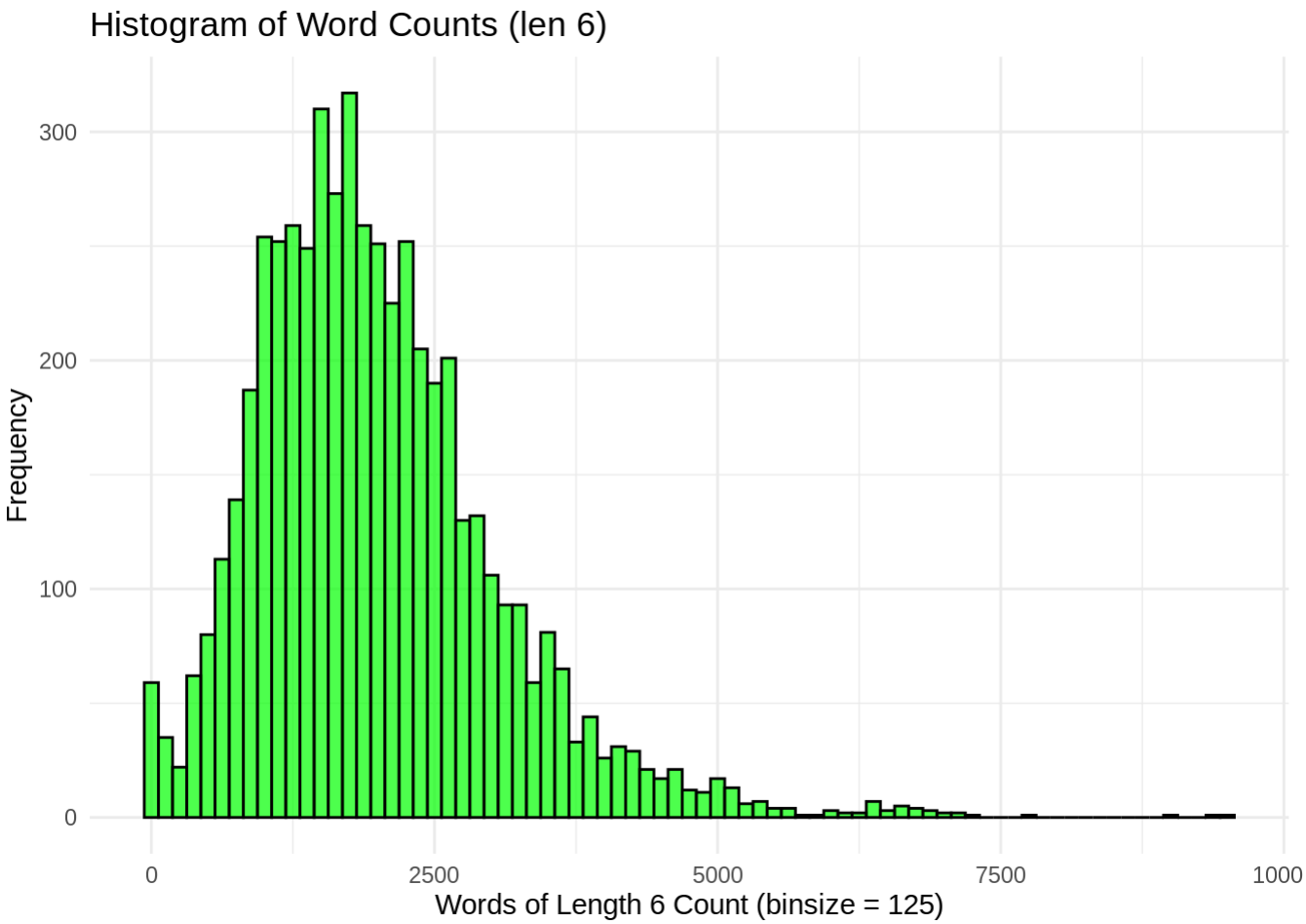

```
ggplot(is.2, aes(x = len4)) +
  geom_histogram(binwidth = 150, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 4)",
       x = "Words of Length 4 Count (binsize = 150)",
       y = "Frequency") +
  theme_minimal()
```



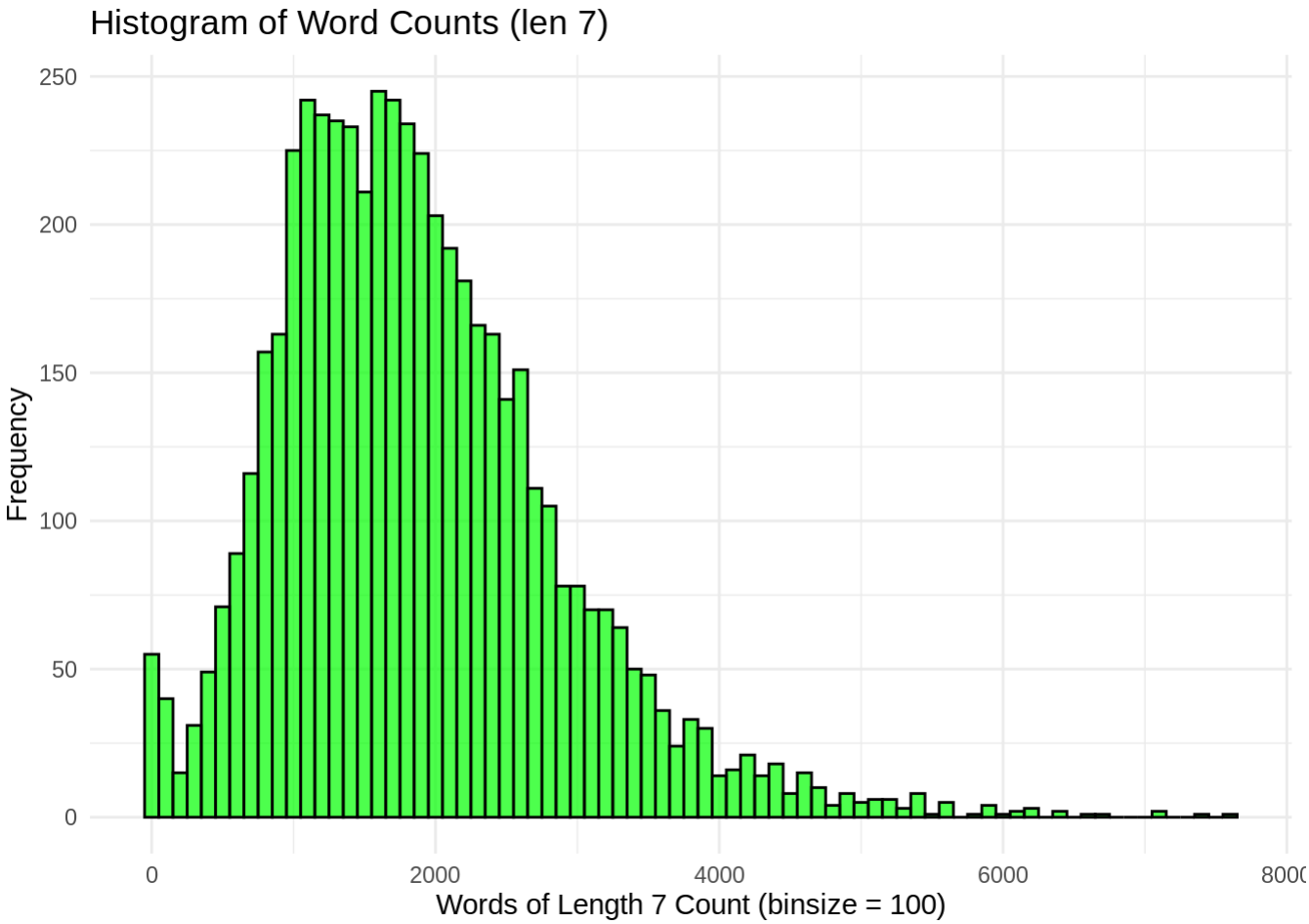

```
ggplot(is.2, aes(x = len5)) +
  geom_histogram(binwidth = 150, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 5)",
       x = "Words of Length 5 Count (binsize = 150)",
       y = "Frequency") +
  theme_minimal()
```



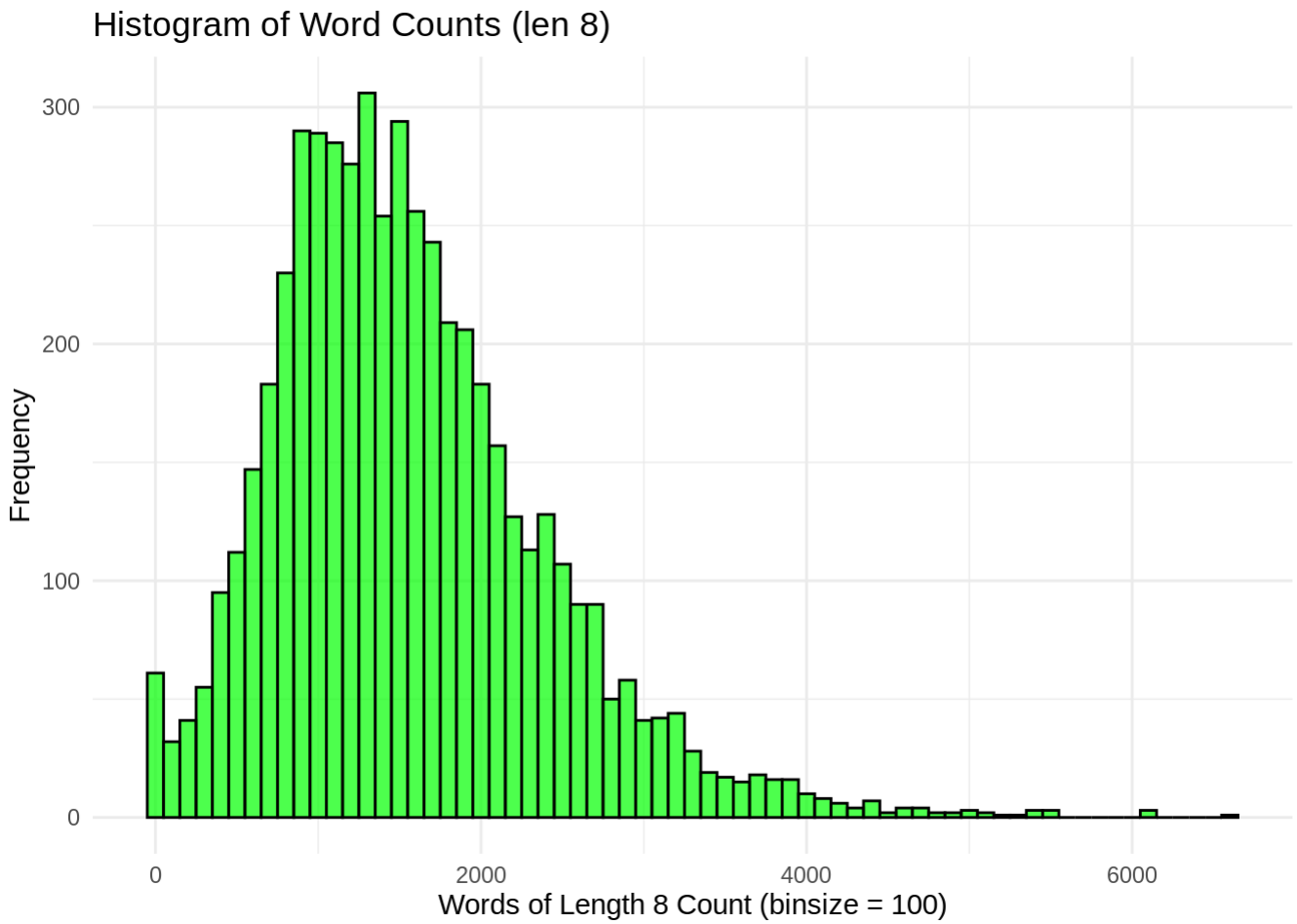

```
ggplot(is.2, aes(x = len6)) +
  geom_histogram(binwidth = 125, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 6)",
       x = "Words of Length 6 Count (binsize = 125)",
       y = "Frequency") +
  theme_minimal()
```



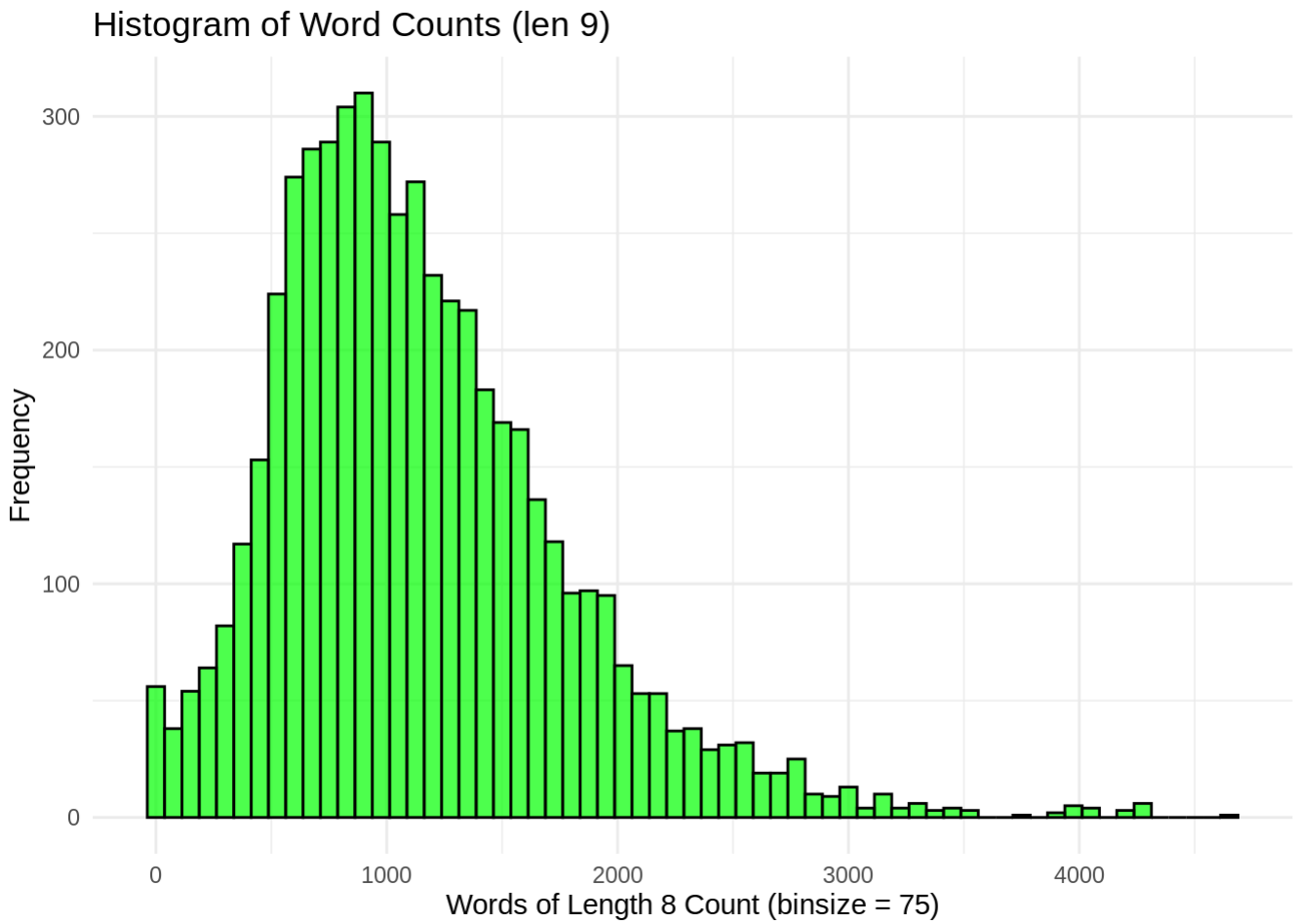

```
ggplot(is.2, aes(x = len7)) +
  geom_histogram(binwidth = 100, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 7)",
       x = "Words of Length 7 Count (binsize = 100)",
       y = "Frequency") +
  theme_minimal()
```



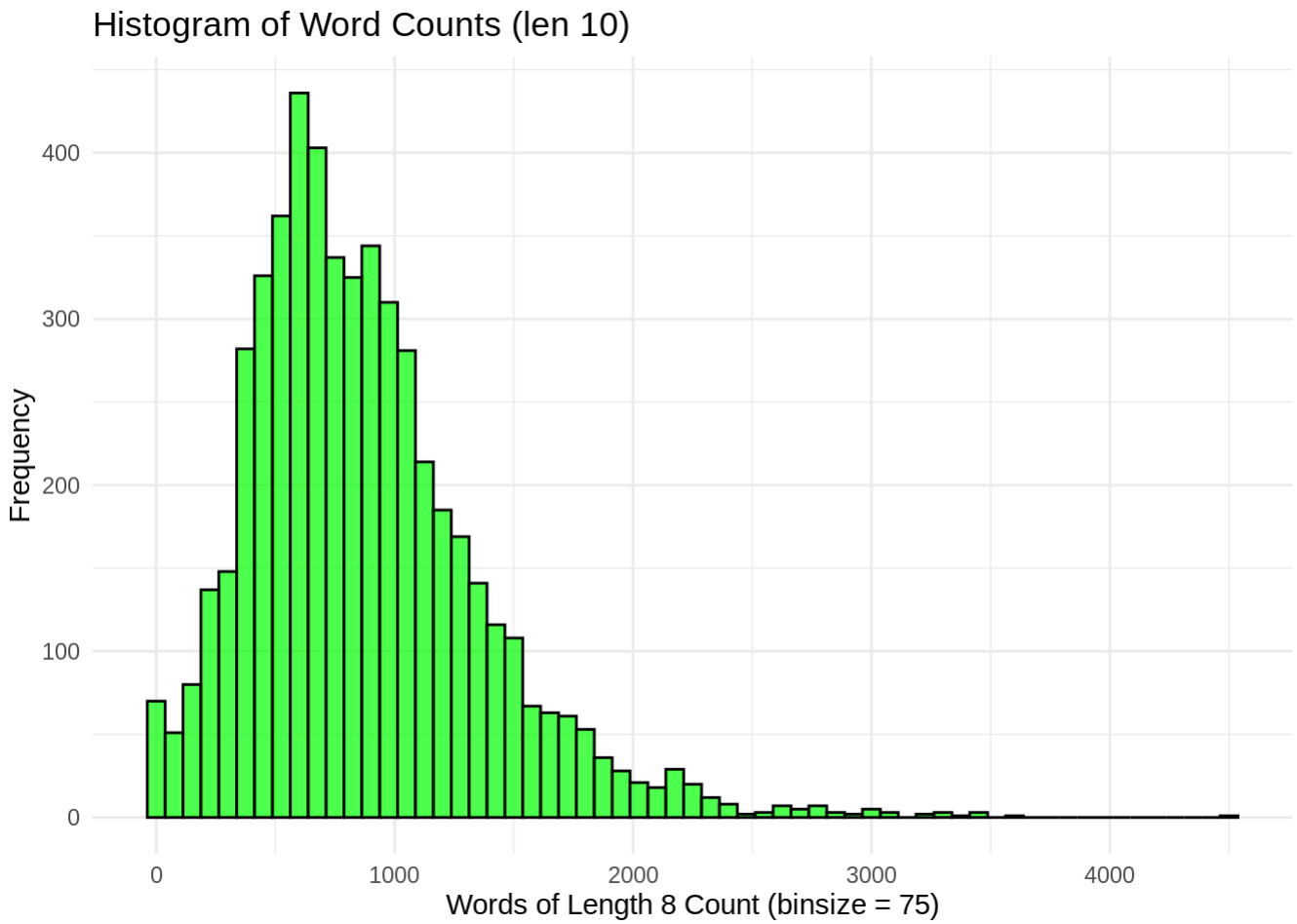

```
ggplot(is.2, aes(x = len8)) +
  geom_histogram(binwidth = 100, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 8)",
       x = "Words of Length 8 Count (binsize = 100)",
       y = "Frequency") +
  theme_minimal()
```



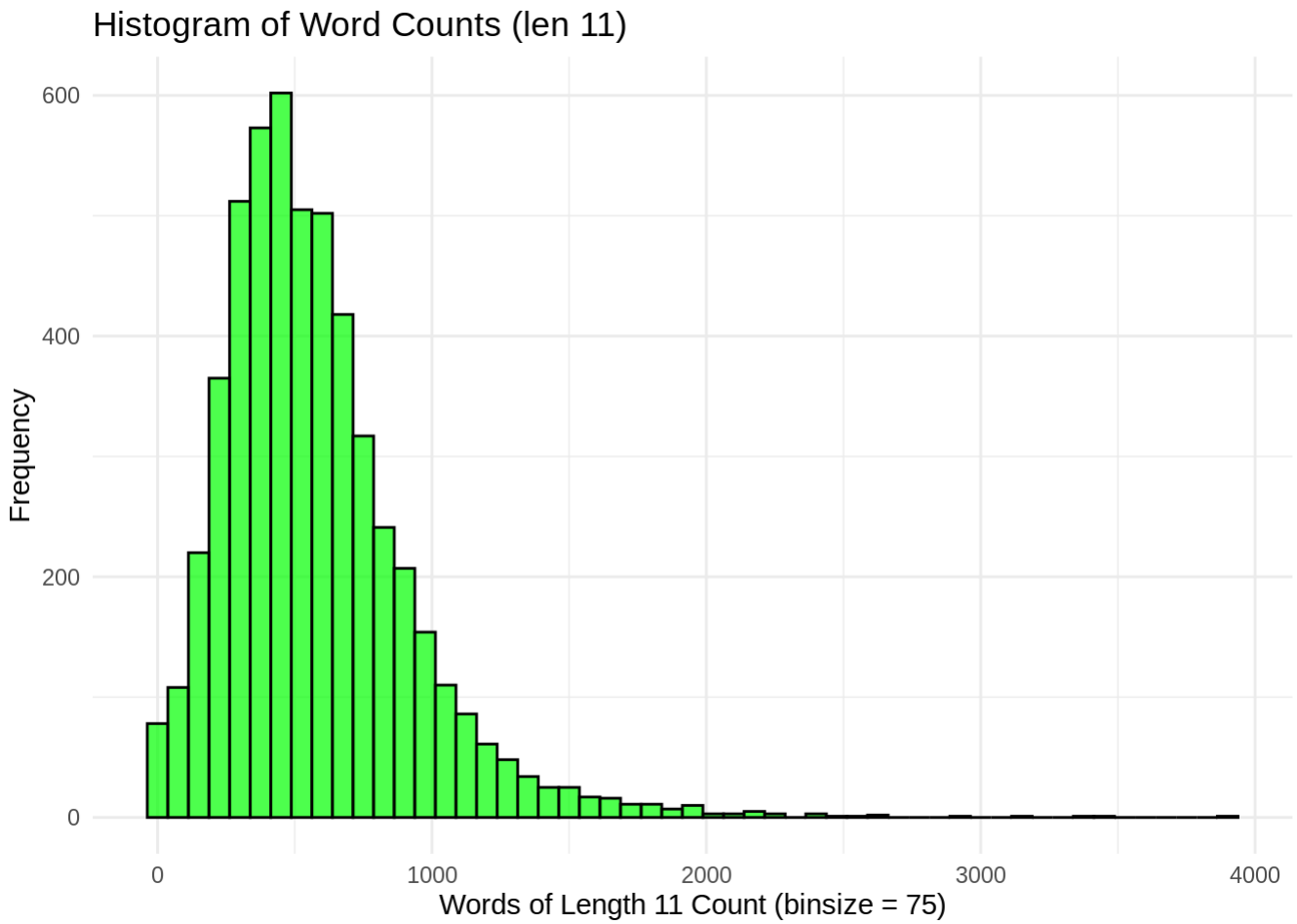

```
ggplot(is.2, aes(x = len9)) +
  geom_histogram(binwidth = 75, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 9)",
       x = "Words of Length 8 Count (binsize = 75)",
       y = "Frequency") +
  theme_minimal()
```



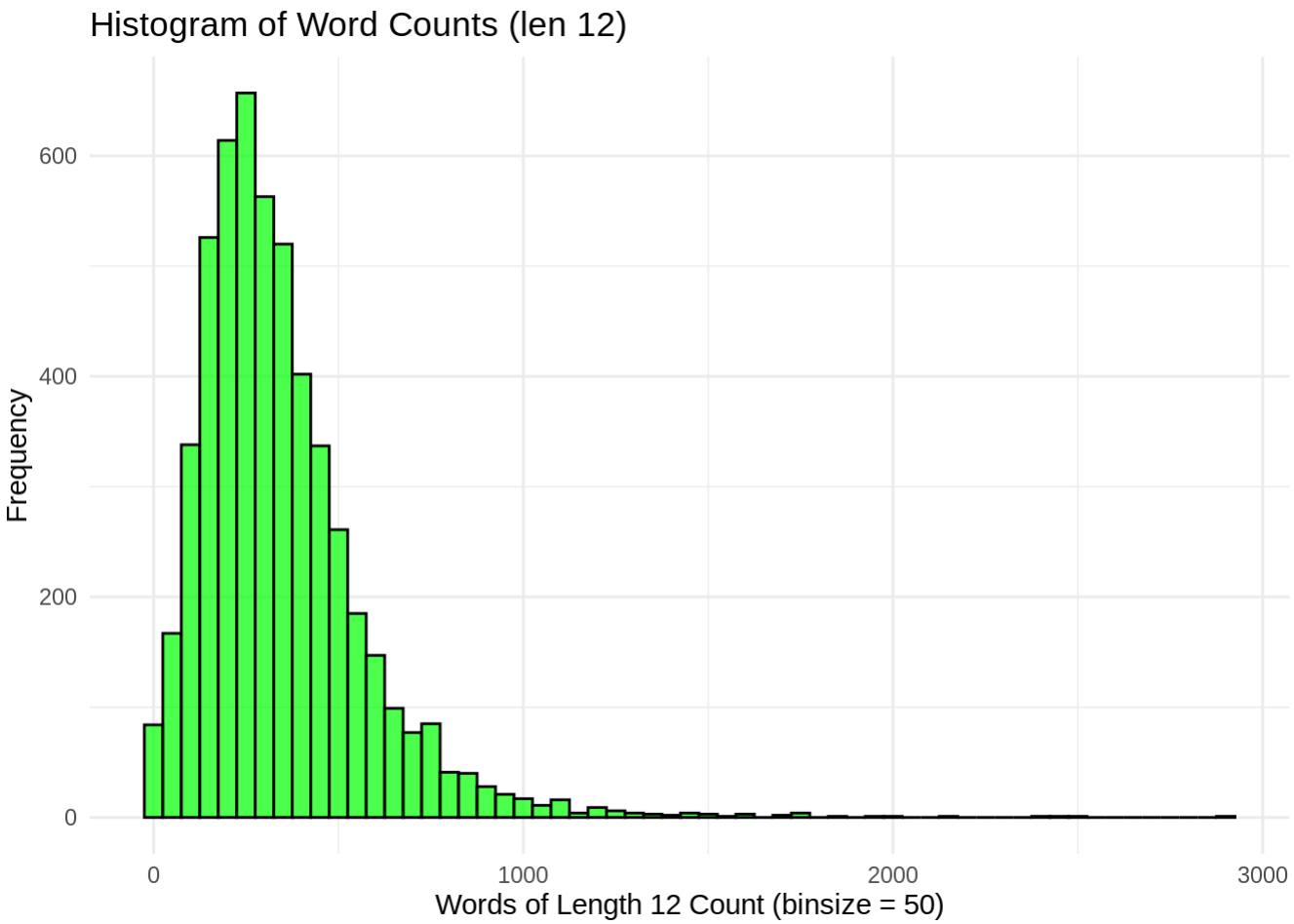

```
ggplot(is.2, aes(x = len10)) +
  geom_histogram(binwidth = 75, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 10)",
       x = "Words of Length 8 Count (binsize = 75)",
       y = "Frequency") +
  theme_minimal()
```



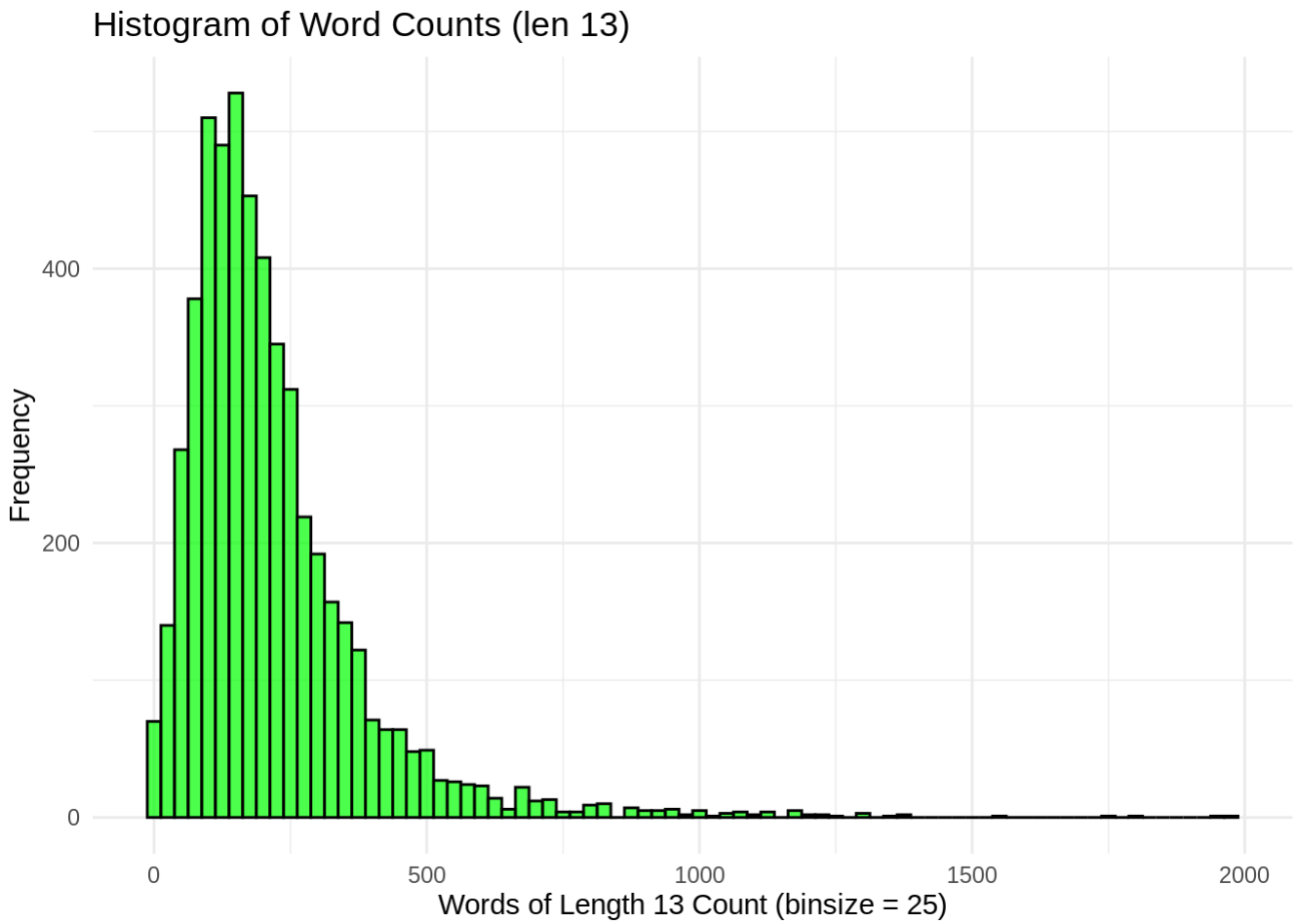

```
ggplot(is.2, aes(x = len11)) +
  geom_histogram(binwidth = 75, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 11)",
       x = "Words of Length 11 Count (binsize = 75)",
       y = "Frequency") +
  theme_minimal()
```



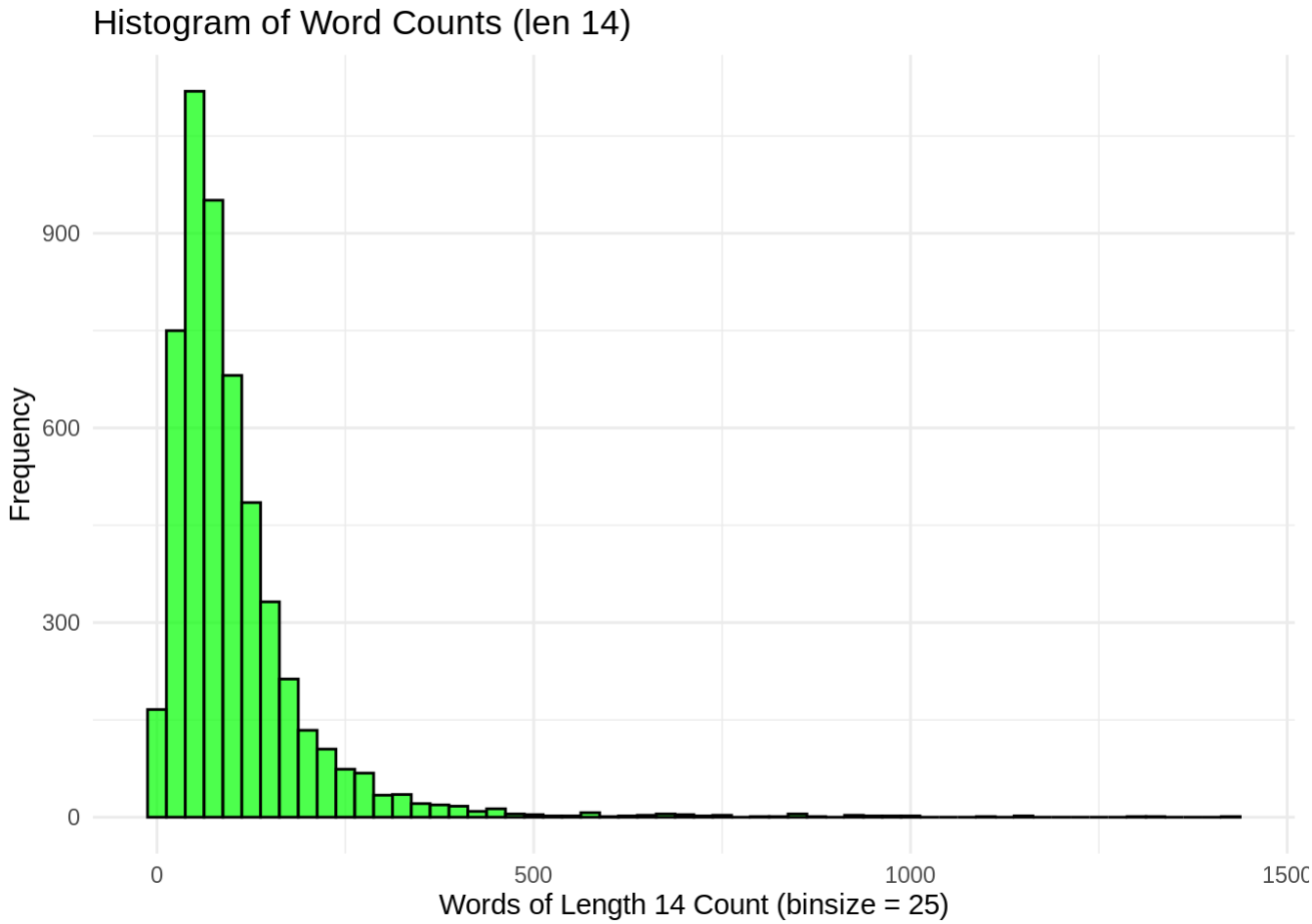

```
ggplot(is.2, aes(x = len12)) +
  geom_histogram(binwidth = 50, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 12)",
       x = "Words of Length 12 Count (binsize = 50)",
       y = "Frequency") +
  theme_minimal()
```



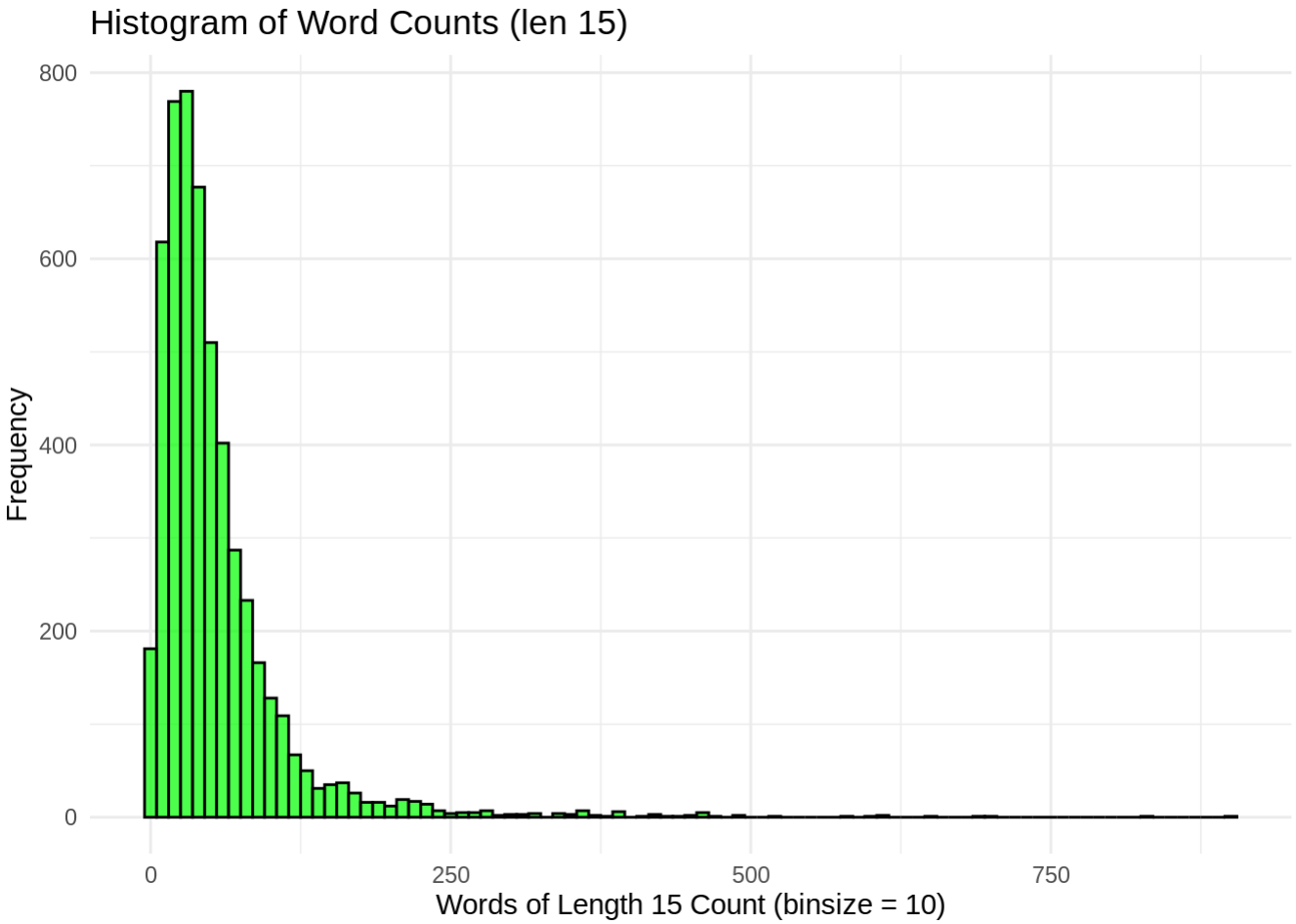

```
ggplot(is.2, aes(x = len13)) +
  geom_histogram(binwidth = 25, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 13)",
       x = "Words of Length 13 Count (binsize = 25)",
       y = "Frequency") +
  theme_minimal()
```




```
ggplot(is.2, aes(x = len14)) +
  geom_histogram(binwidth = 25, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 14)",
       x = "Words of Length 14 Count (binsize = 25)",
       y = "Frequency") +
  theme_minimal()
```




```
ggplot(is.2, aes(x = len15)) +
  geom_histogram(binwidth = 10, fill = "green", color = "black", alpha = 0.7) +
  labs(title = "Histogram of Word Counts (len 15)",
       x = "Words of Length 15 Count (binsize = 10)",
       y = "Frequency") +
  theme_minimal()
```



Token Statistics

```
favstats(is.2$typeTokenRatioVal)
```

```
##        min    Q1    median        Q3       max      mean         sd    n
##  0.2073614 0.373 0.3985455 0.4220833 0.6923333 0.3971423 0.03735019 5289
##  missing
##        0
```

```
favstats(is.2$avgToksSentVal)
```

```
##       min       Q1  median       Q3      max     mean      sd    n missing
##  1.575359 22.83813 25.0309 27.52108 446.8962 25.48352 8.79816 5289       0
```

```
favstats(is.2$avgTokLenVal)
```

```
##       min       Q1   median   Q3     max     mean        sd    n missing
##  2.288667 4.286226 4.490944 4.67 19.5864 4.500096 0.6880502 5289       0
```

```
favstats(is.2$wordc)
```

```
##  min    Q1 median    Q3    max     mean       sd    n missing
##    6 16375  24118 32852 186638 25980.94 14020.06 5289       0
```

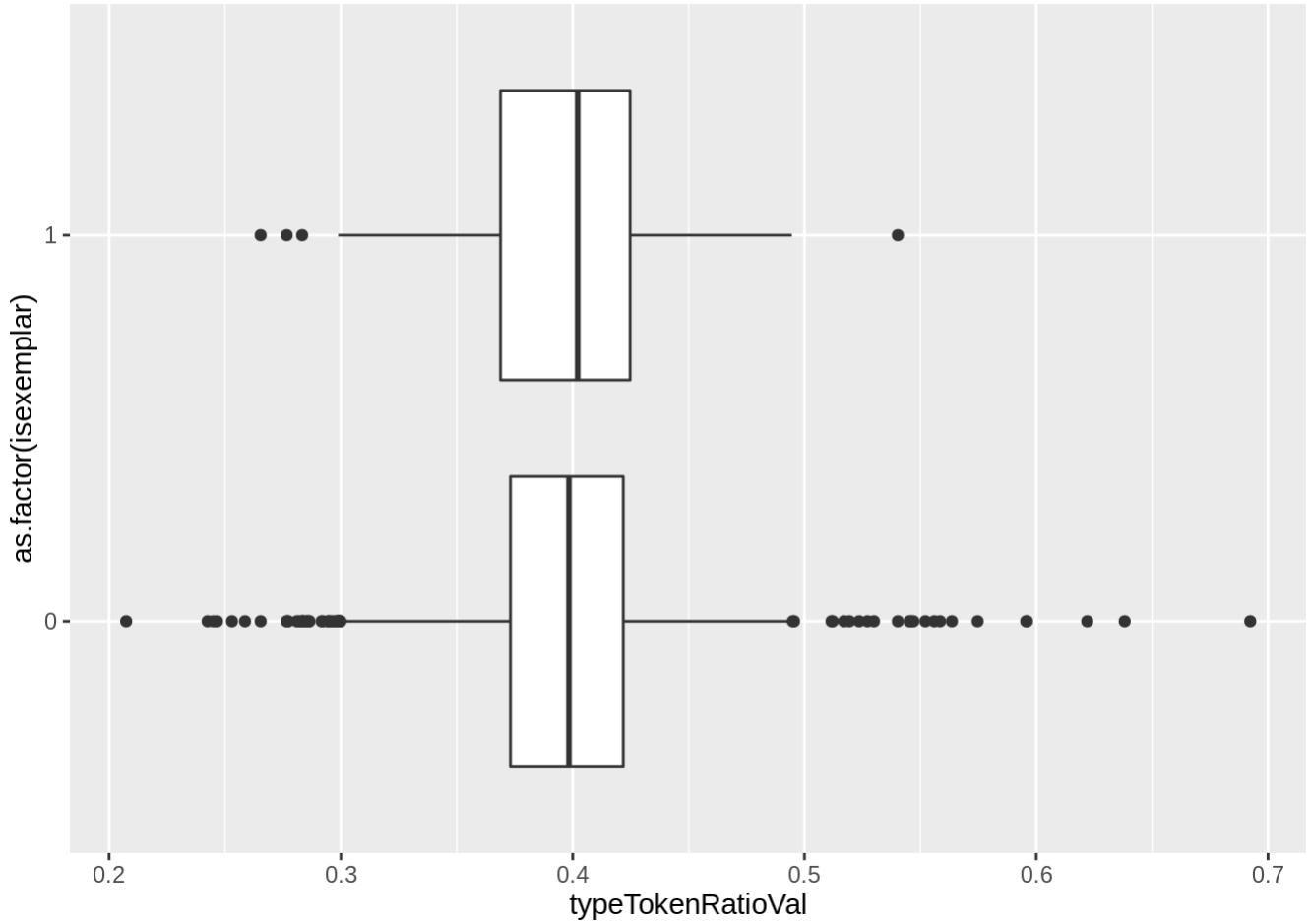
```
favstats(is.2$punctPerSent)
```

```
##         min        Q1    median        Q3       max      mean        sd    n
##  0.06214628 0.1119681 0.1266181 0.1463077 0.4970587 0.1328789 0.0337806 5289
##  missing
##        0
```

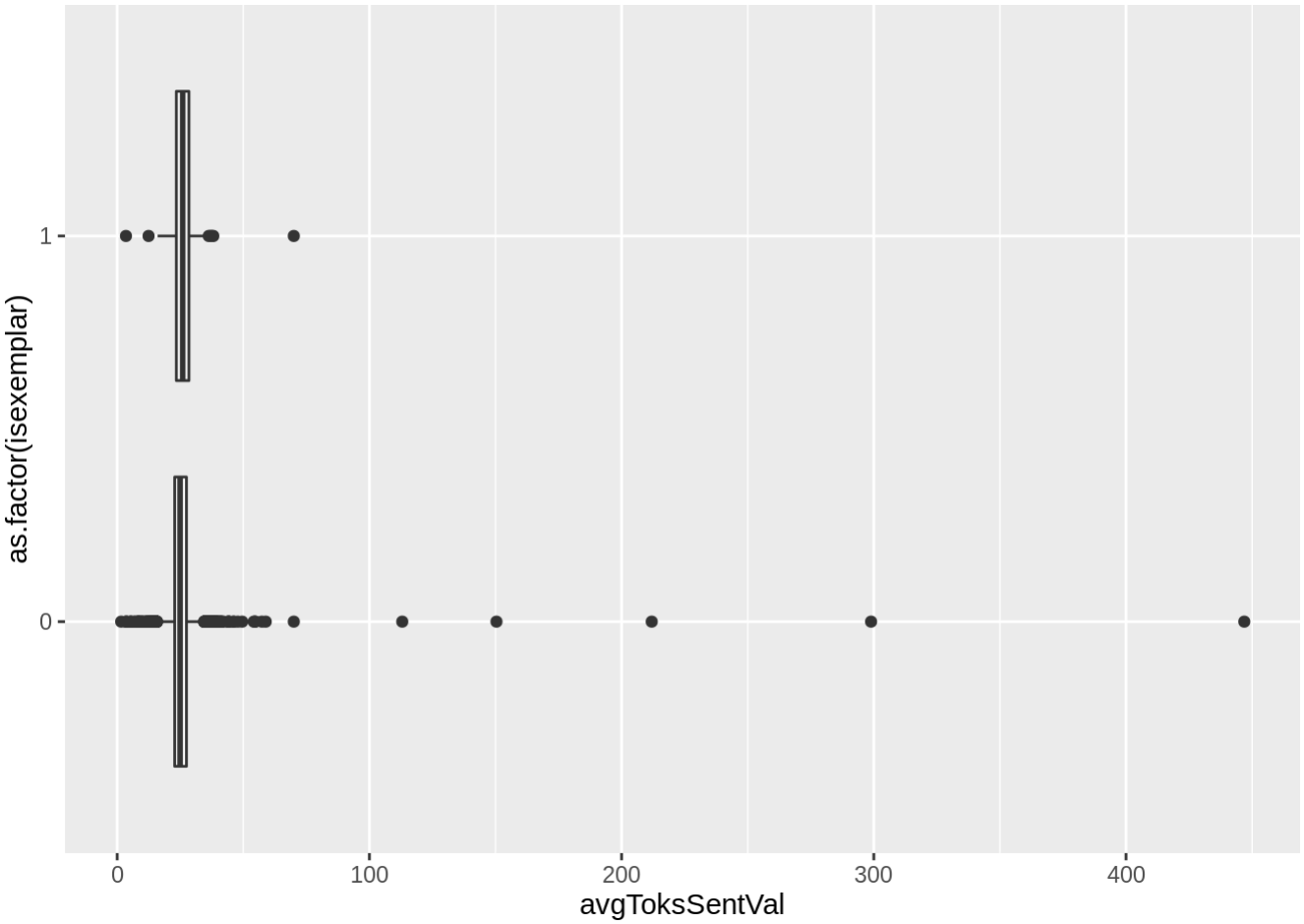
```
favstats(is.2$punctPerTok)
```

```
##         min        Q1    median        Q3       max      mean        sd    n
##  0.06214628 0.1119681 0.1266181 0.1463077 0.4970587 0.1328789 0.0337806 5289
##  missing
##        0
```

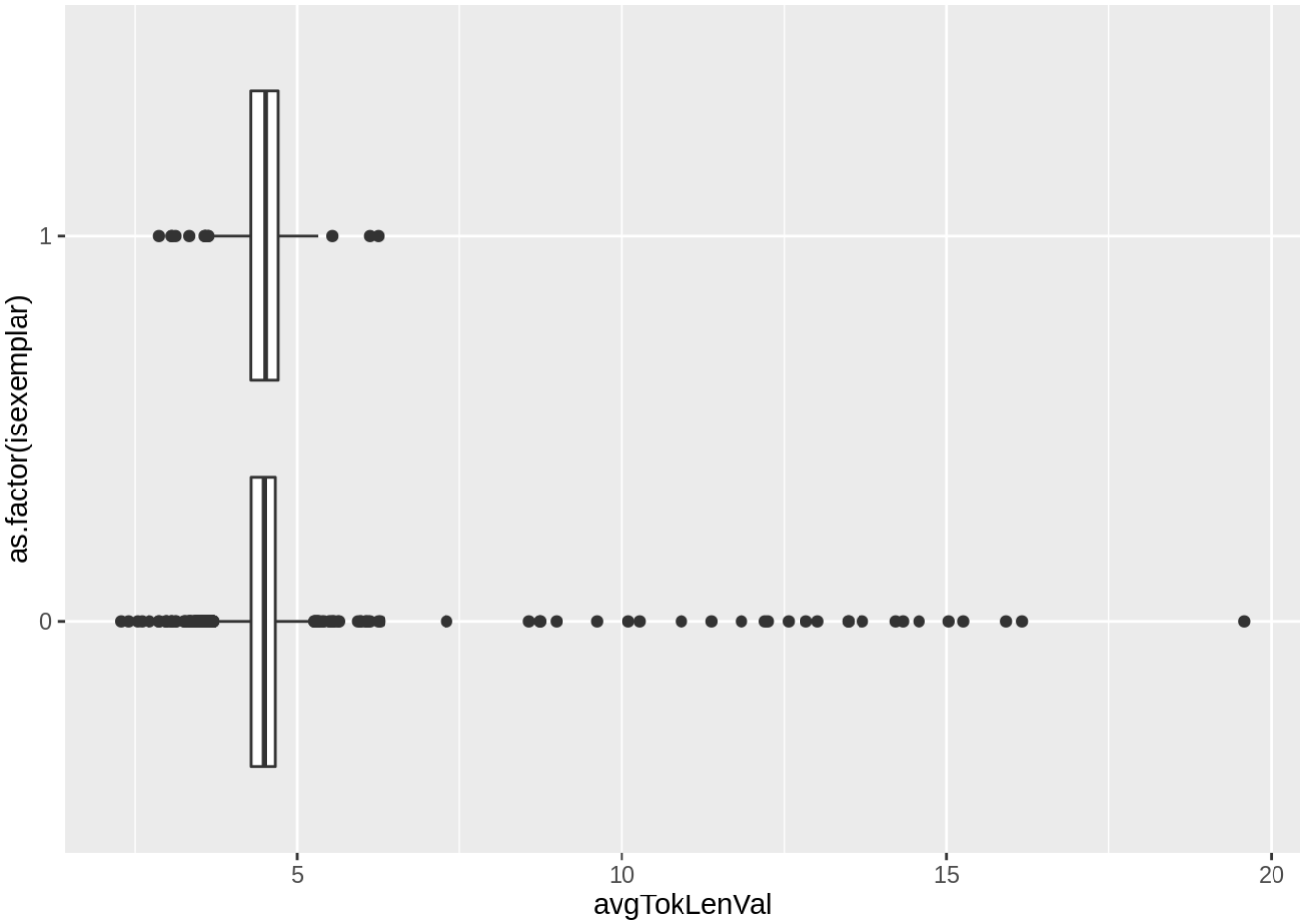
```
gf_boxplot(as.factor(isexemplar) ~ typeTokenRatioVal, data = is.2)
```



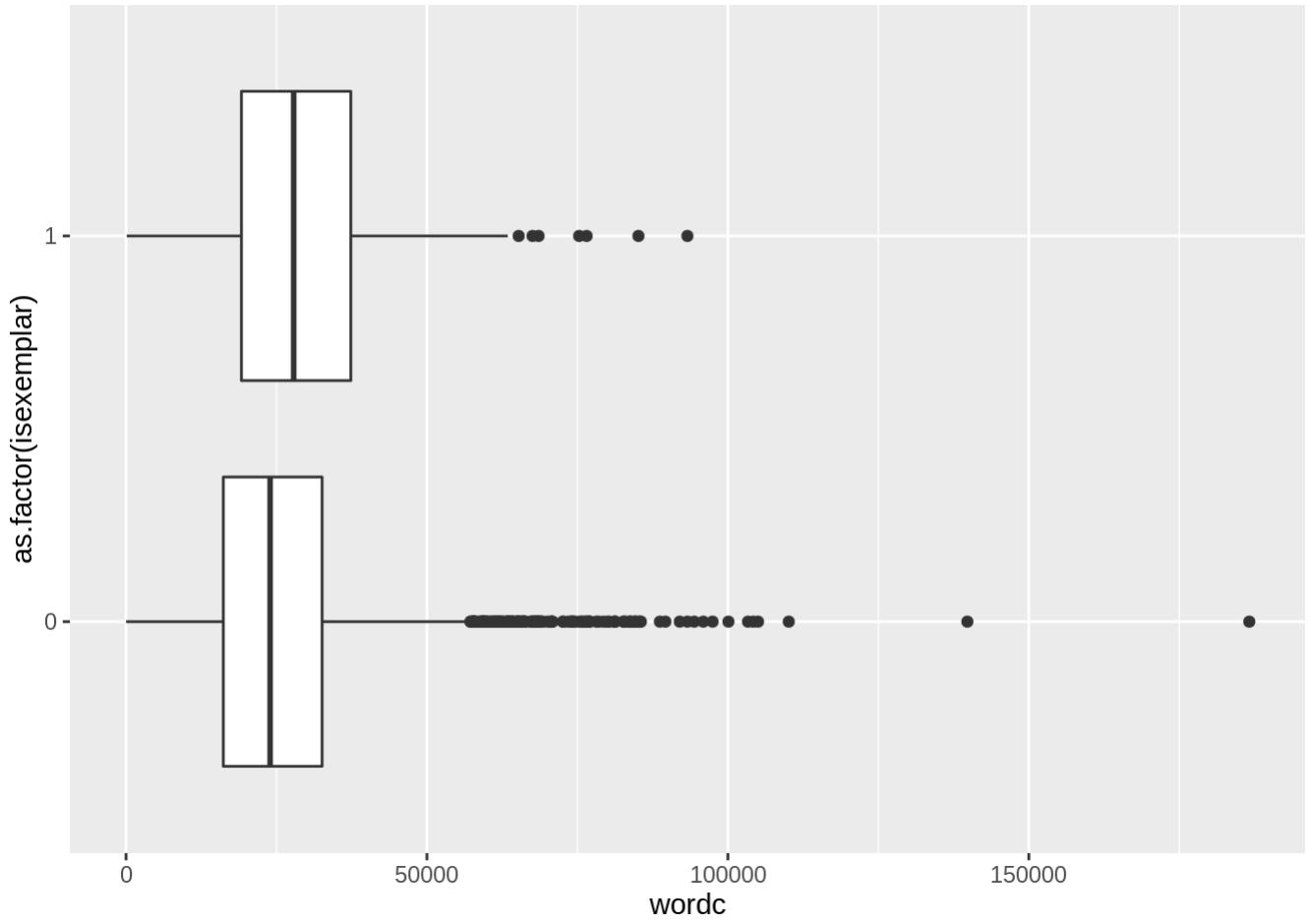

```
gf_boxplot(as.factor(isexemplar) ~ avgToksSentVal, data = is.2)
```



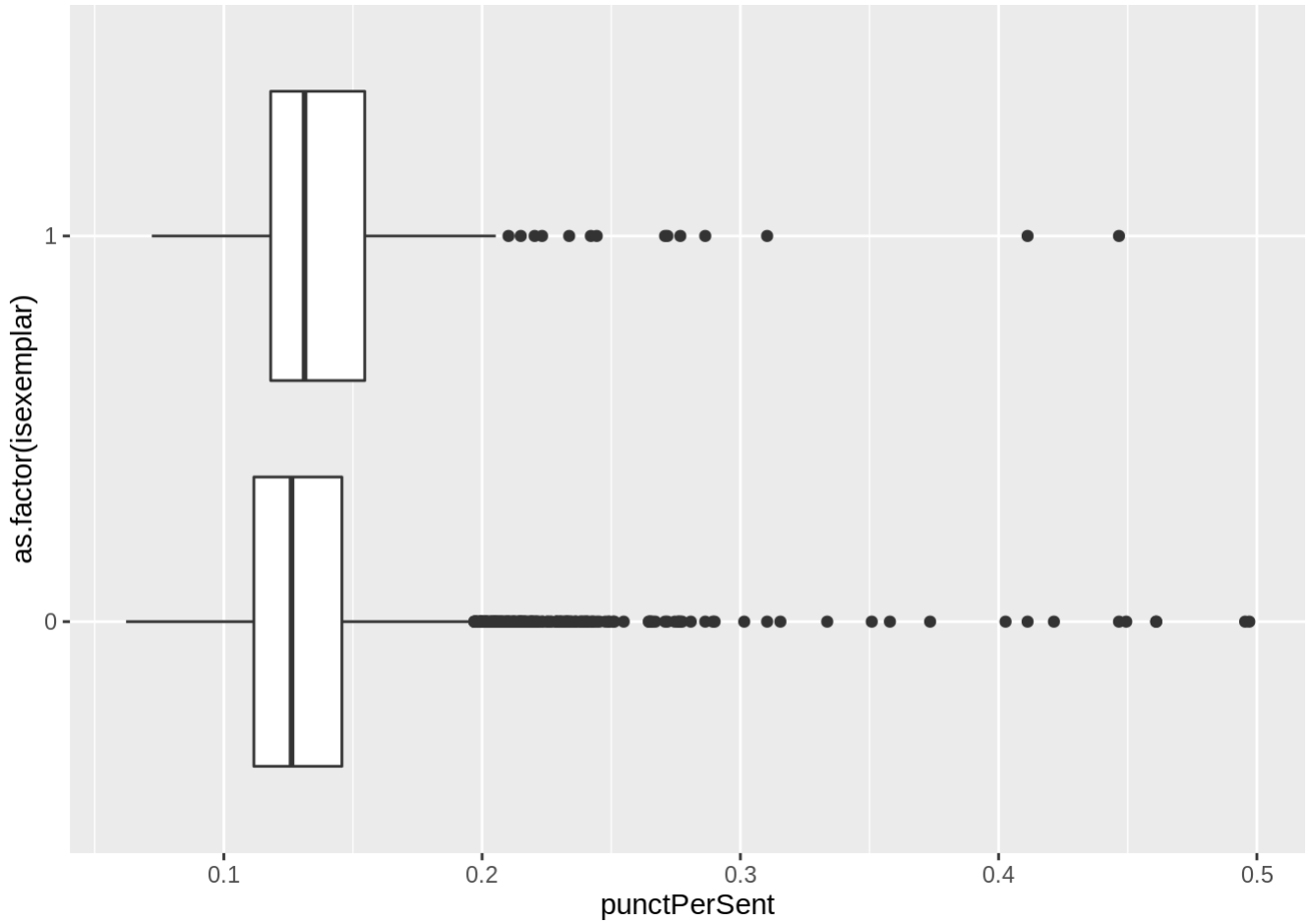

```
gf_boxplot(as.factor(isexemplar) ~ avgTokLenVal, data = is.2)
```



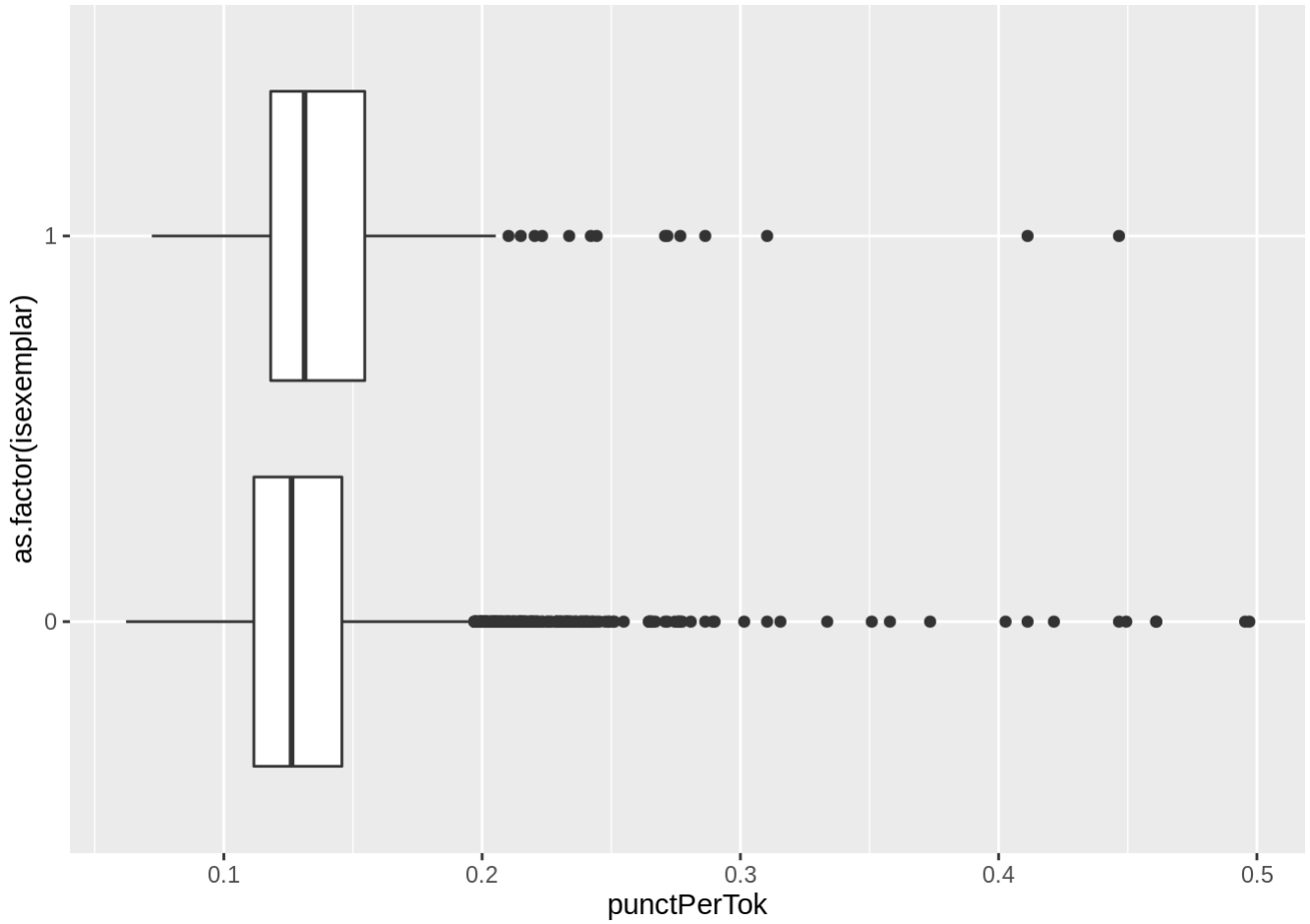

```
gf_boxplot(as.factor(isexemplar) ~ wordc, data = is.2)
```



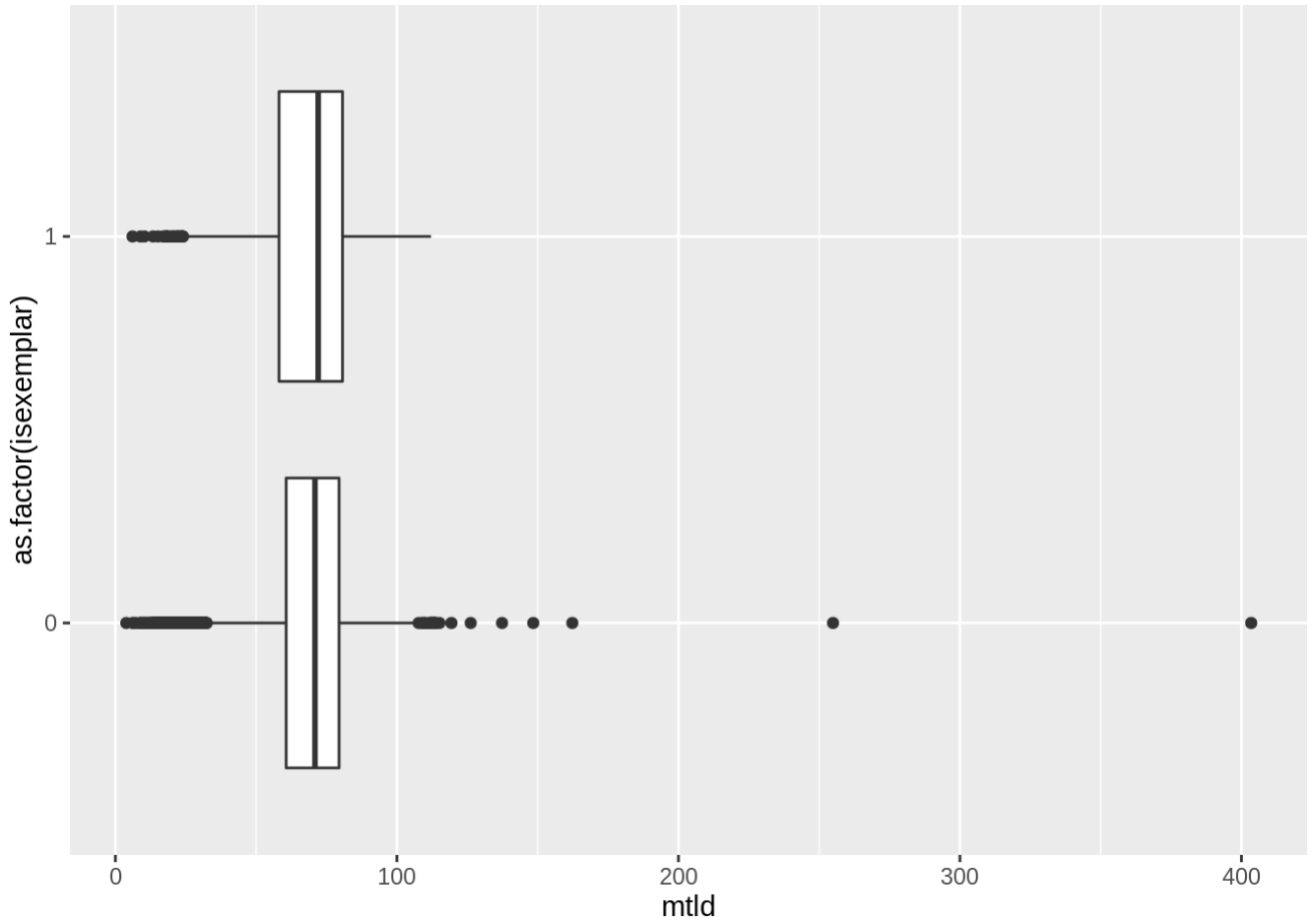

```
gf_boxplot(as.factor(isexemplar) ~ punctPerSent, data = is.2)
```



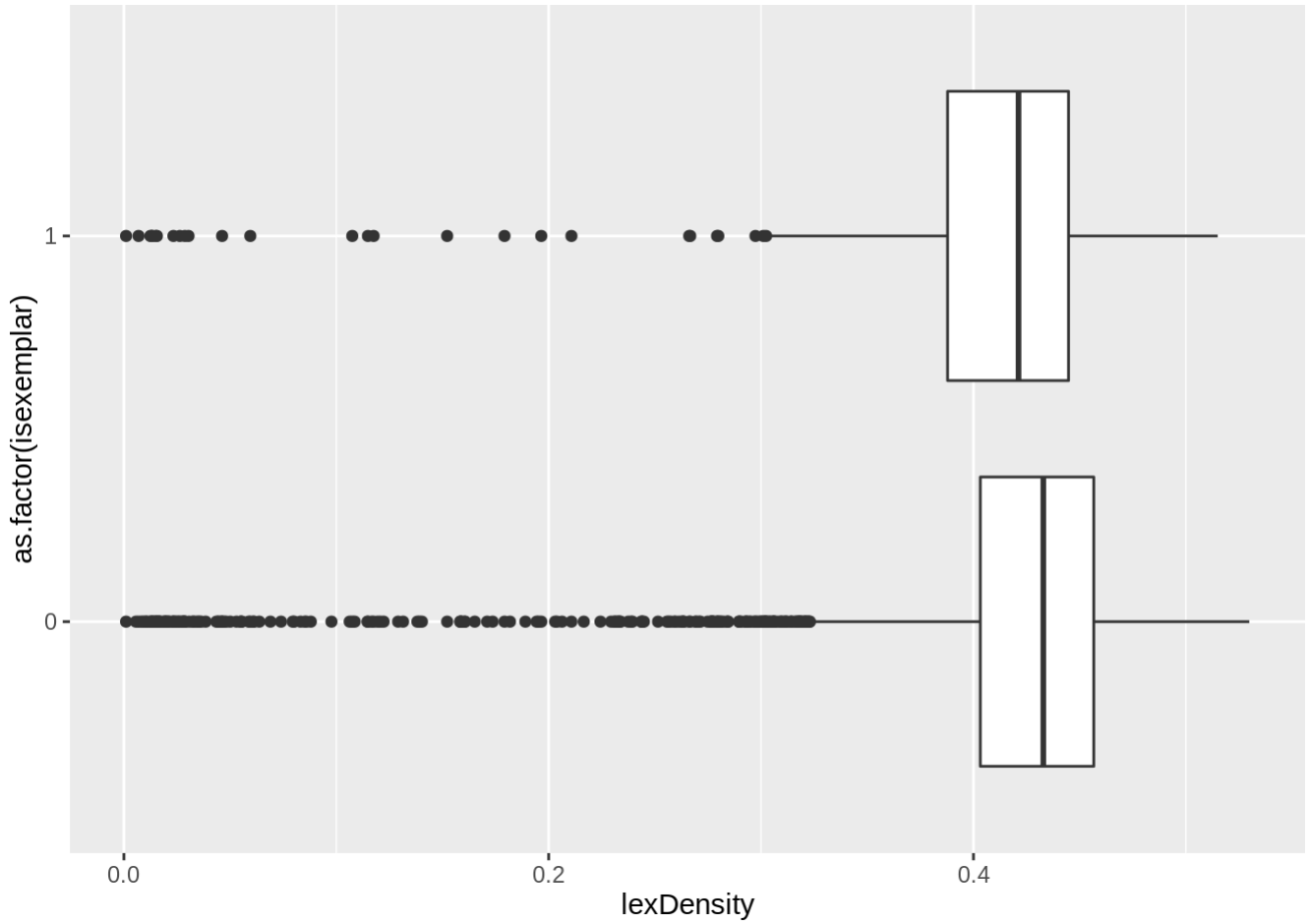

```
gf_boxplot(as.factor(isexemplar) ~ punctPerTok, data = is.2)
```



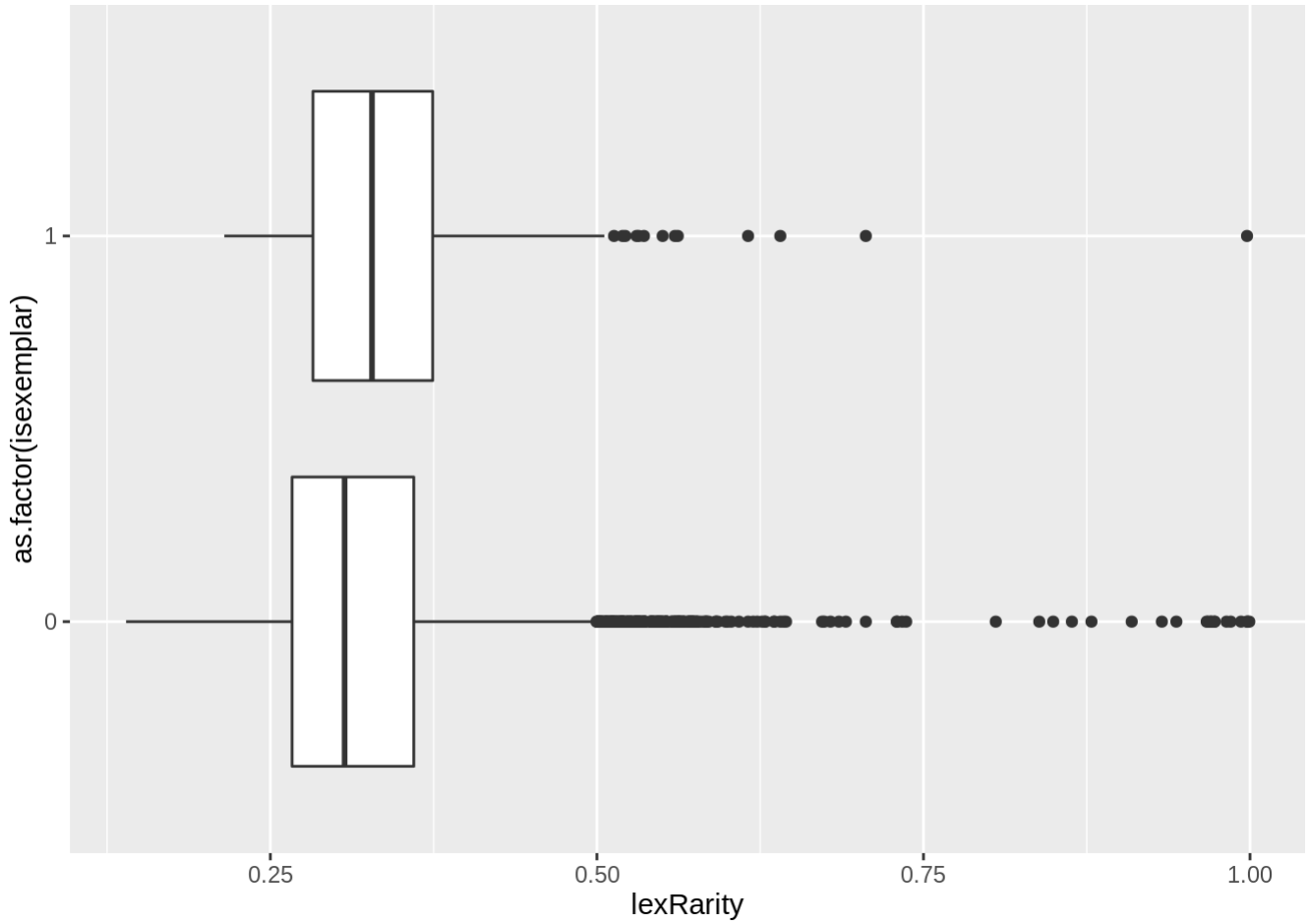

```
gf_boxplot(as.factor(isexemplar) ~ mtld, data = is.2)
```



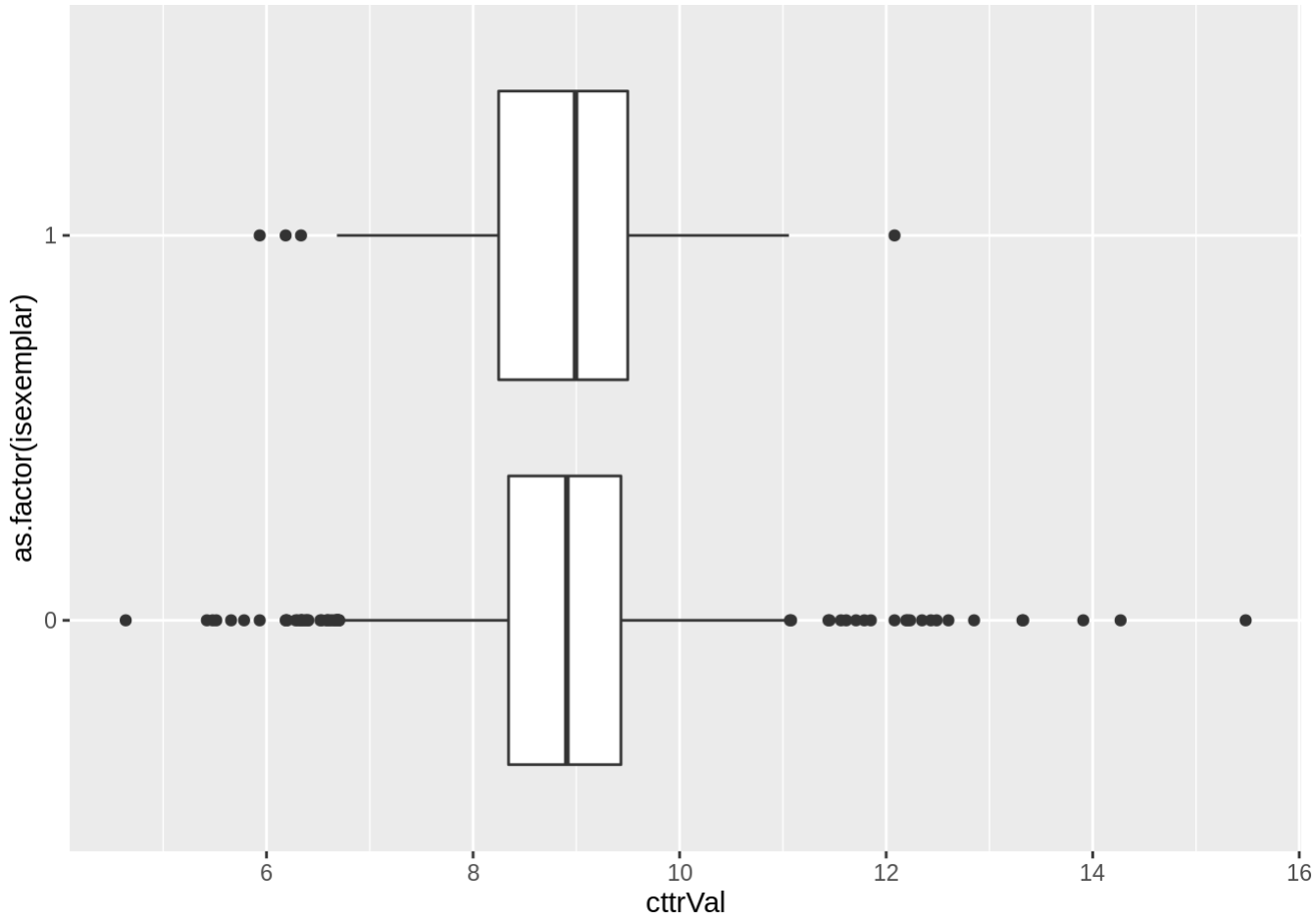

```
gf_boxplot(as.factor(isexemplar) ~ lexDensity, data = is.2) # promising..
```



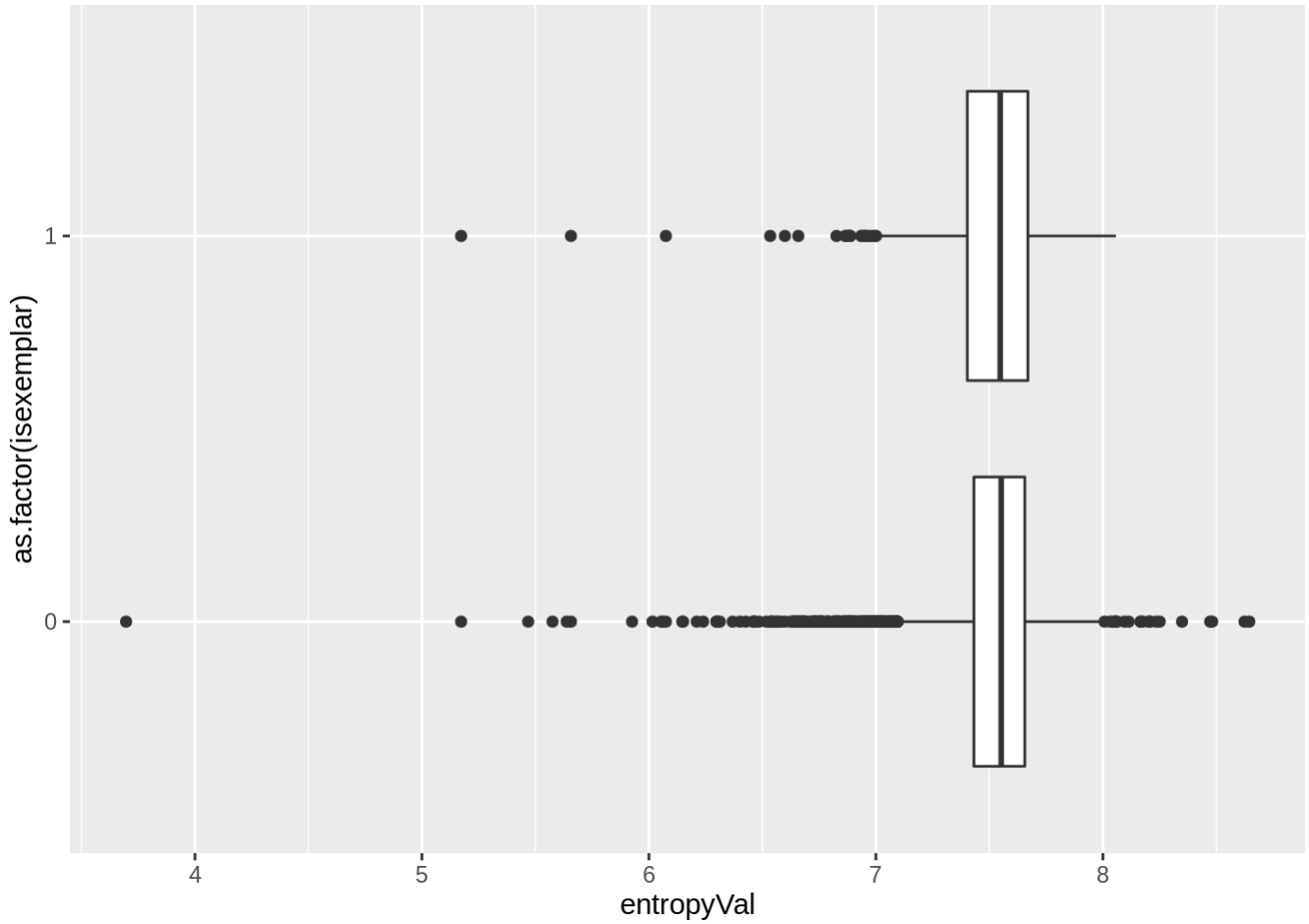

```
gf_boxplot(as.factor(isexemplar) ~ lexRarity, data = is.2) # also promising...?
```



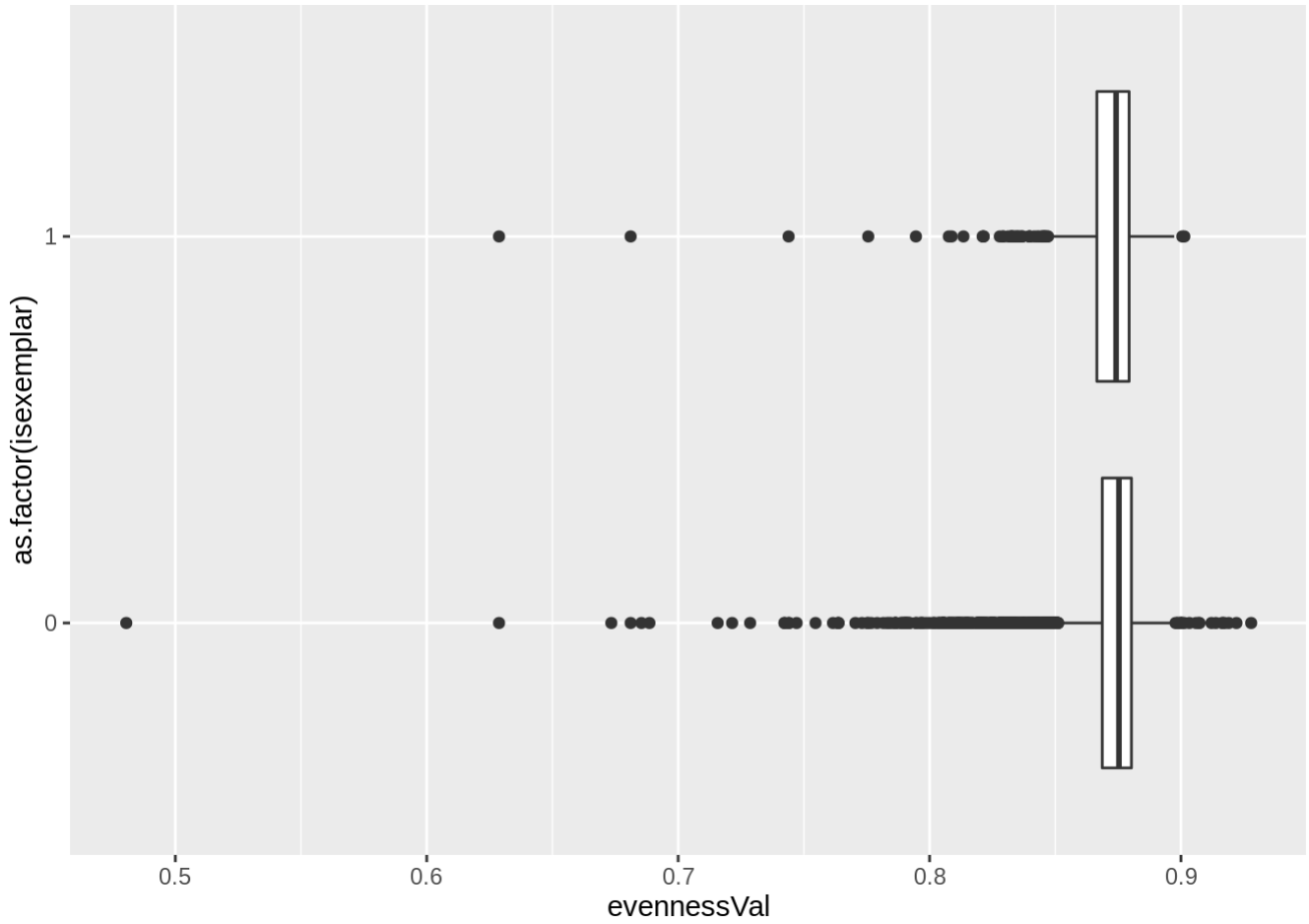

```
gf_boxplot(as.factor(isexemplar) ~ cttrVal, data = is.2)
```



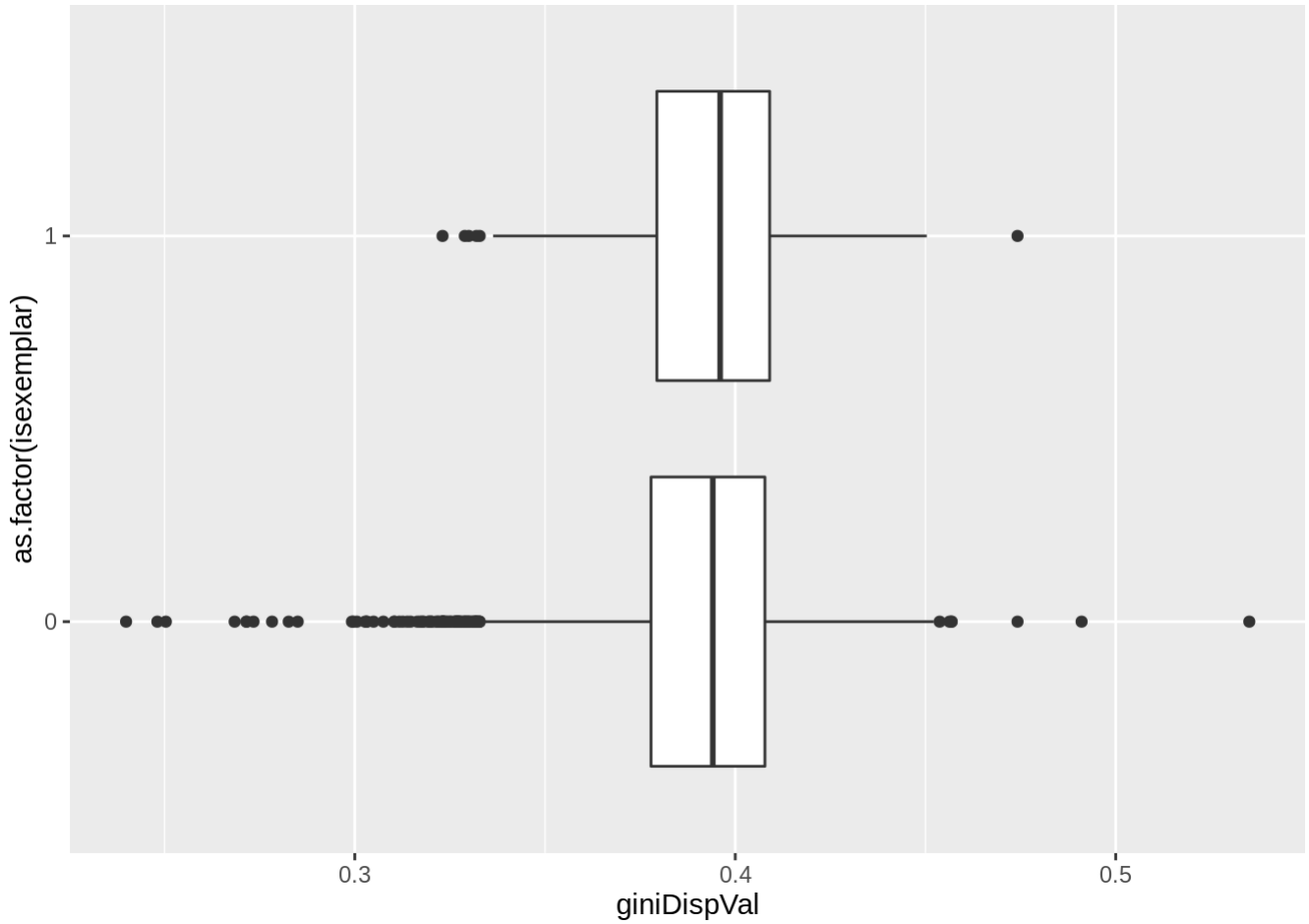

```
gf_boxplot(as.factor(isexemplar) ~ entropyVal, data = is.2)
```



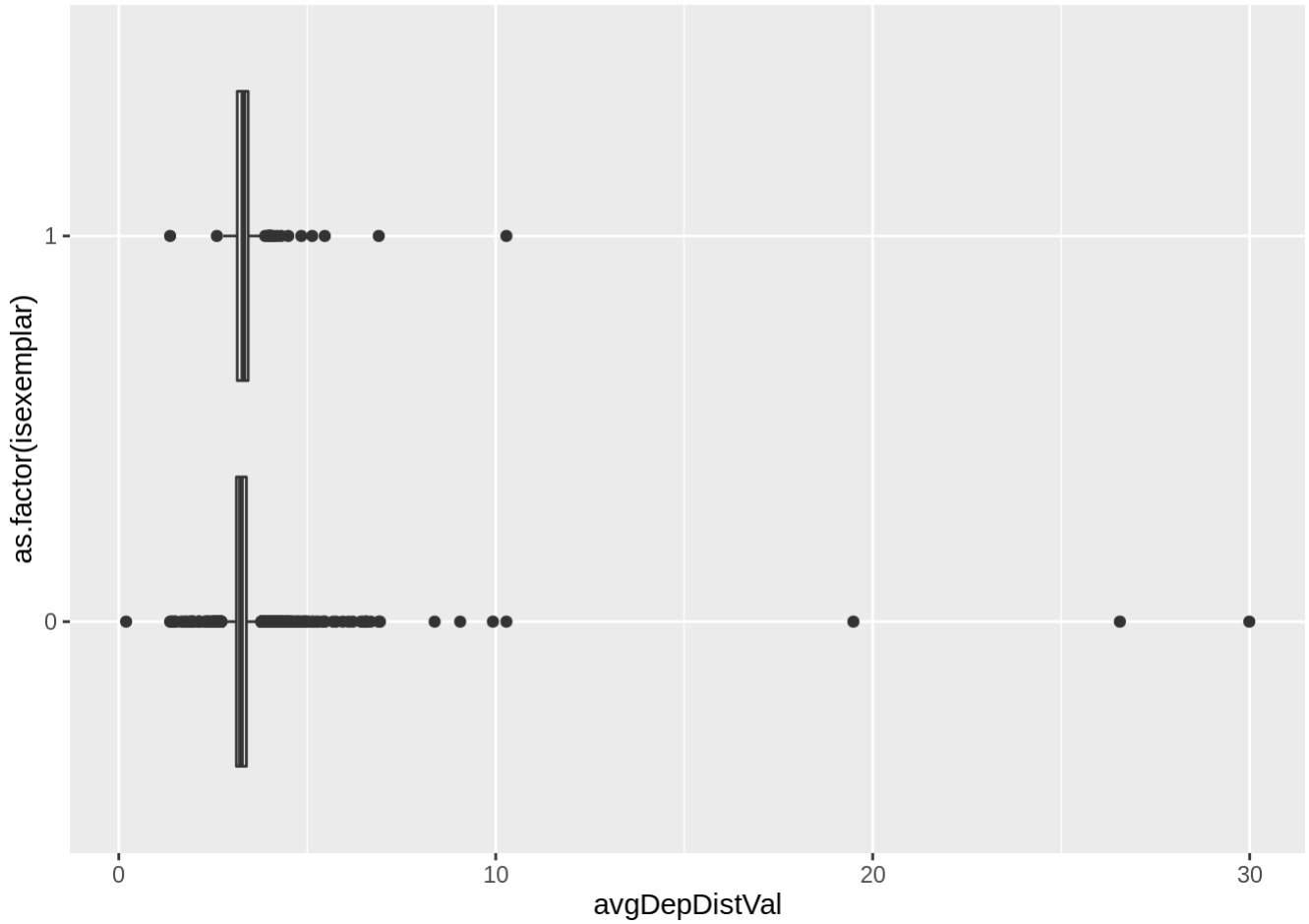

```
gf_boxplot(as.factor(isexemplar) ~ evennessVal, data = is.2)
```




```
gf_boxplot(as.factor(isexemplar) ~ giniDispVal, data = is.2)
```

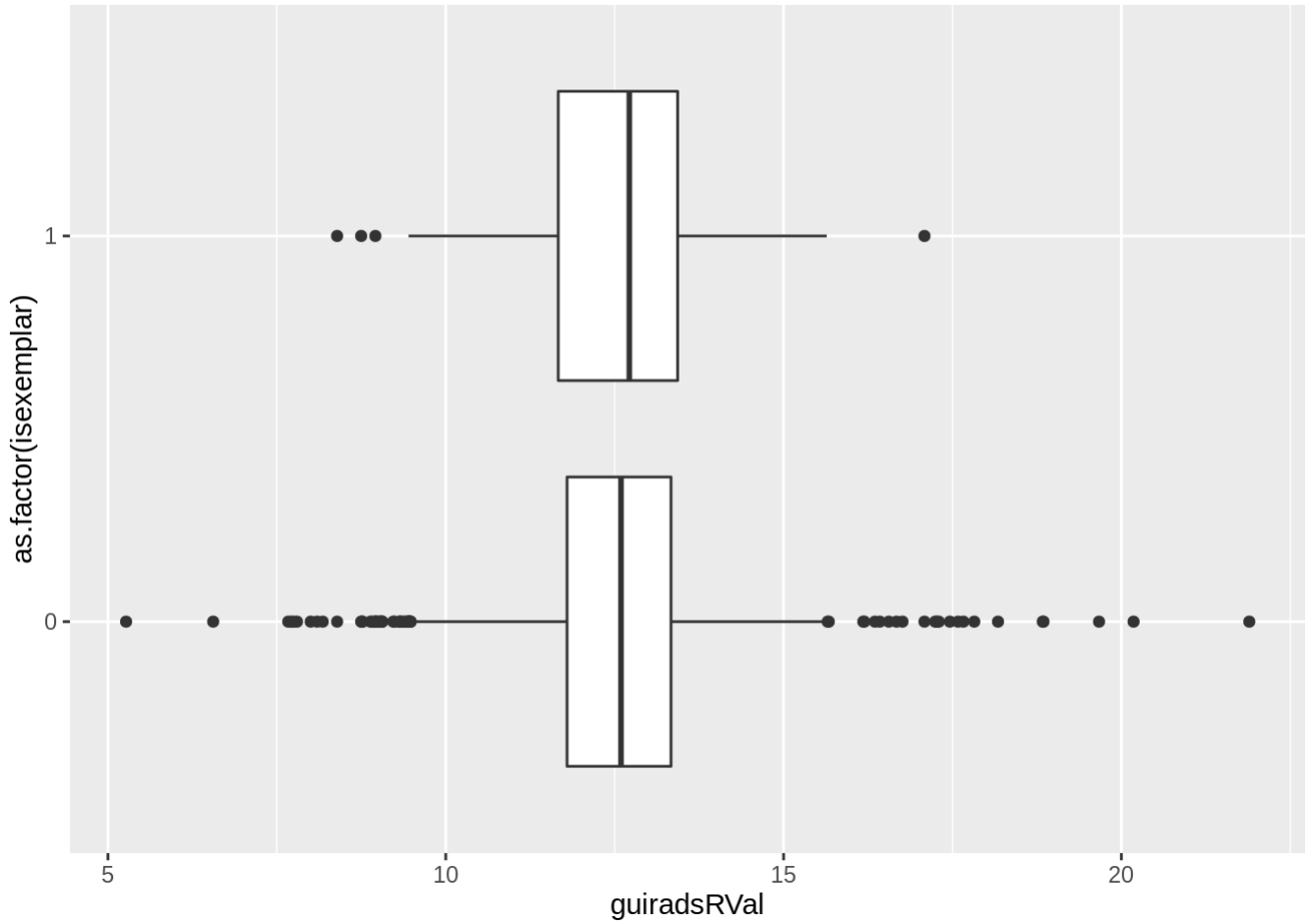



```
gf_boxplot(as.factor(isexemplar) ~ avgDepDistVal, data = is.2)
```

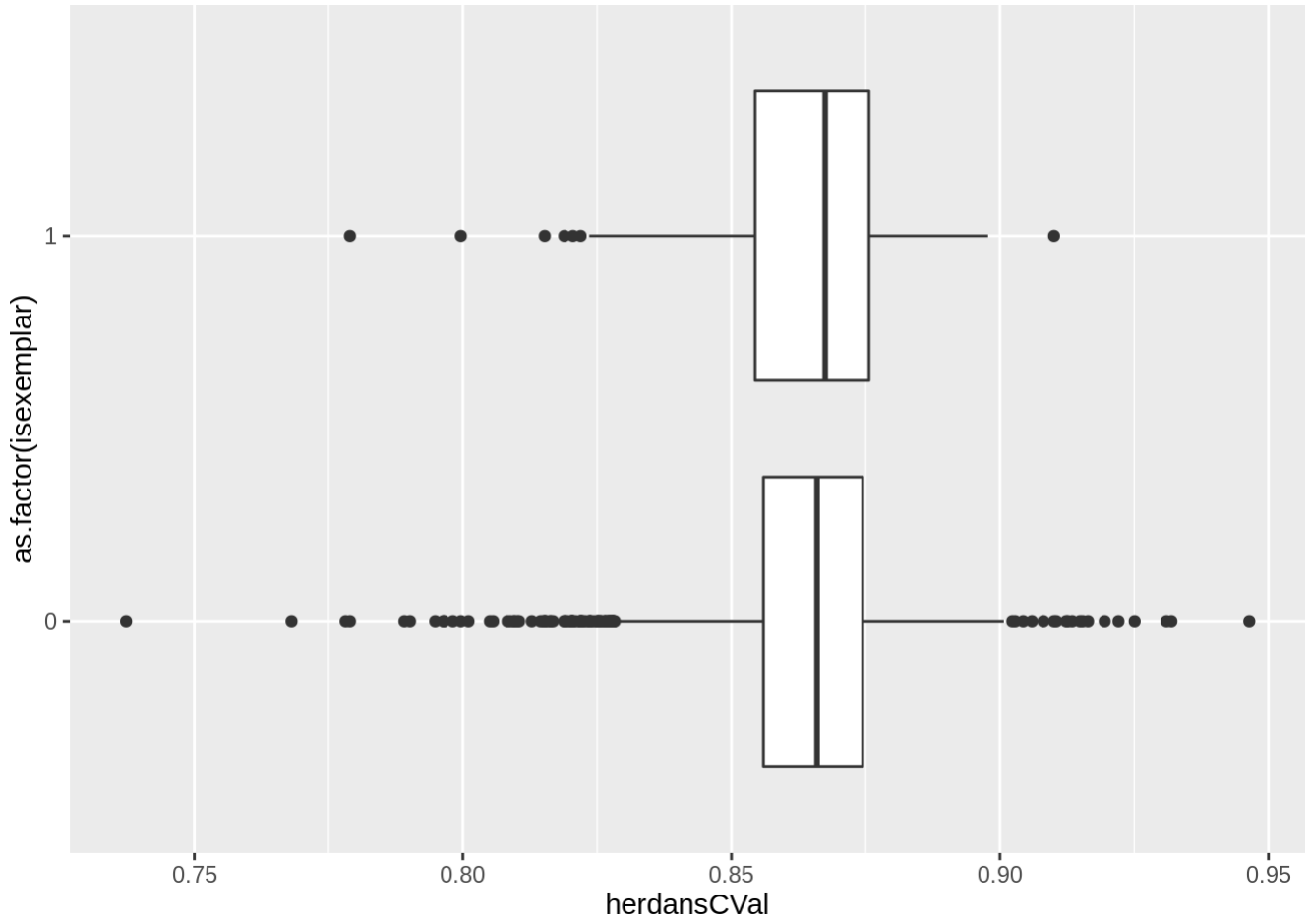


Various Named/Correlatted with Prior Values

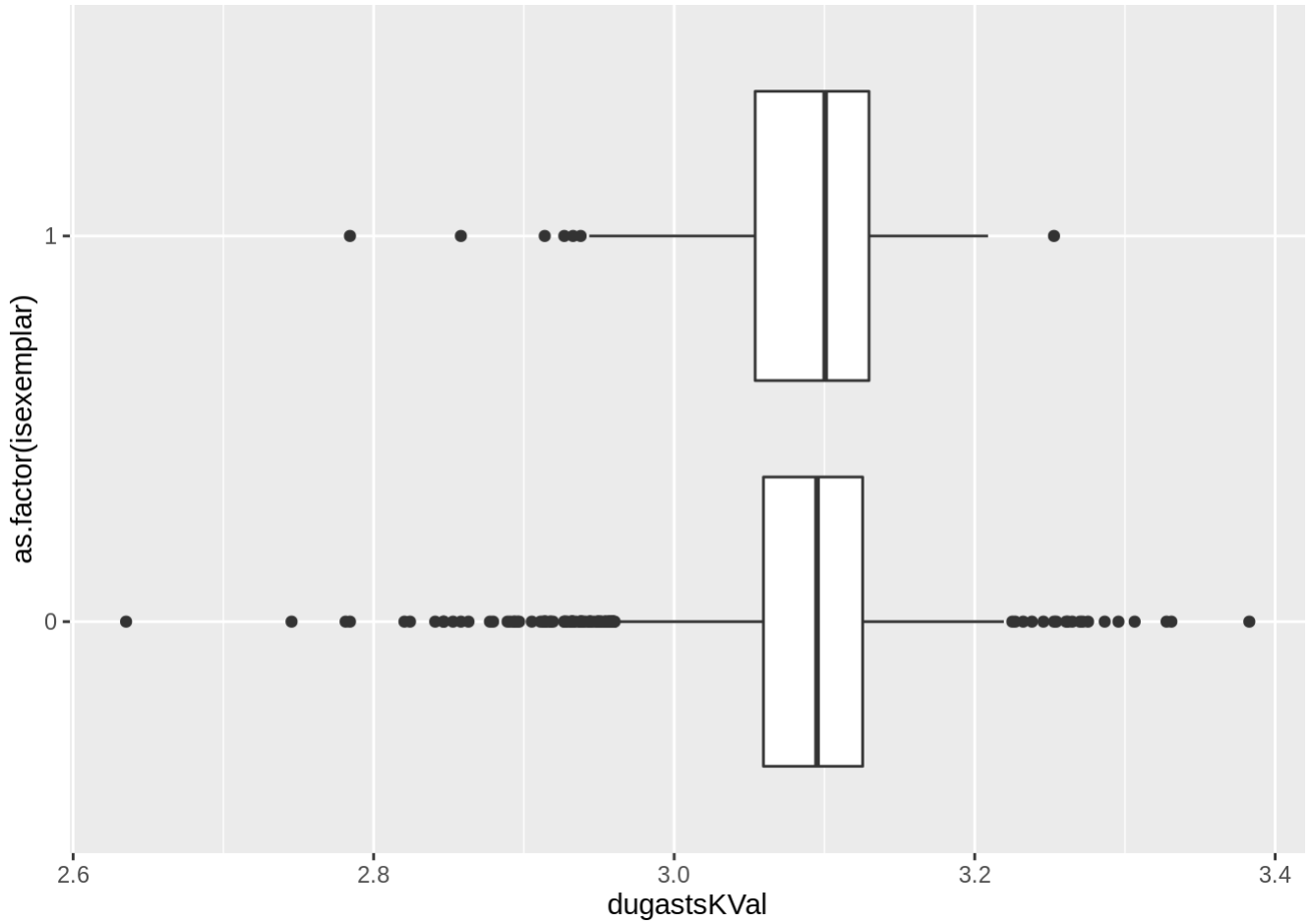
```
gf_boxplot(as.factor(isexemplar) ~ guiradsRVal, data = isdata.noNA)
```



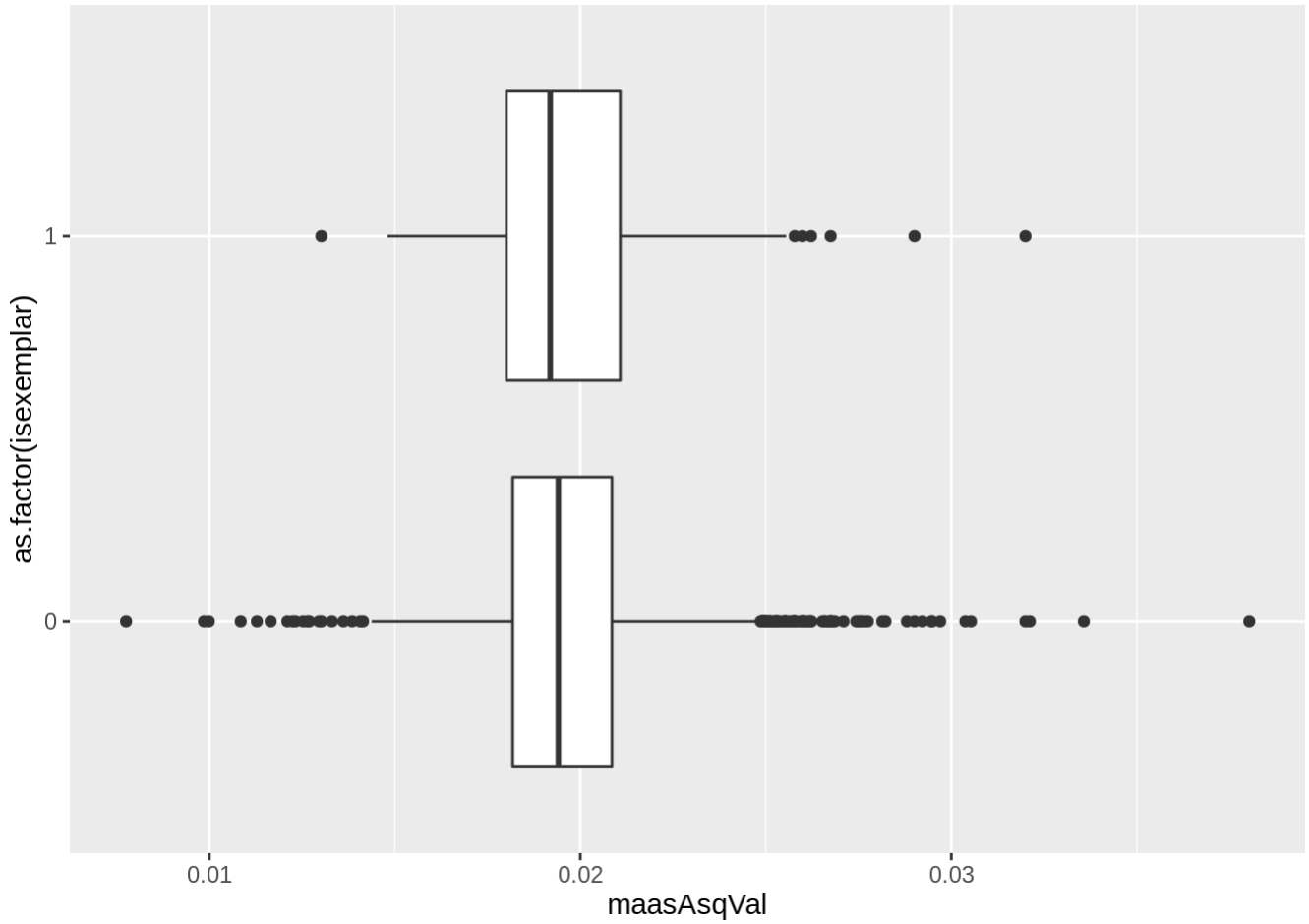

```
gf_boxplot(as.factor(isexemplar) ~ herdansCVal, data = isdata.noNA)
```



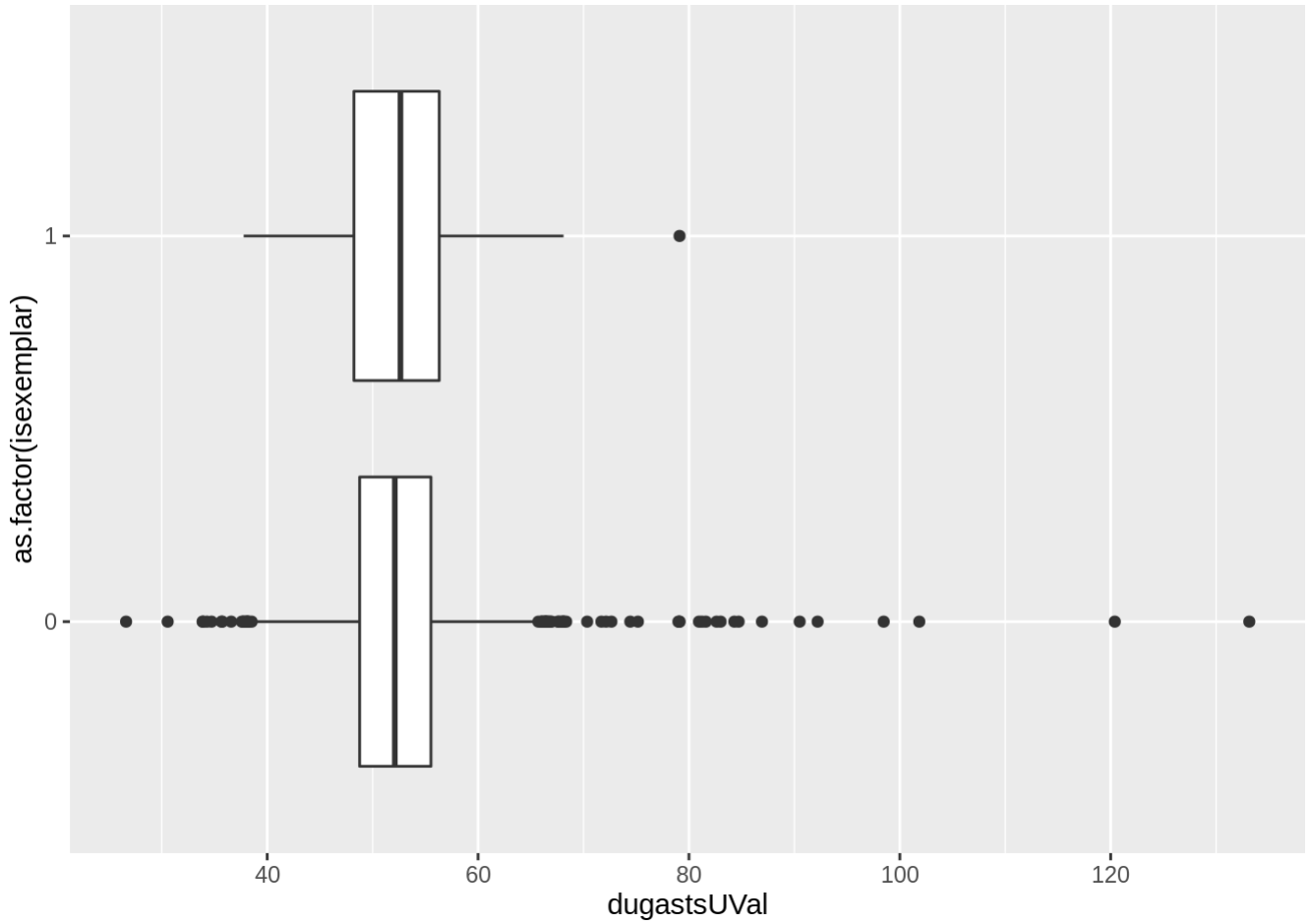

```
gf_boxplot(as.factor(isexemplar) ~ dugastsKVal, data = isdata.noNA)
```



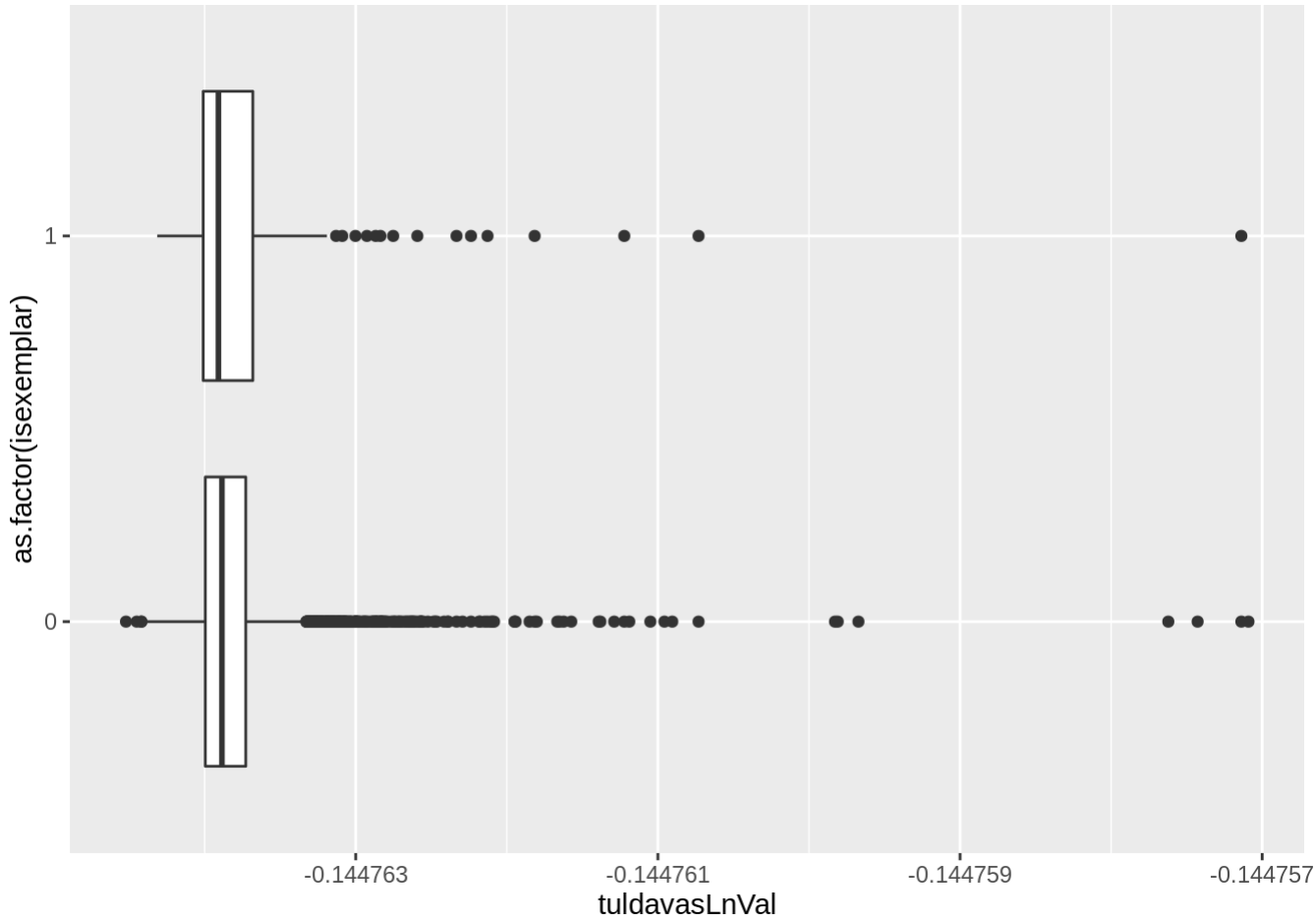

```
gf_boxplot(as.factor(isexemplar) ~ maasAsqVal, data = isdata.noNA)
```



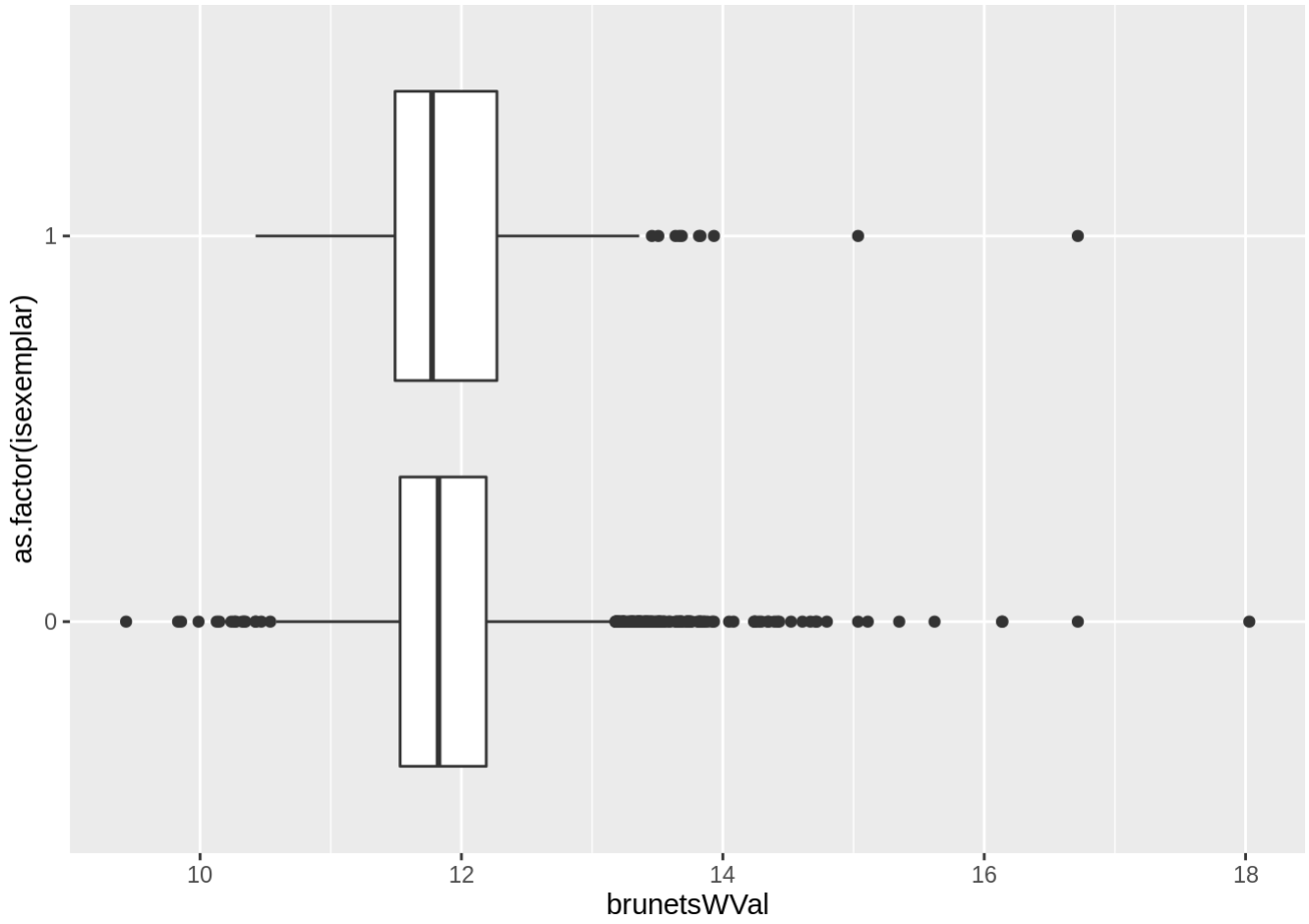

```
gf_boxplot(as.factor(isexemplar) ~ dugastsUVal, data = isdata.noNA)
```



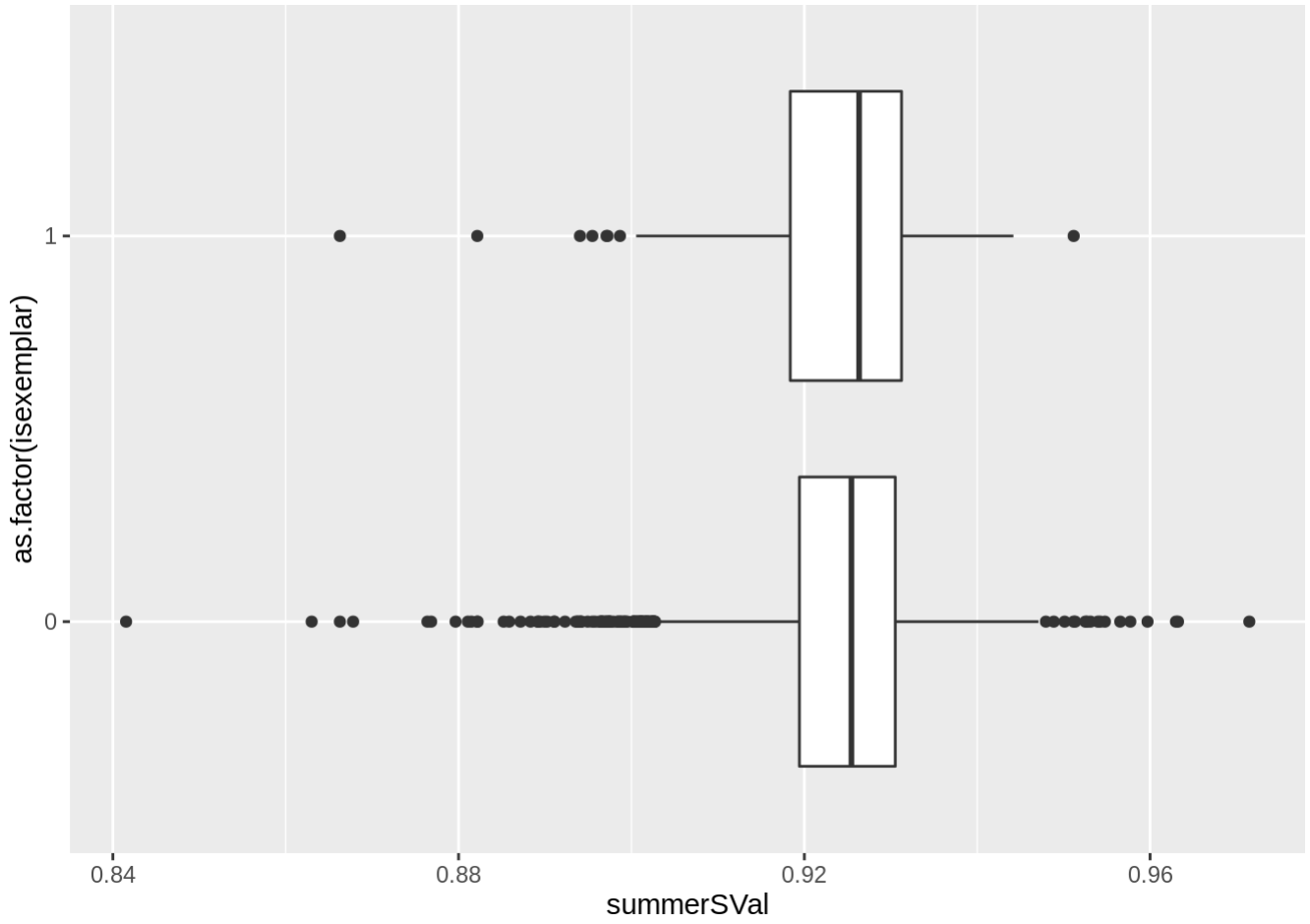

```
gf_boxplot(as.factor(isexemplar) ~ tuldavasLnVal, data = isdata.noNA)
```



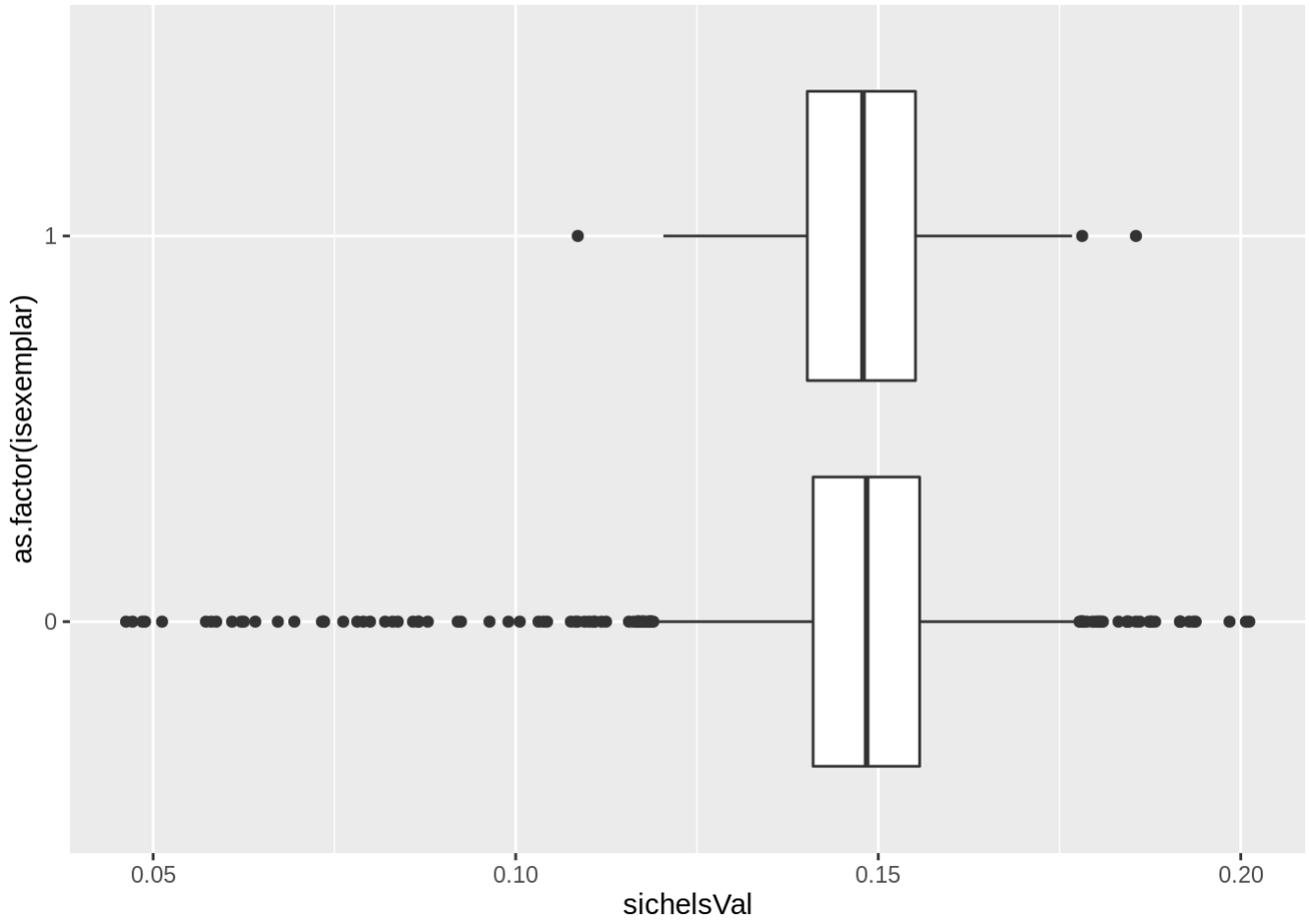

```
gf_boxplot(as.factor(isexemplar) ~ brunetsWVal, data = isdata.noNA)
```



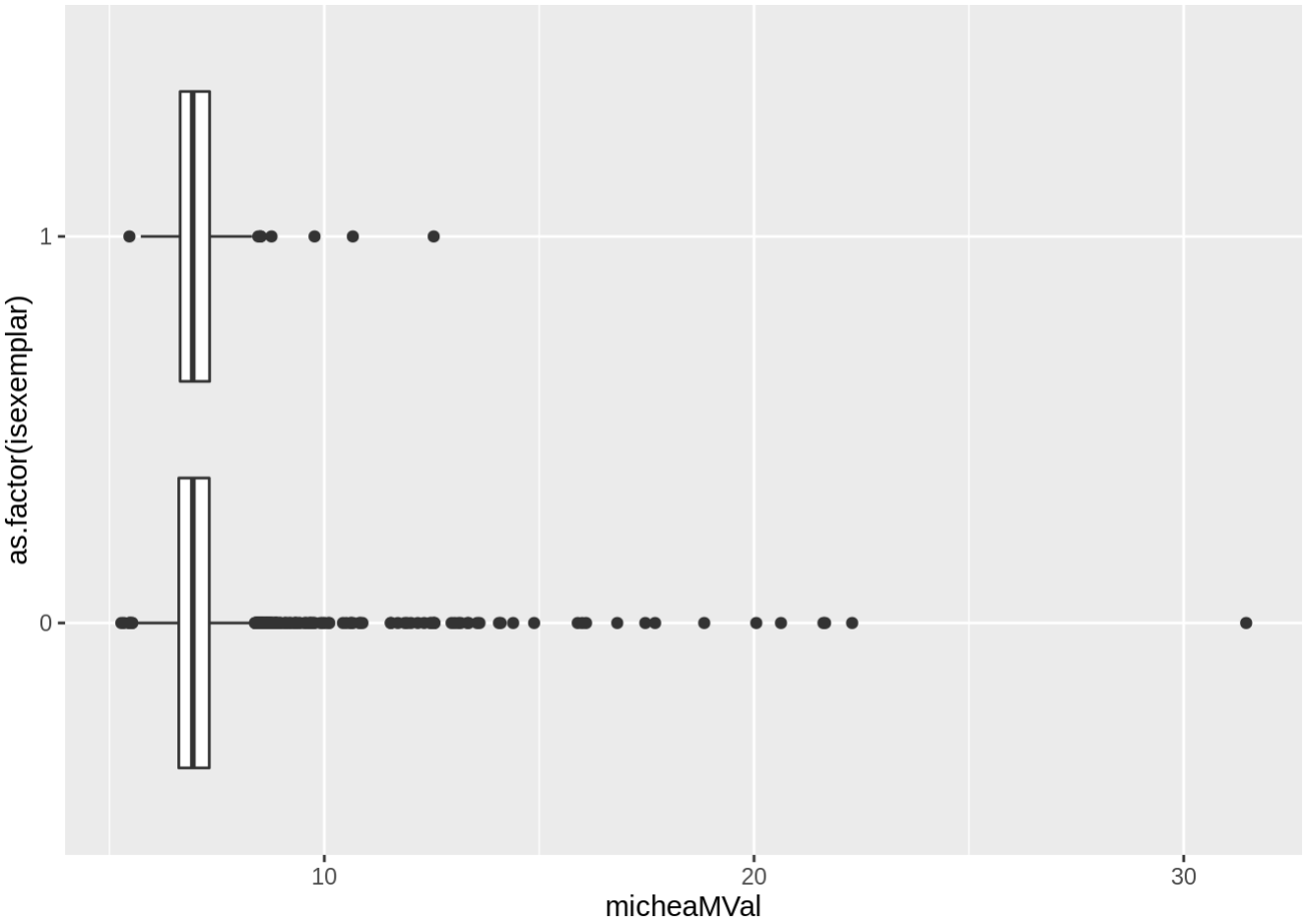

```
gf_boxplot(as.factor(isexemplar) ~ summerSVal, data = isdata.noNA)
```



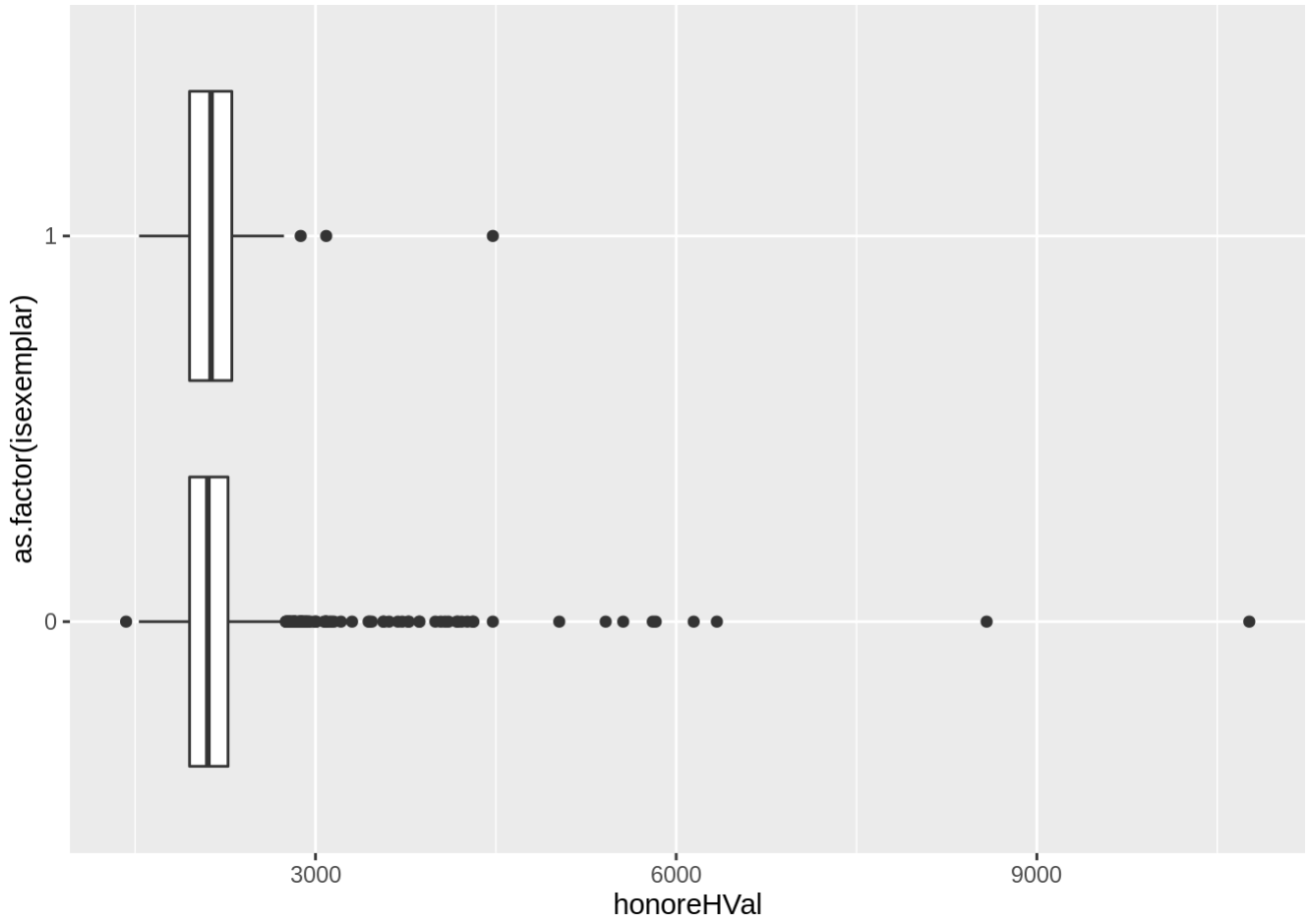

```
gf_boxplot(as.factor(isexemplar) ~ sichelsVal, data = isdata.noNA)
```



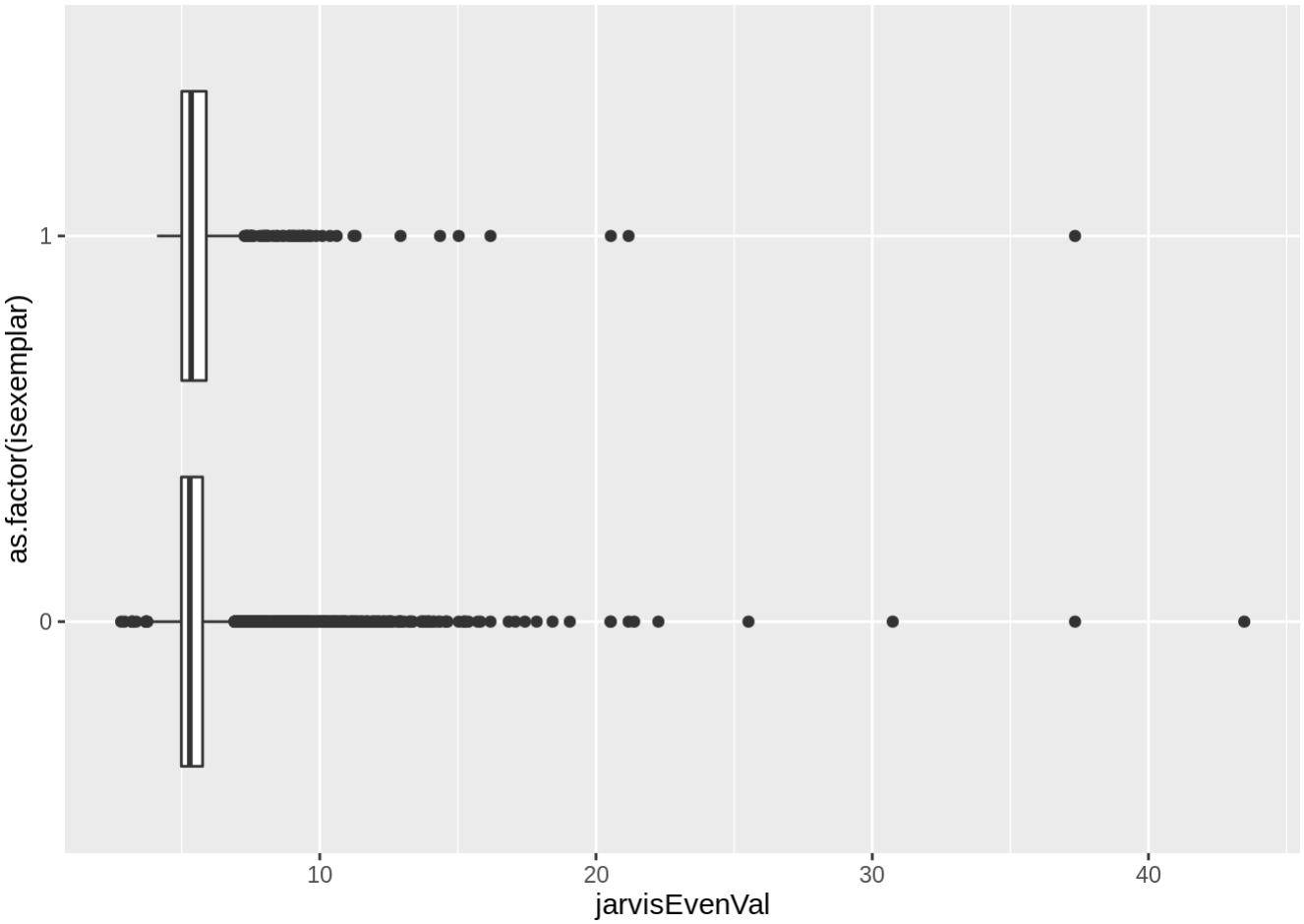

```
gf_boxplot(as.factor(isexemplar) ~ micheaMVal, data = isdata.noNA)
```



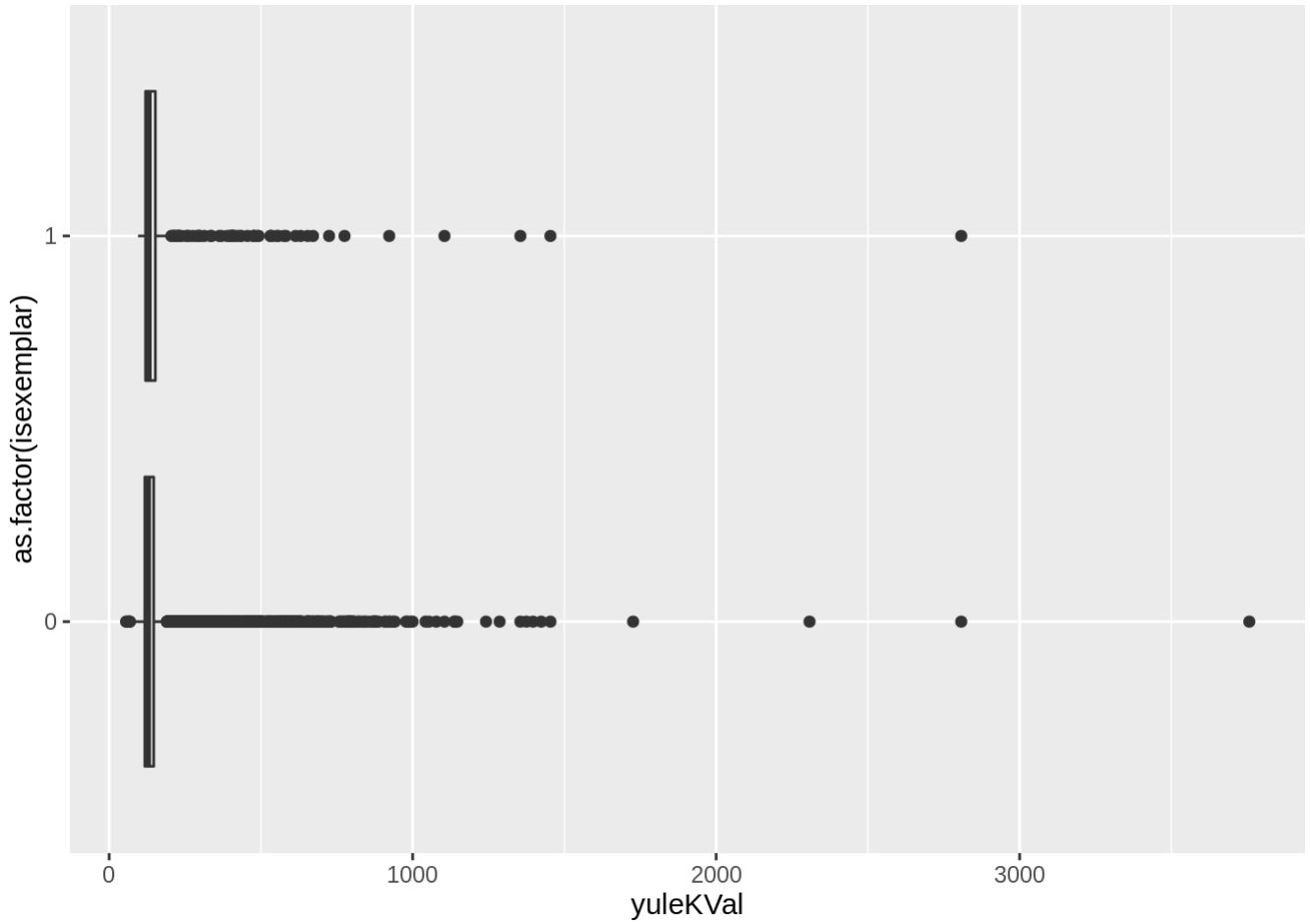

```
gf_boxplot(as.factor(isexemplar) ~ honoreHVal, data = isdata.noNA)
```



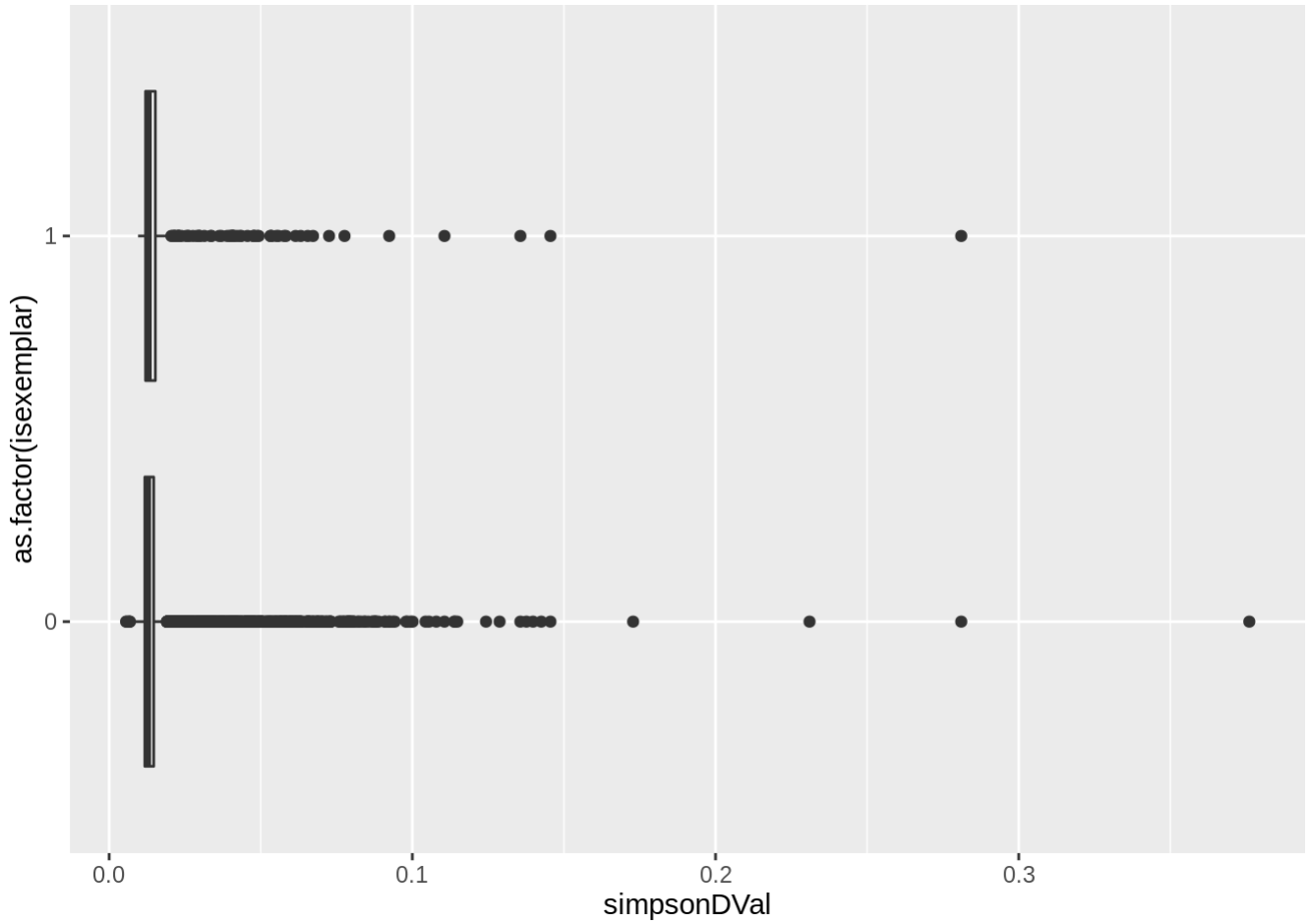

```
gf_boxplot(as.factor(isexemplar) ~ jarvisEvenVal, data = isdata.noNA)
```



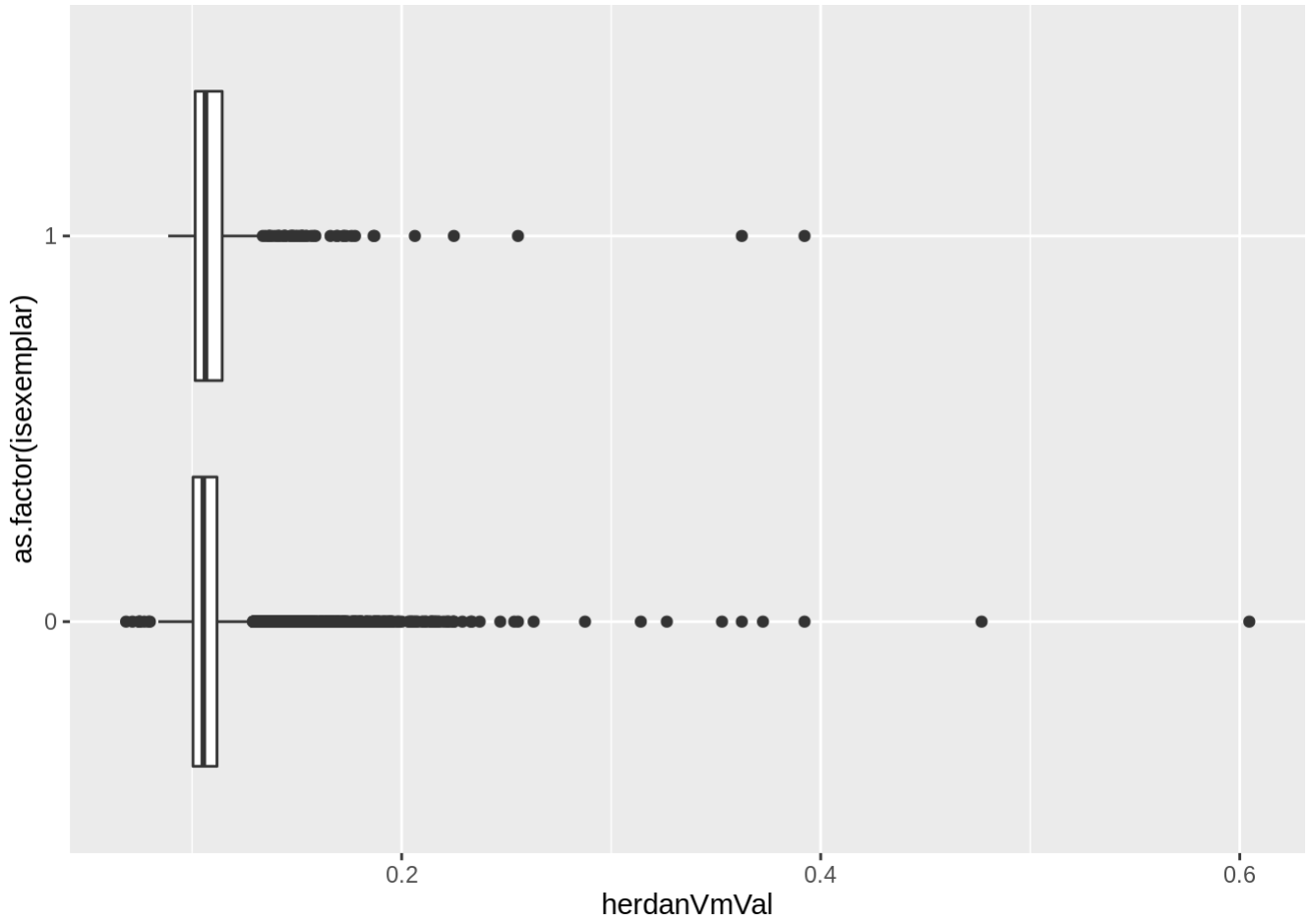

```
gf_boxplot(as.factor(isexemplar) ~ yuleKVal, data = isdata.noNA)
```



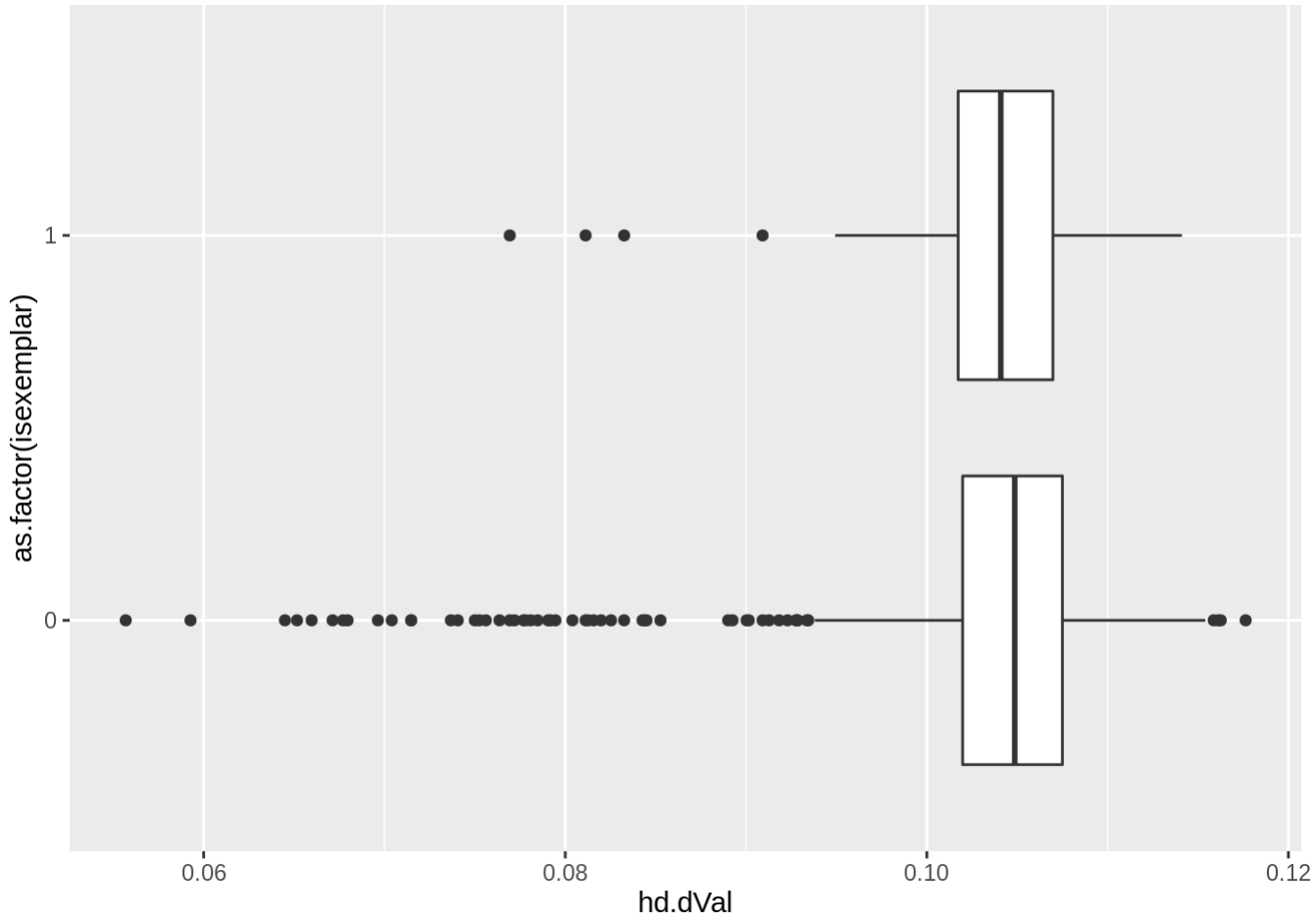

```
gf_boxplot(as.factor(isexemplar) ~ simpsonDVal, data = isdata.noNA)
```



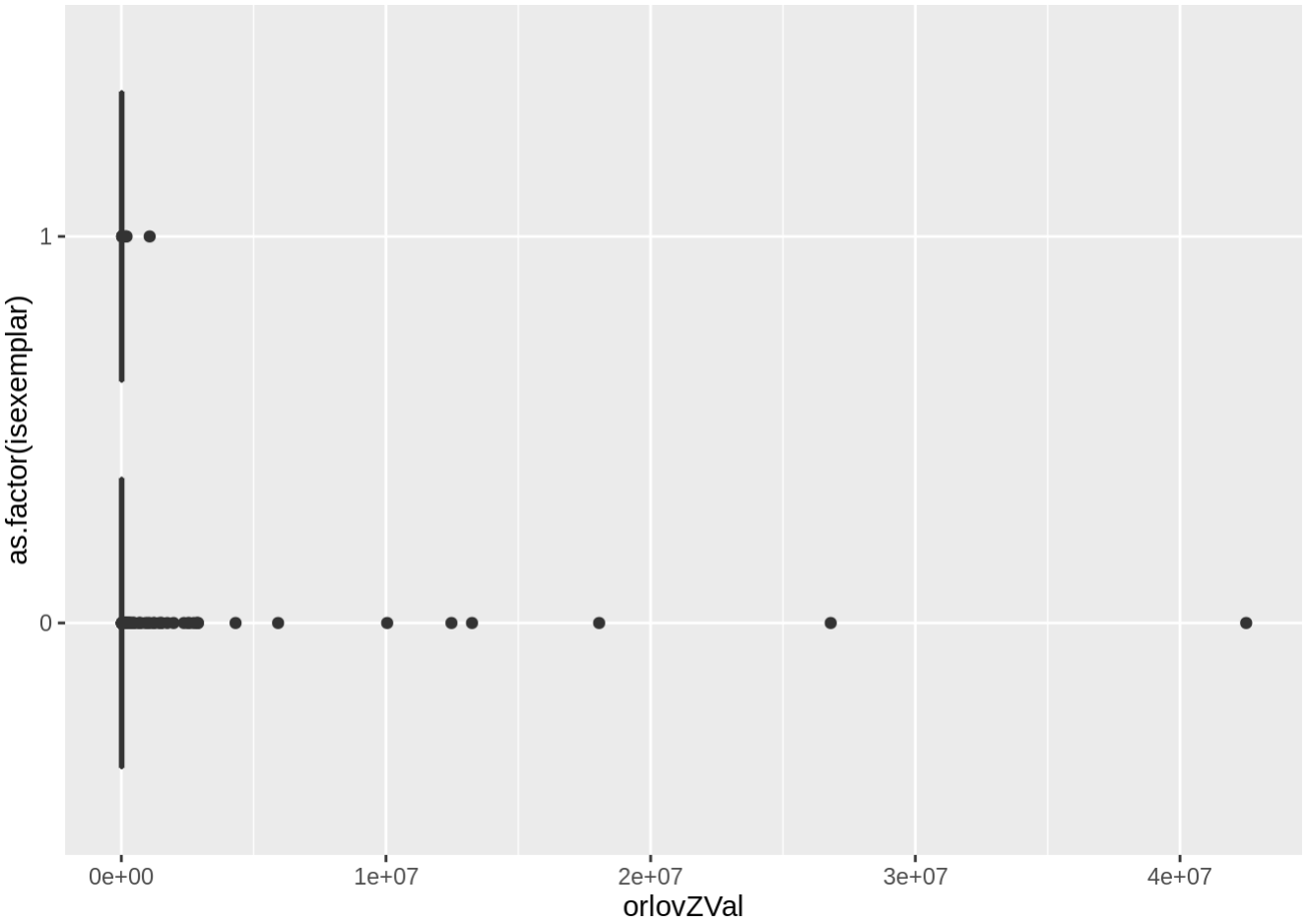

```
gf_boxplot(as.factor(isexemplar) ~ herdanVmVal, data = isdata.noNA)
```



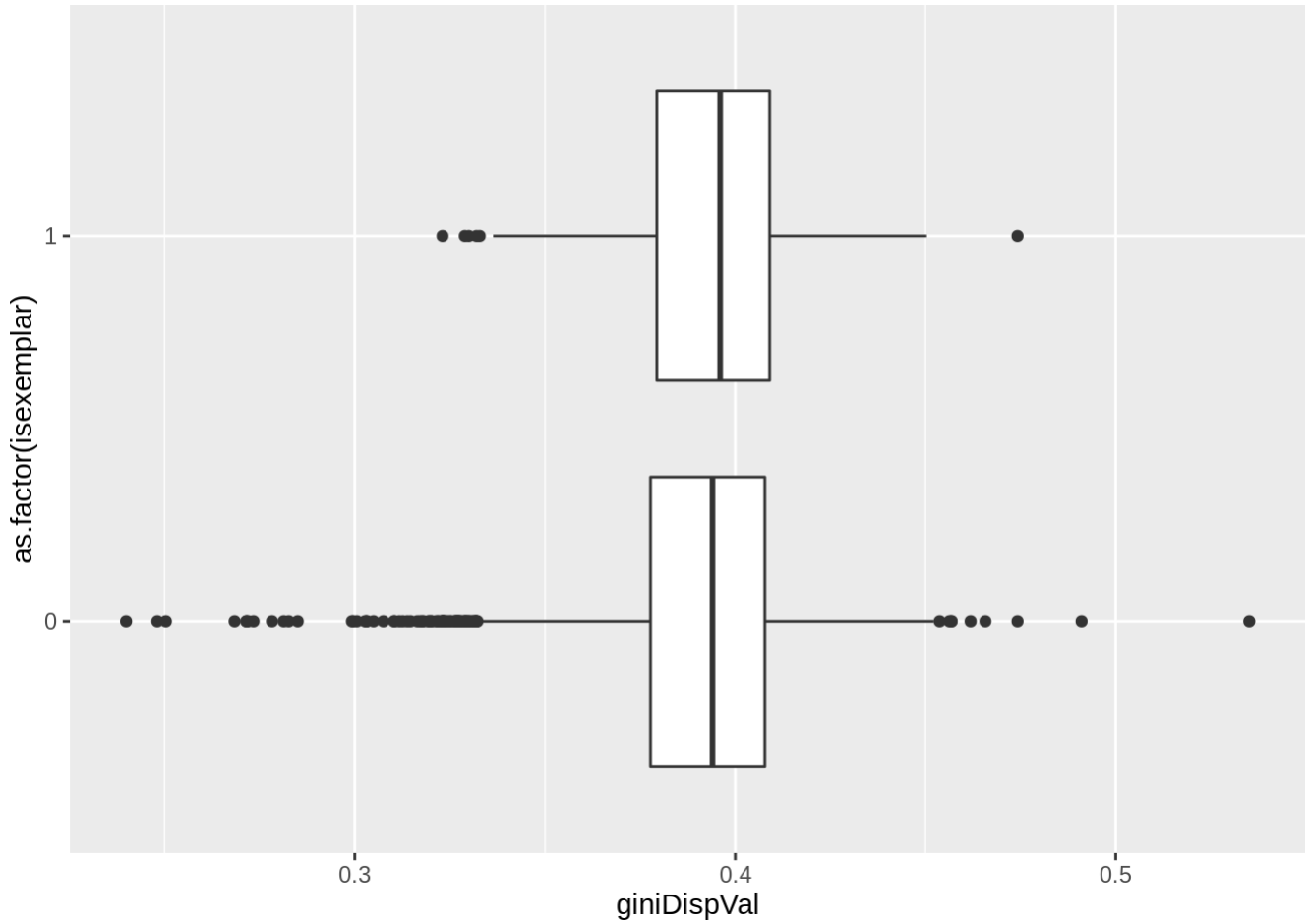

```
gf_boxplot(as.factor(isexemplar) ~ hd.dVal, data = isdata.noNA)
```



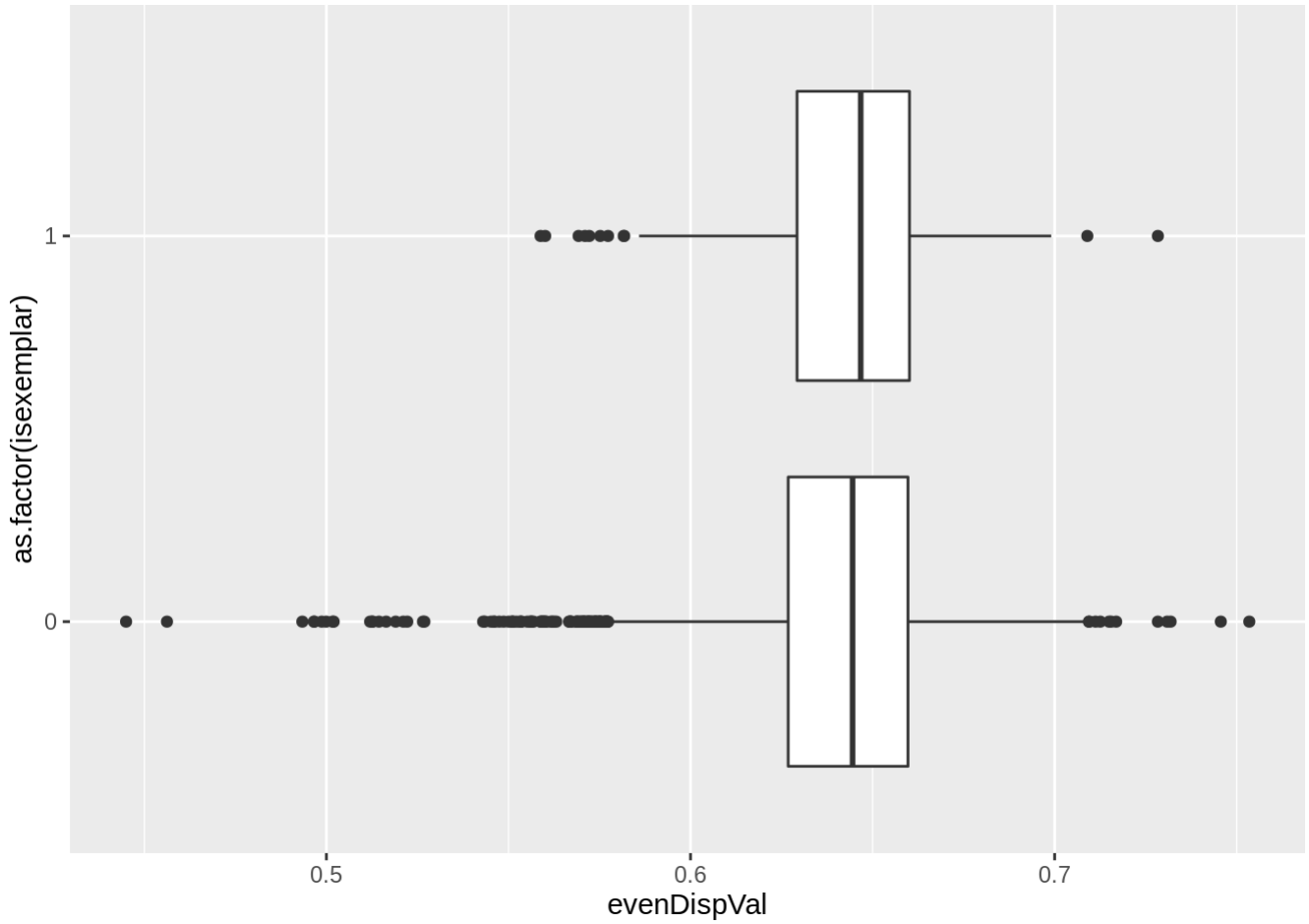

```
gf_boxplot(as.factor(isexemplar) ~ orlovZVal, data = isdata.noNA)
```



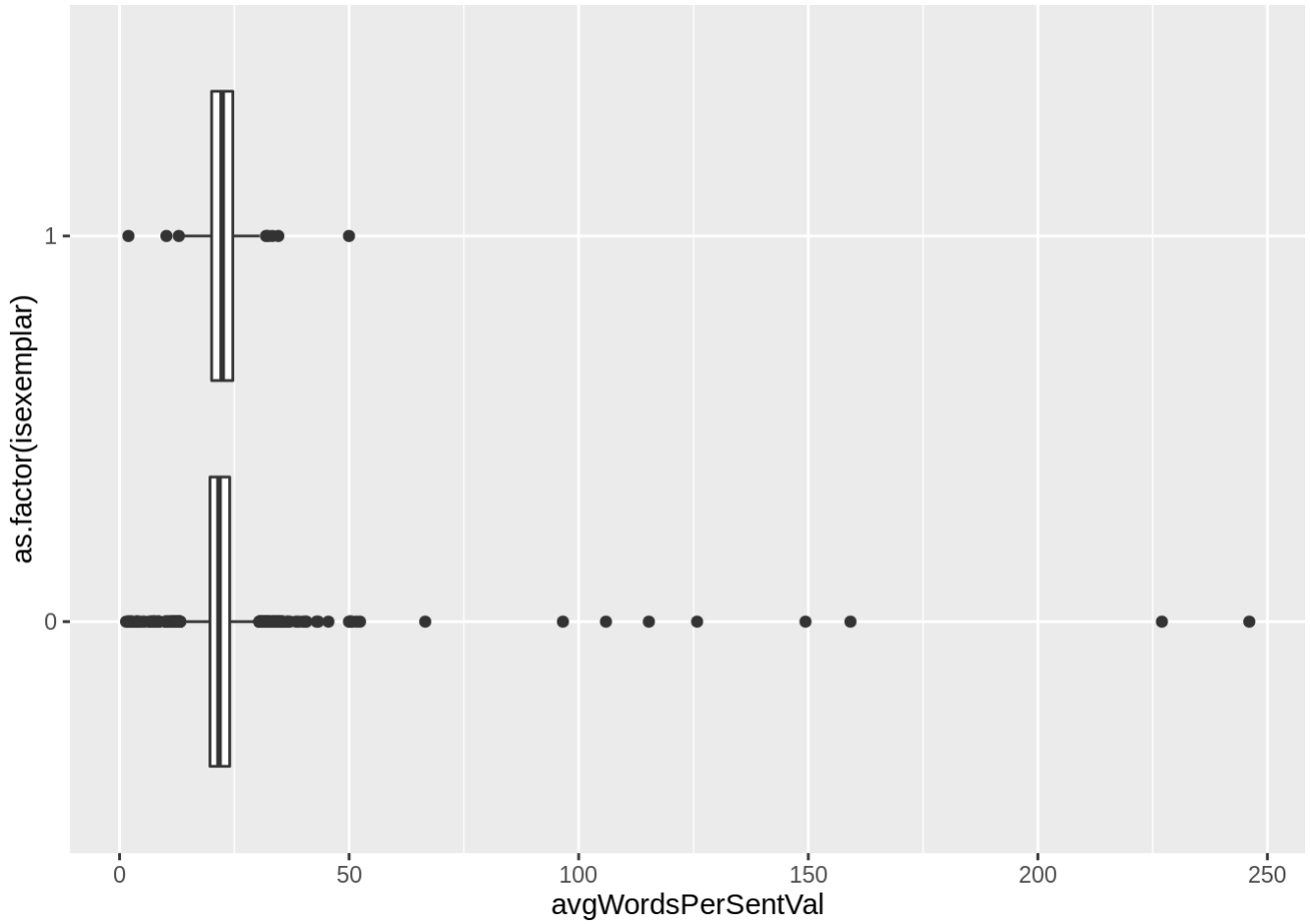

```
gf_boxplot(as.factor(isexemplar) ~ giniDispVal, data = isdata.noNA)
```



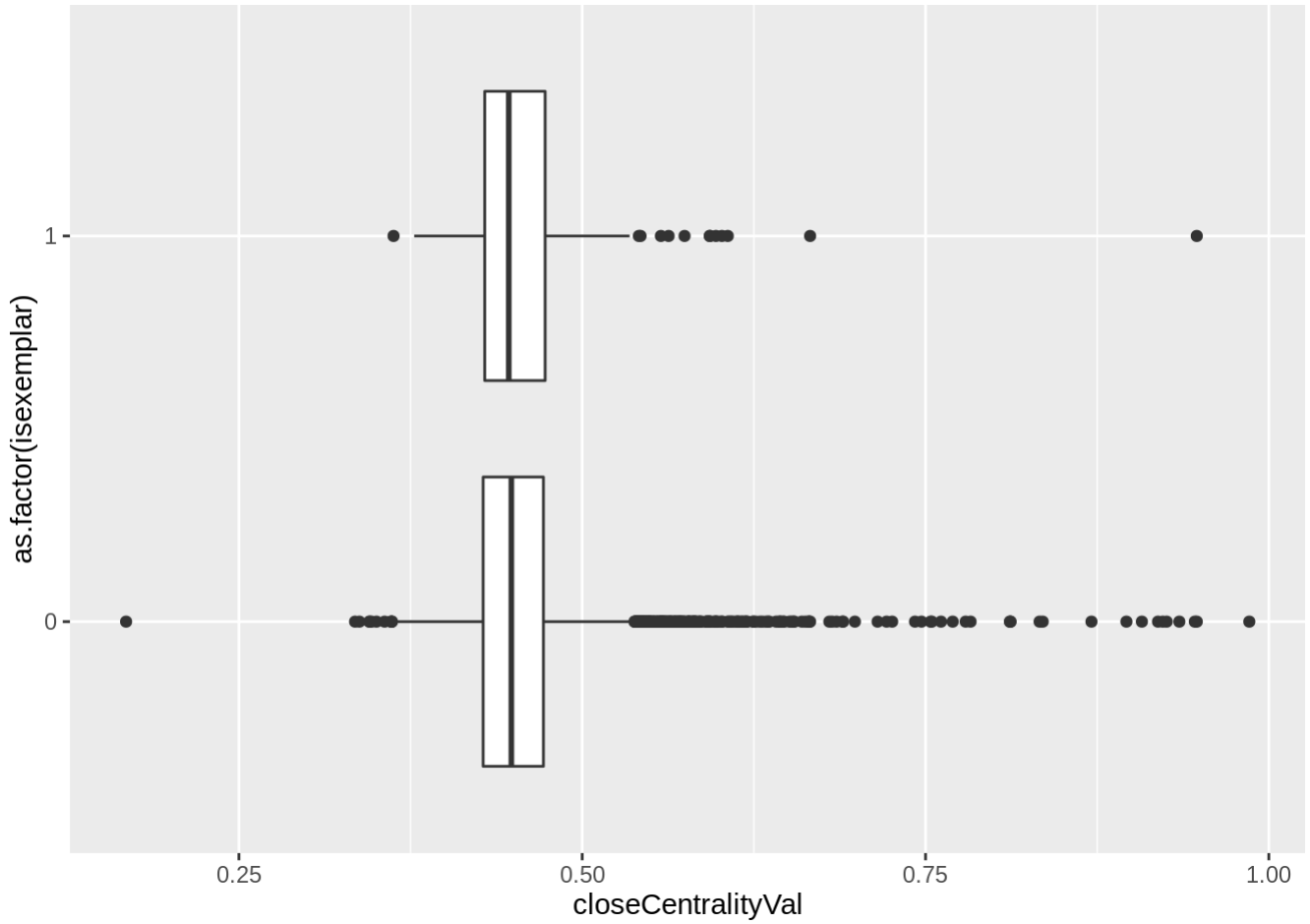

```
gf_boxplot(as.factor(isexemplar) ~ evenDispVal, data = isdata.noNA)
```



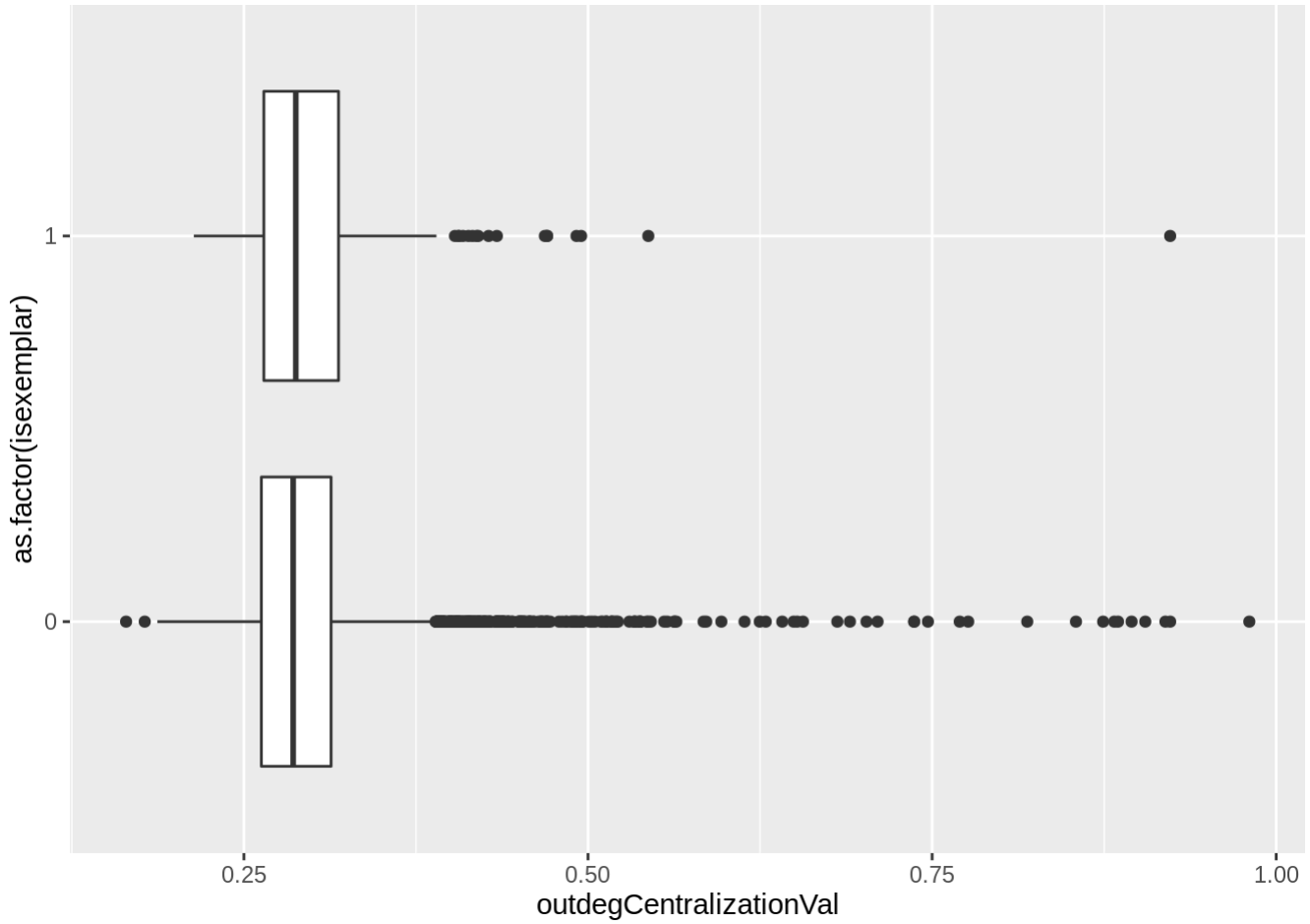

```
gf_boxplot(as.factor(isexemplar) ~ avgWordsPerSentVal, data = isdata.noNA)
```



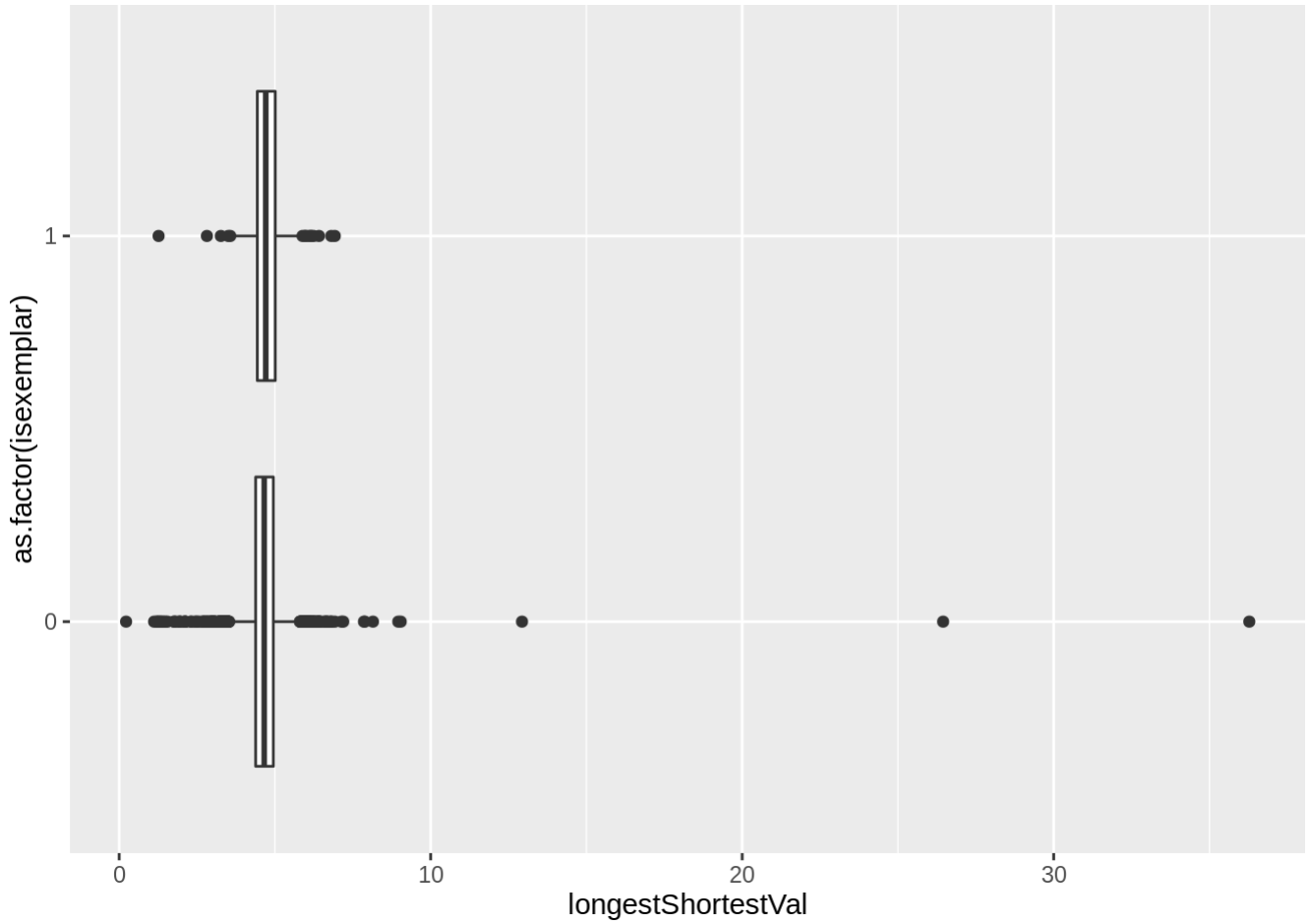

```
gf_boxplot(as.factor(isexemplar) ~ closeCentralityVal, data = isdata.noNA)
```



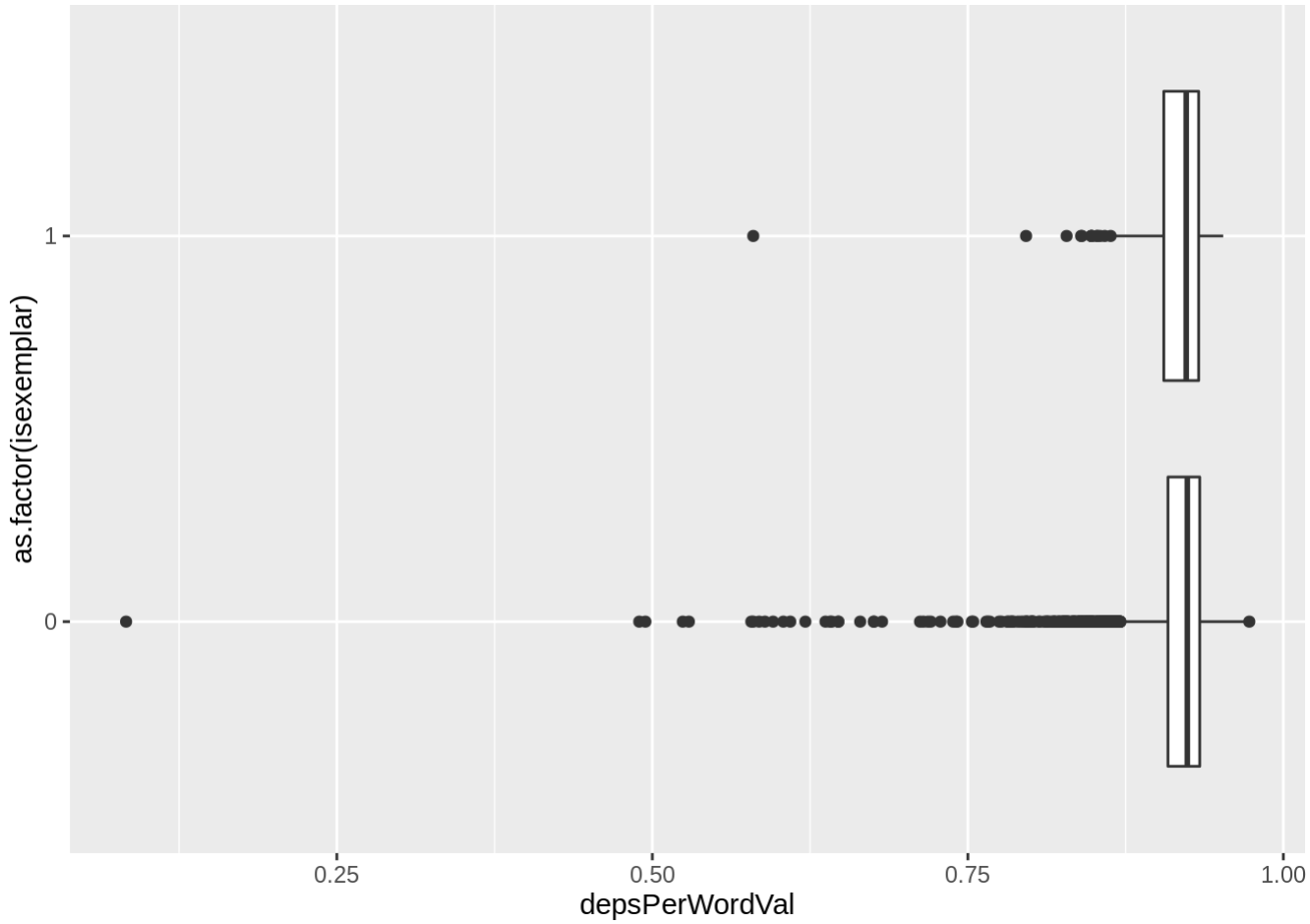

```
gf_boxplot(as.factor(isexemplar) ~ outdegCentralizationVal, data = isdata.noNA)
```




```
gf_boxplot(as.factor(isexemplar) ~ longestShortestVal, data = isdata.noNA)
```



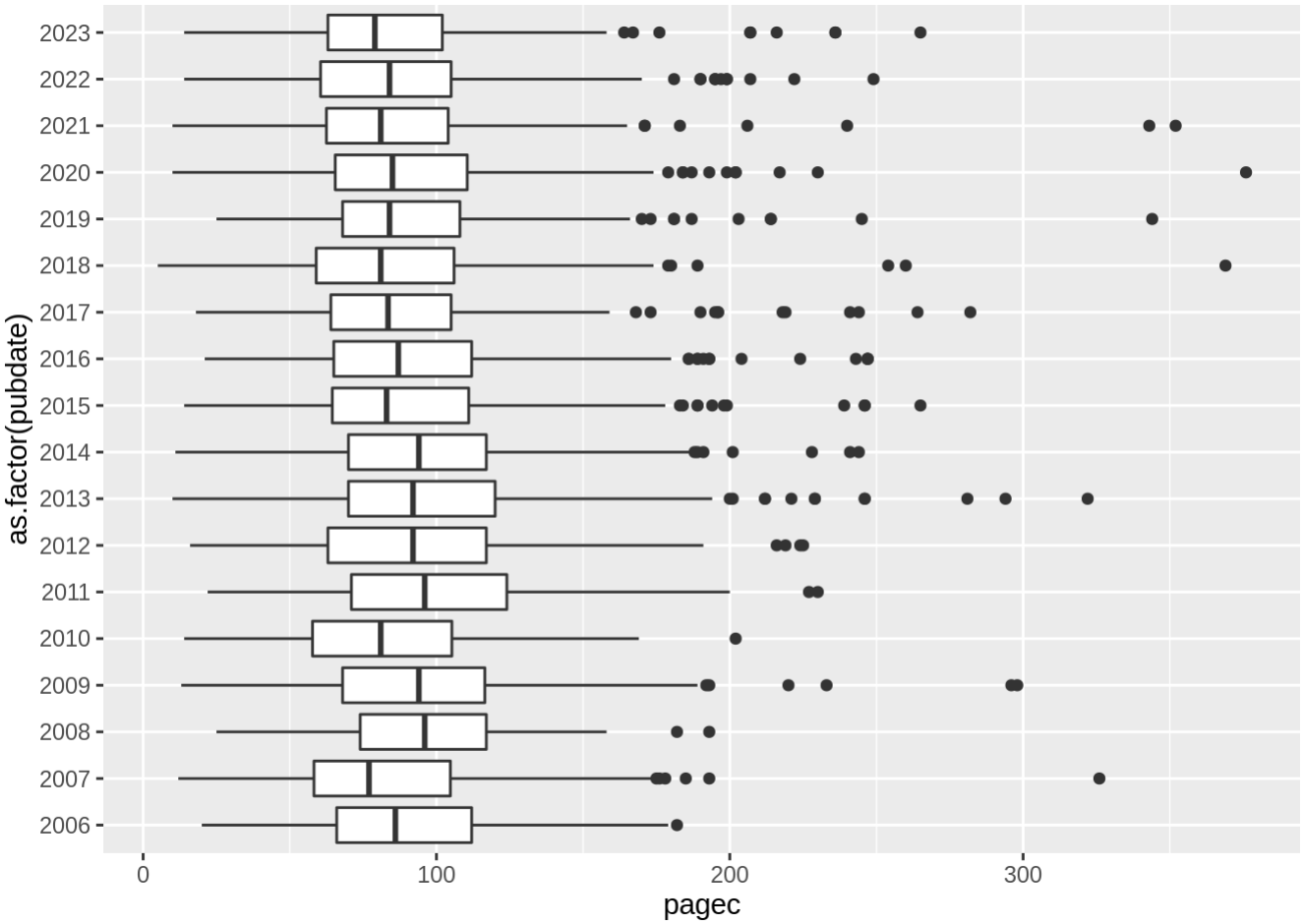

```
gf_boxplot(as.factor(isexemplar) ~ depsPerWordVal, data = isdata.noNA)
```



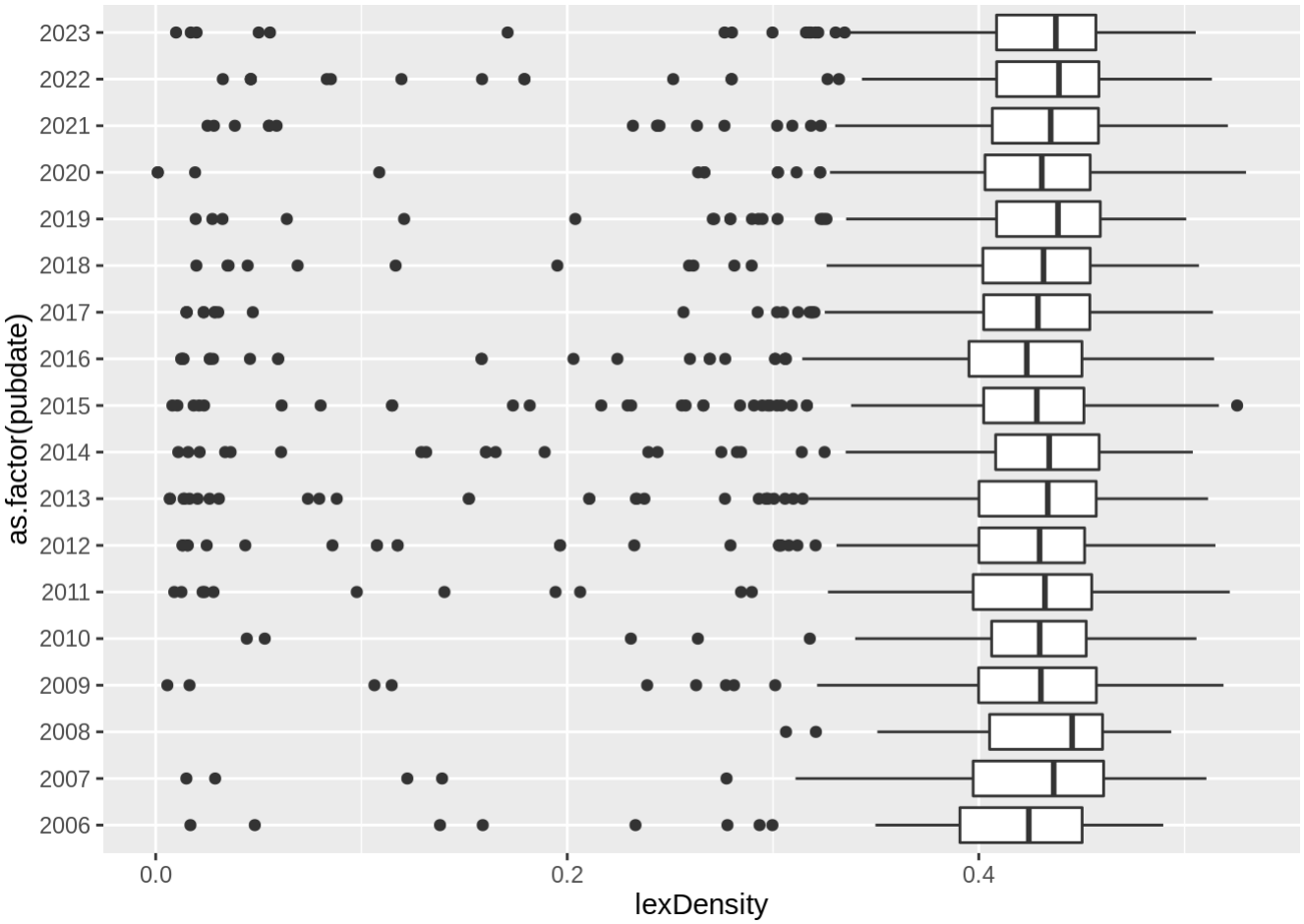
Different Axes

Publication Date

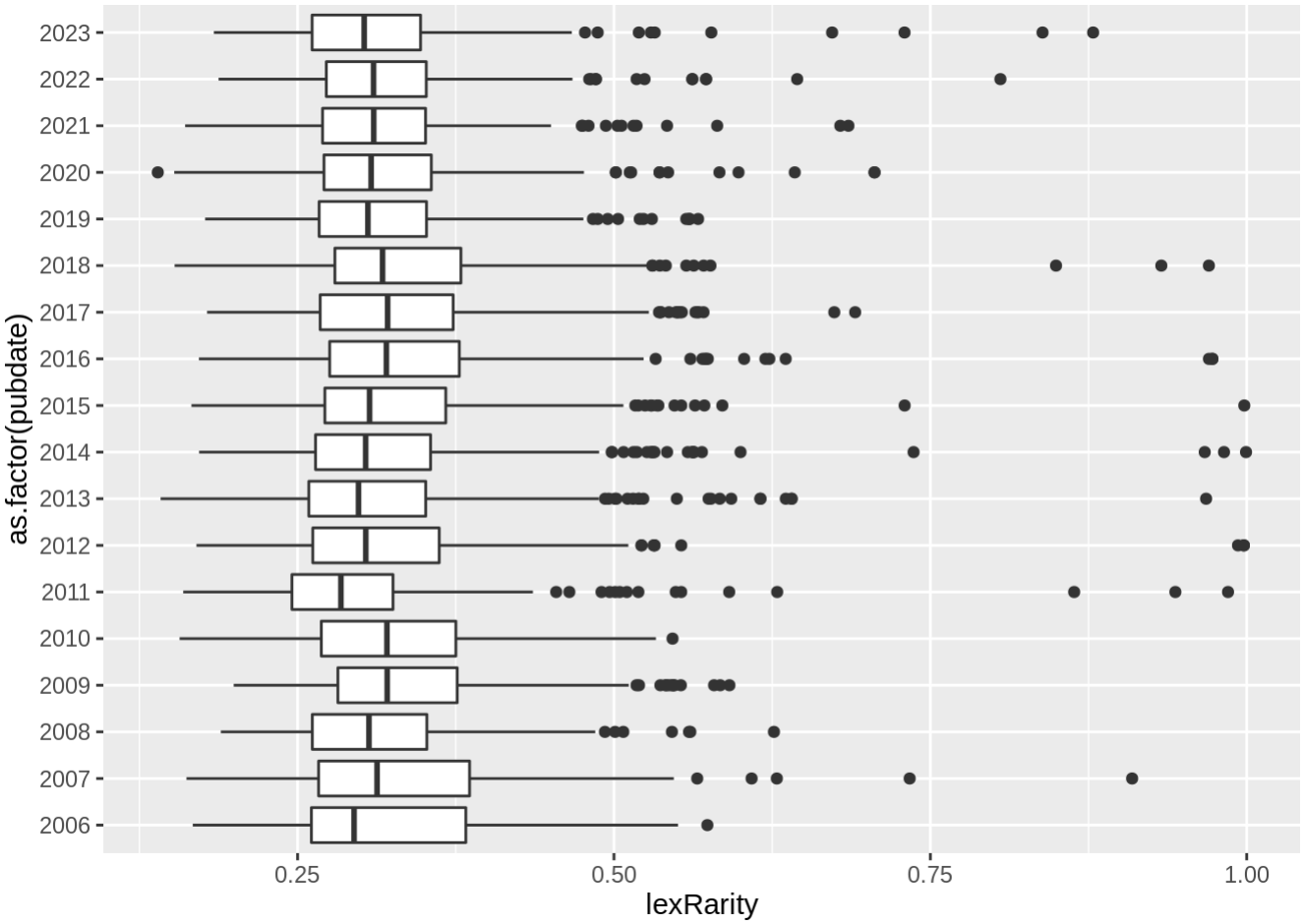
```
gf_boxplot(as.factor(pubdate) ~ pagec, data = is.2)
```



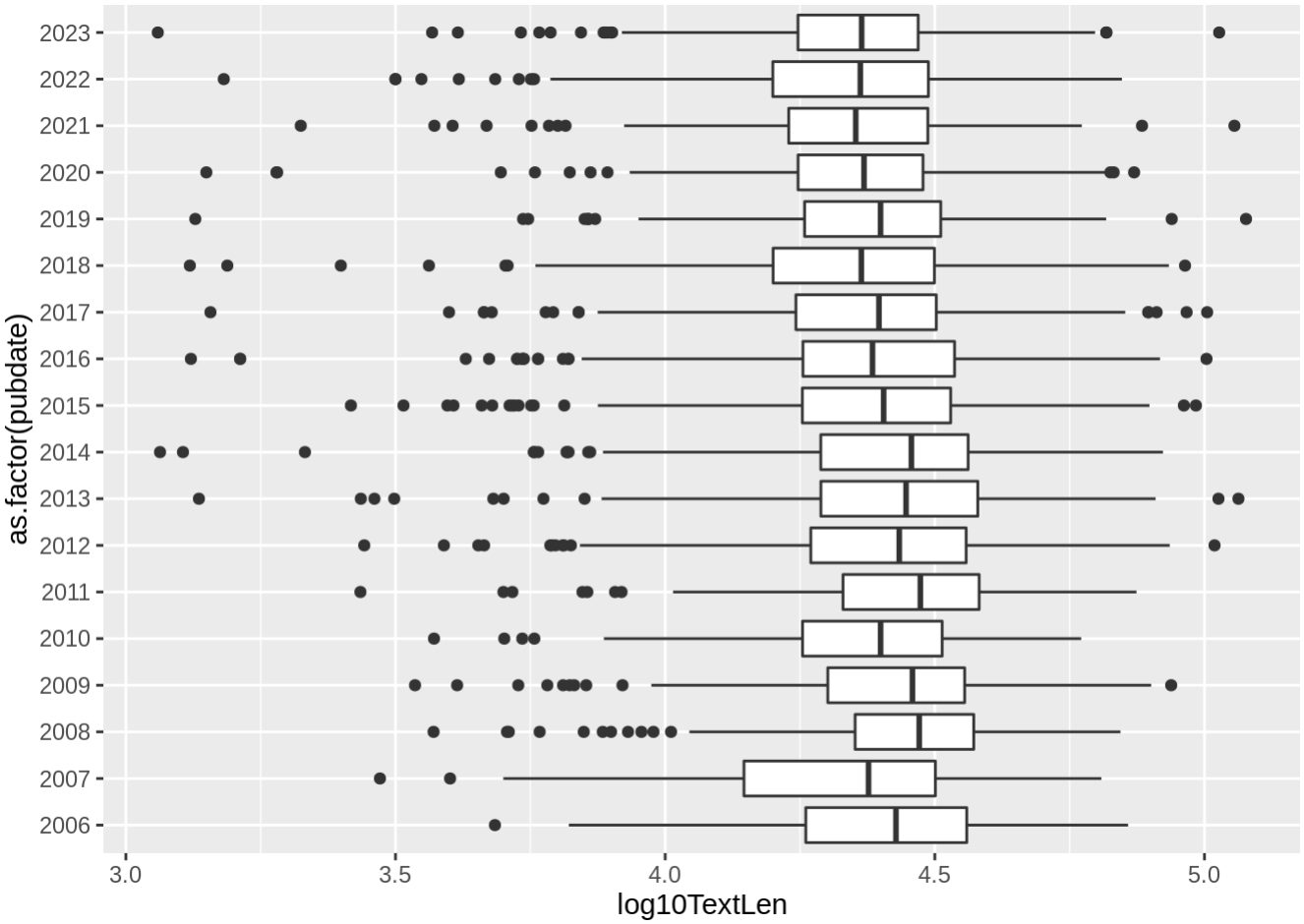

```
gf_boxplot(as.factor(pubdate) ~ lexDensity, data = is.2)
```



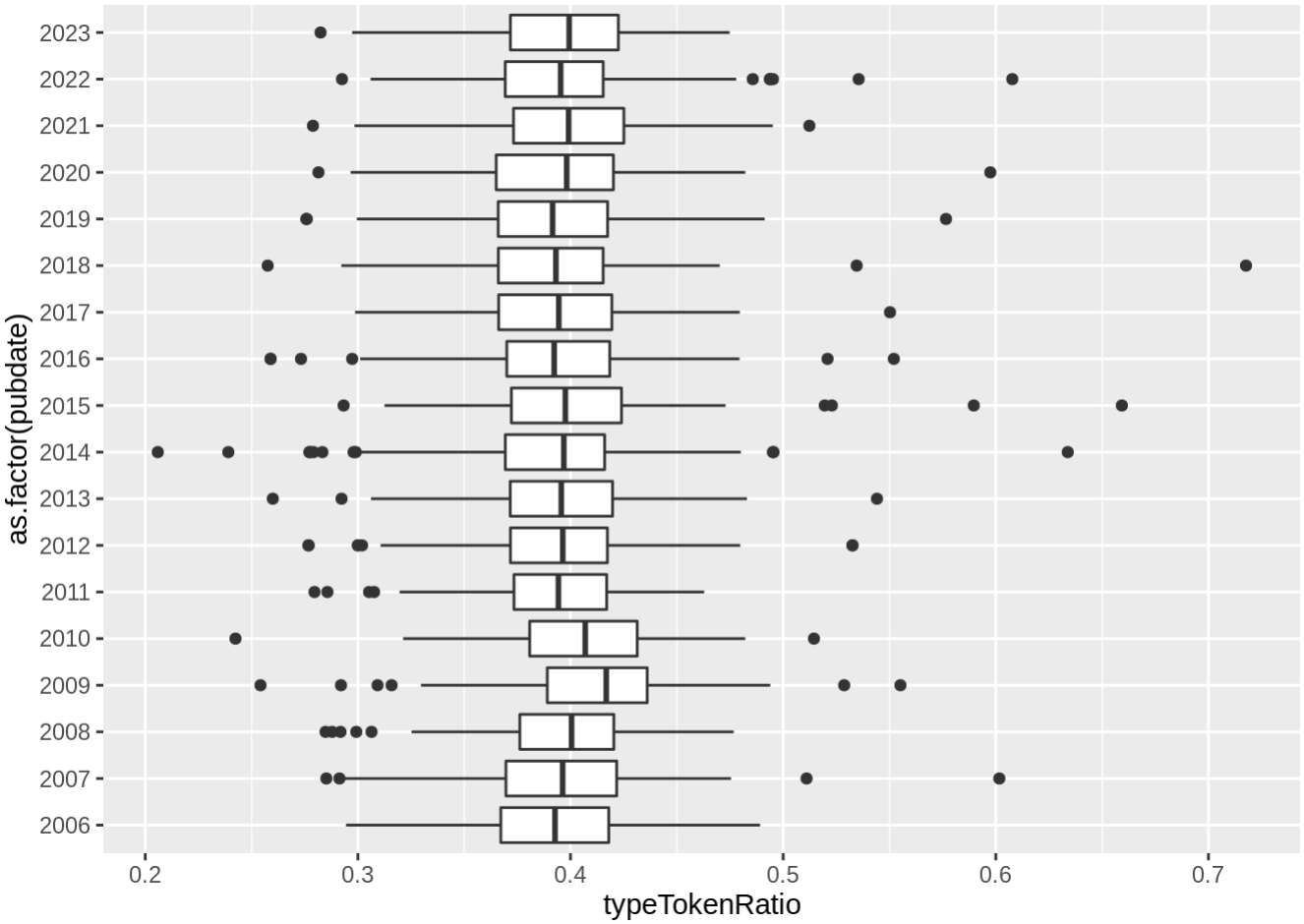

```
gf_boxplot(as.factor(pubdate) ~ lexRarity, data = is.2)
```



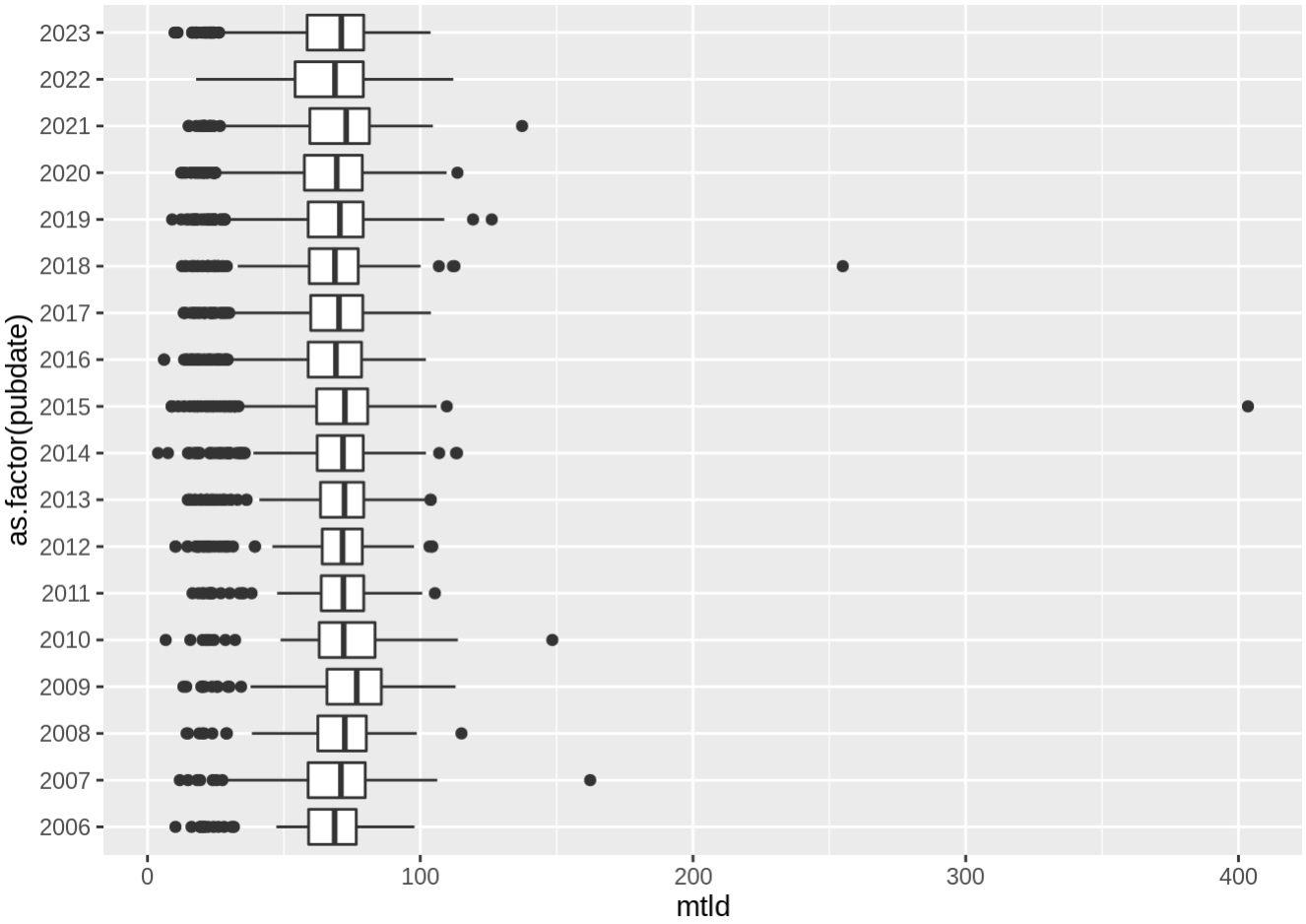

```
gf_boxplot(as.factor(pubdate) ~ log10TextLen, data = is.2)
```



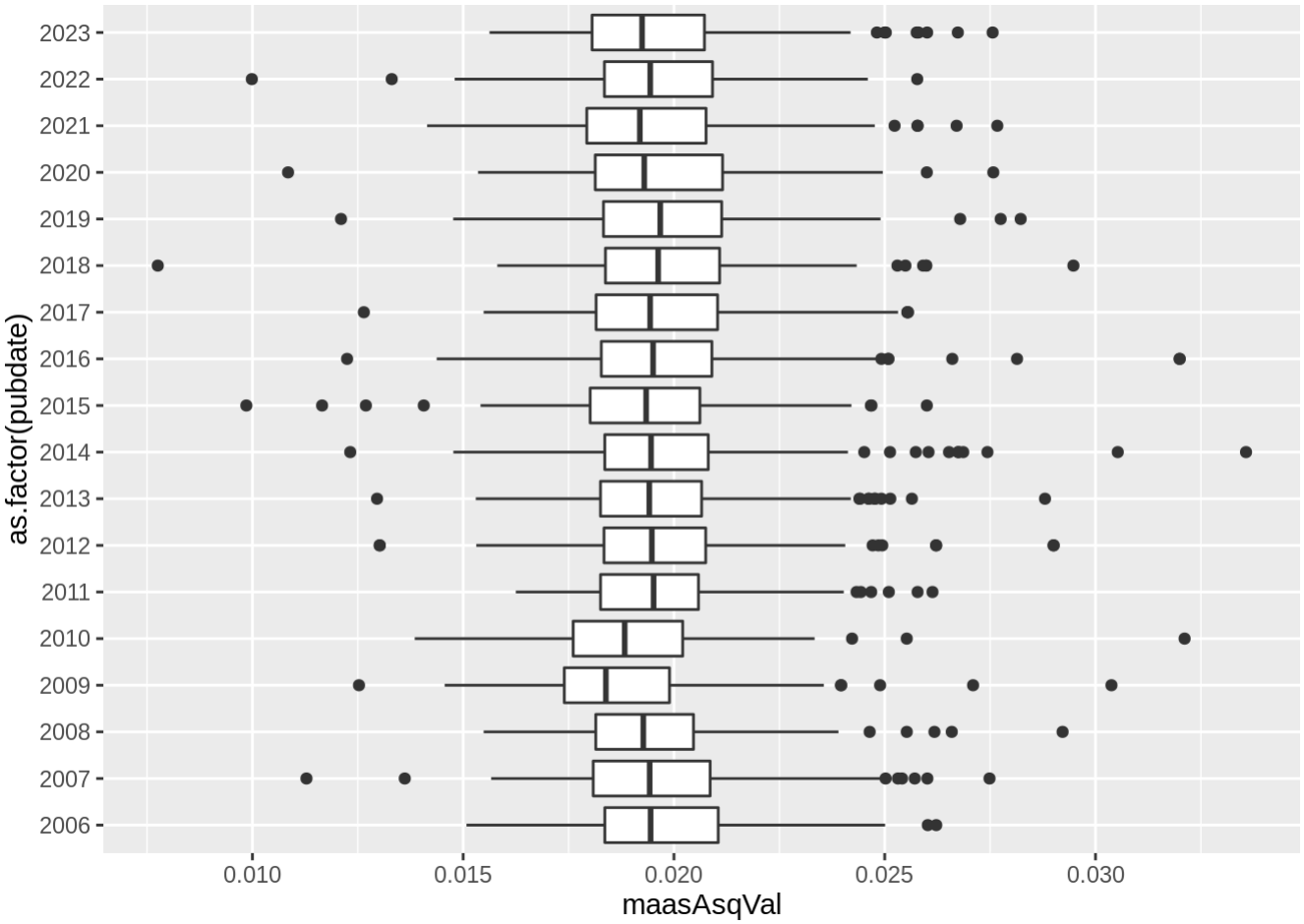

```
gf_boxplot(as.factor(pubdate) ~ typeTokenRatio, data = is.2)
```




```
gf_boxplot(as.factor(pubdate) ~ mtld, data = is.2)
```

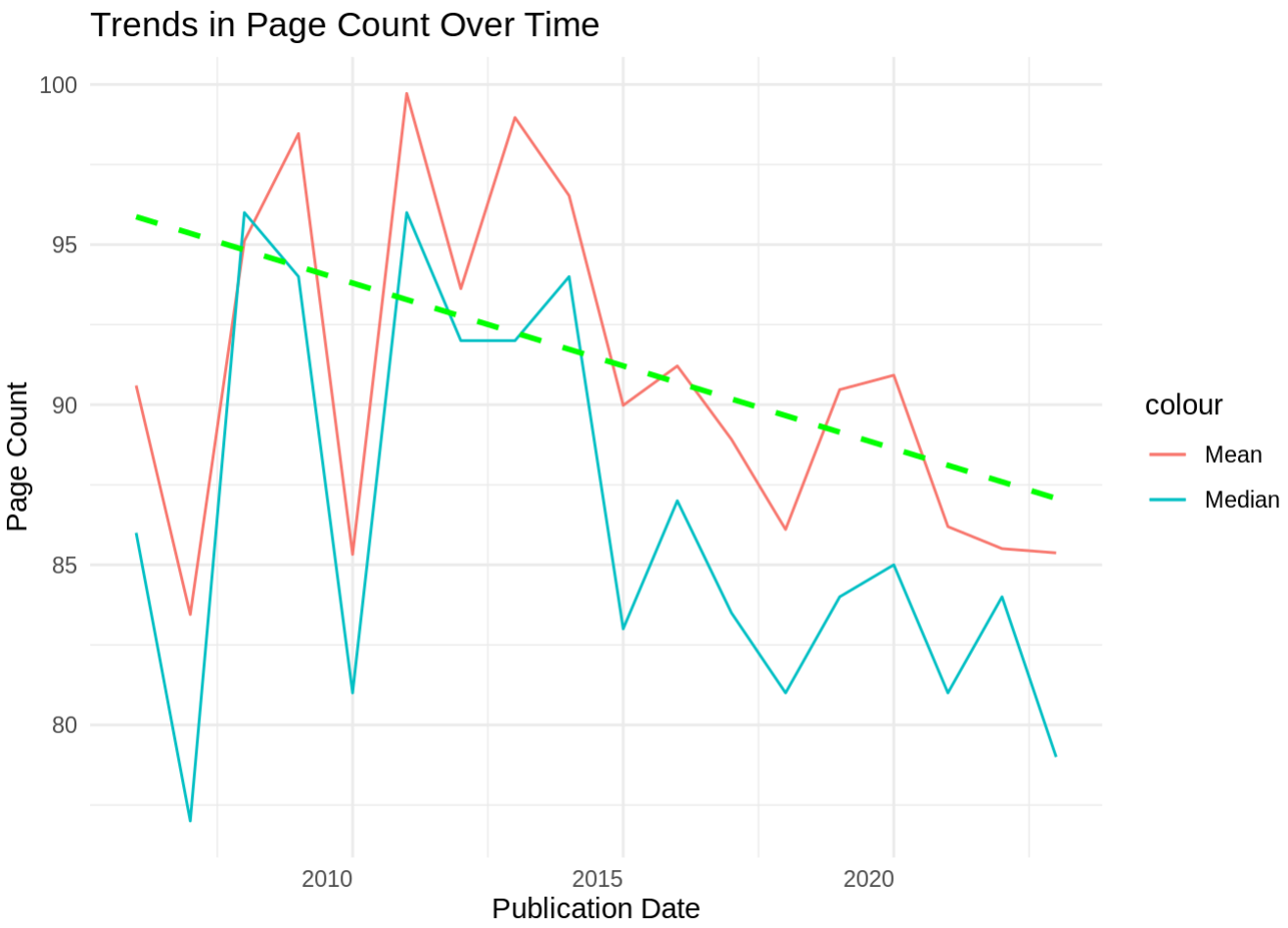



```
gf_boxplot(as.factor(pubdate) ~ maasAsqVal, data = is.2)
```




```
# this plot took painfully long to get working. ugh.
ggplot(is.2, aes(x = pubdate, y = pagec)) +
  stat_summary(fun = "mean", geom = "line",  aes(color = "Mean"),  show.legend = TRUE) +
  stat_summary(fun = "median", geom = "line",  aes(color = "Median"),  show.legend = TRUE) +
  geom_smooth(method = "lm", se = FALSE, color = "green", linetype = "dashed") +
  labs(title = "Trends in Page Count Over Time",
       x = "Publication Date",
       y = "Page Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = , hjust = 1))
```

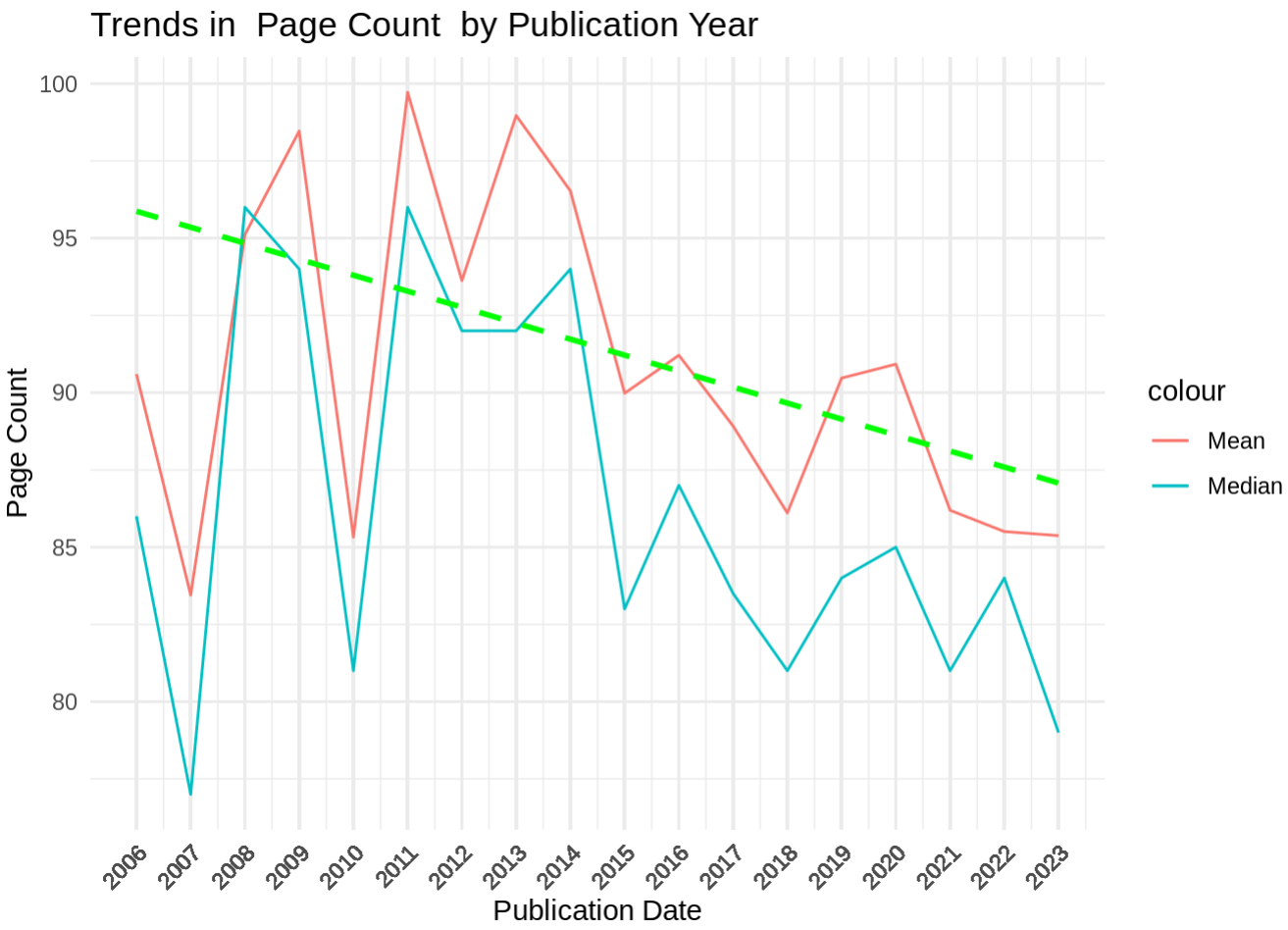
```
## `geom_smooth()` using formula 'y ~ x'
```



Playing Around with Trends More

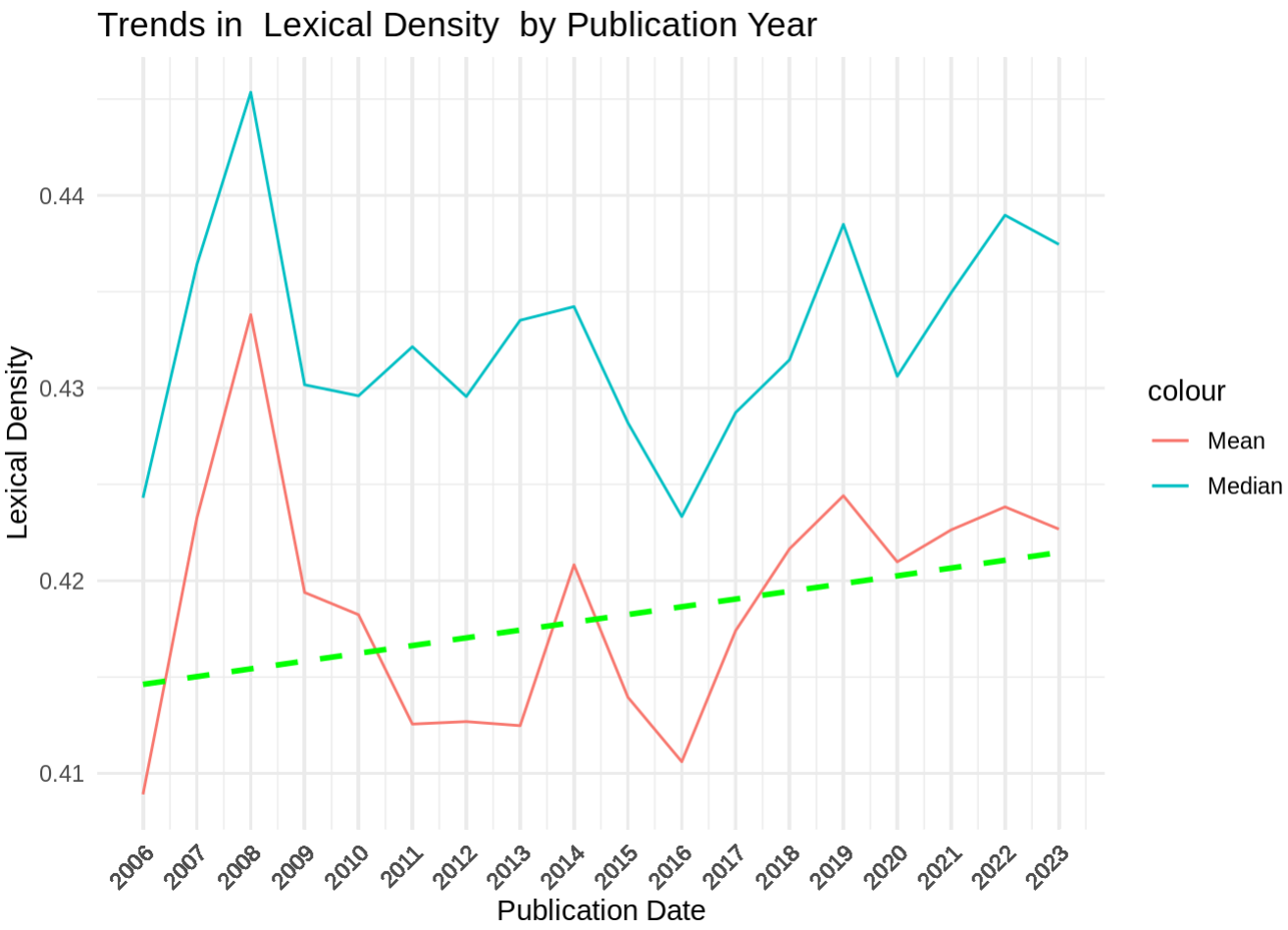
```
# functionalize this for easy reuse
makePublicationPlot <- function(yval, title) {
 ggplot(is.2, aes(x = pubdate, y = yval)) +
  stat_summary(fun = "mean", geom = "line",  aes(color = "Mean"),  show.legend = TRUE) +
  stat_summary(fun = "median", geom = "line",  aes(color = "Median"),  show.legend = TRUE) +
  geom_smooth(method = "lm", se = FALSE, color = "green", linetype = "dashed") +
  labs(title = paste("Trends in ",title," by Publication Year"),
       x = "Publication Date",
       y = title) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
    scale_x_continuous(breaks = is.2$pubdate, labels = is.2$pubdate)
}
# this plot took painfully long to get working. ugh.

makePublicationPlot(is.2$pagec, "Page Count")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



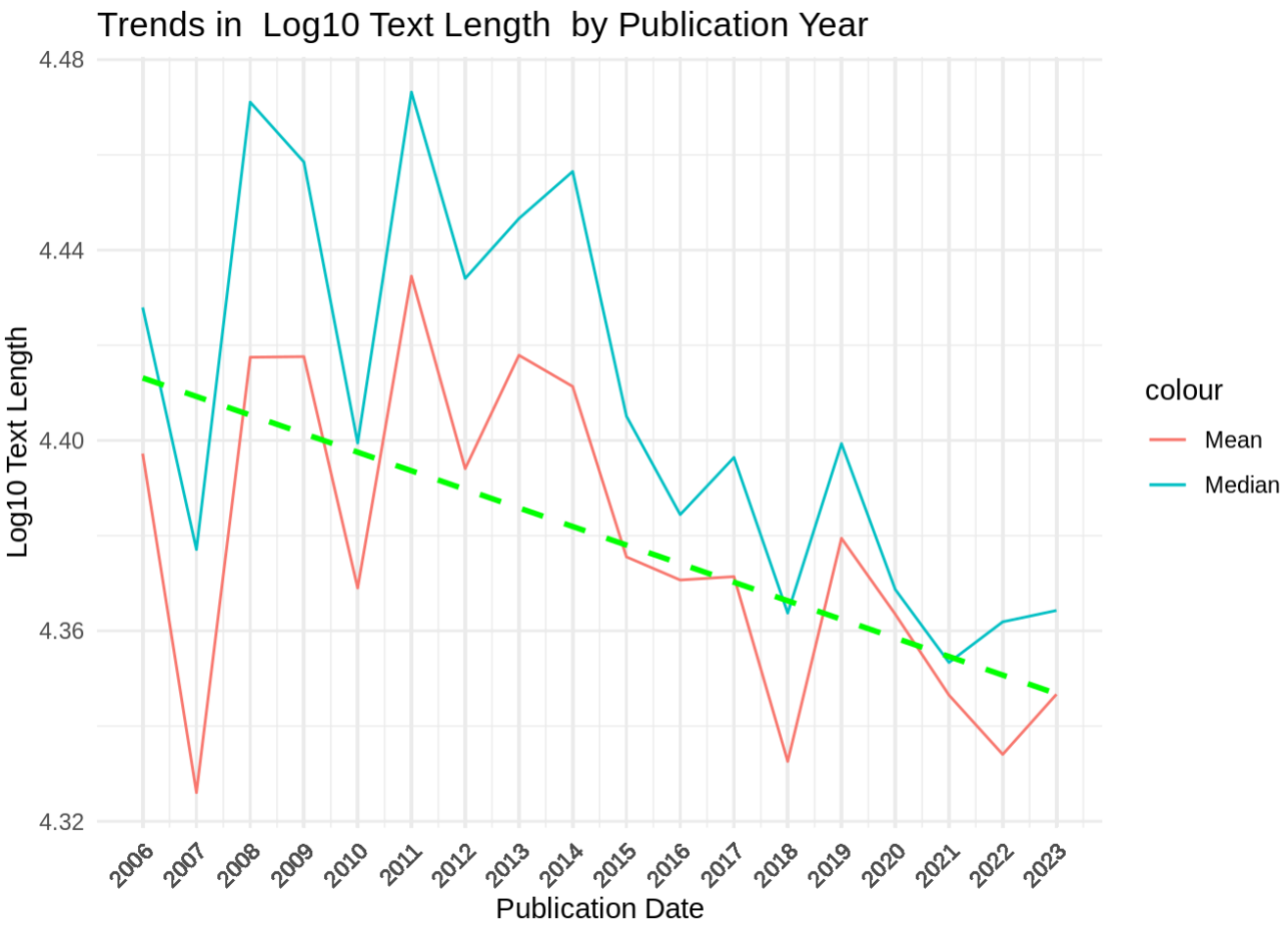

```
makePublicationPlot(is.2$lexDensity, "Lexical Density")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



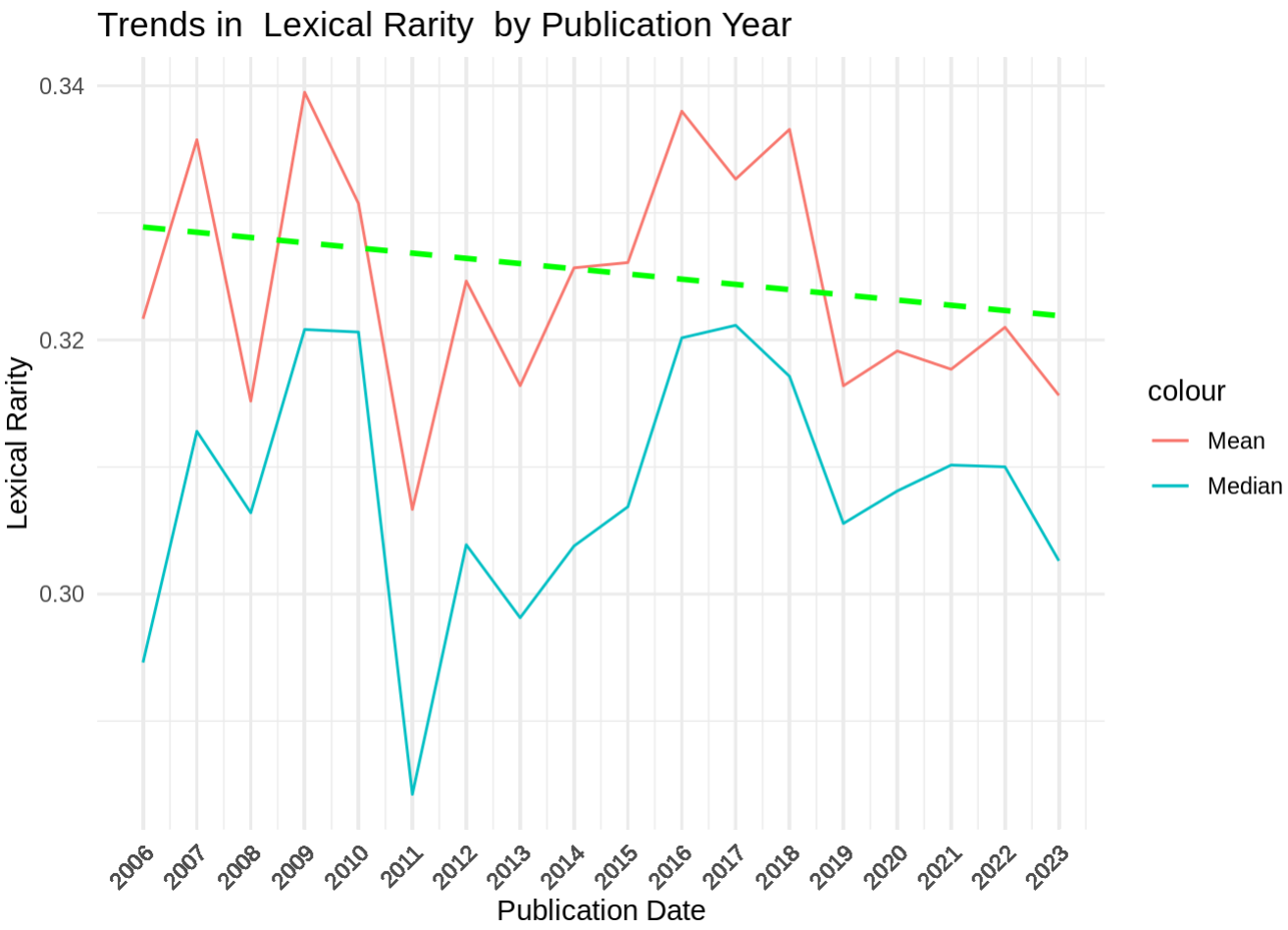

```
makePublicationPlot(is.2$log10TextLen, "Log10 Text Length")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



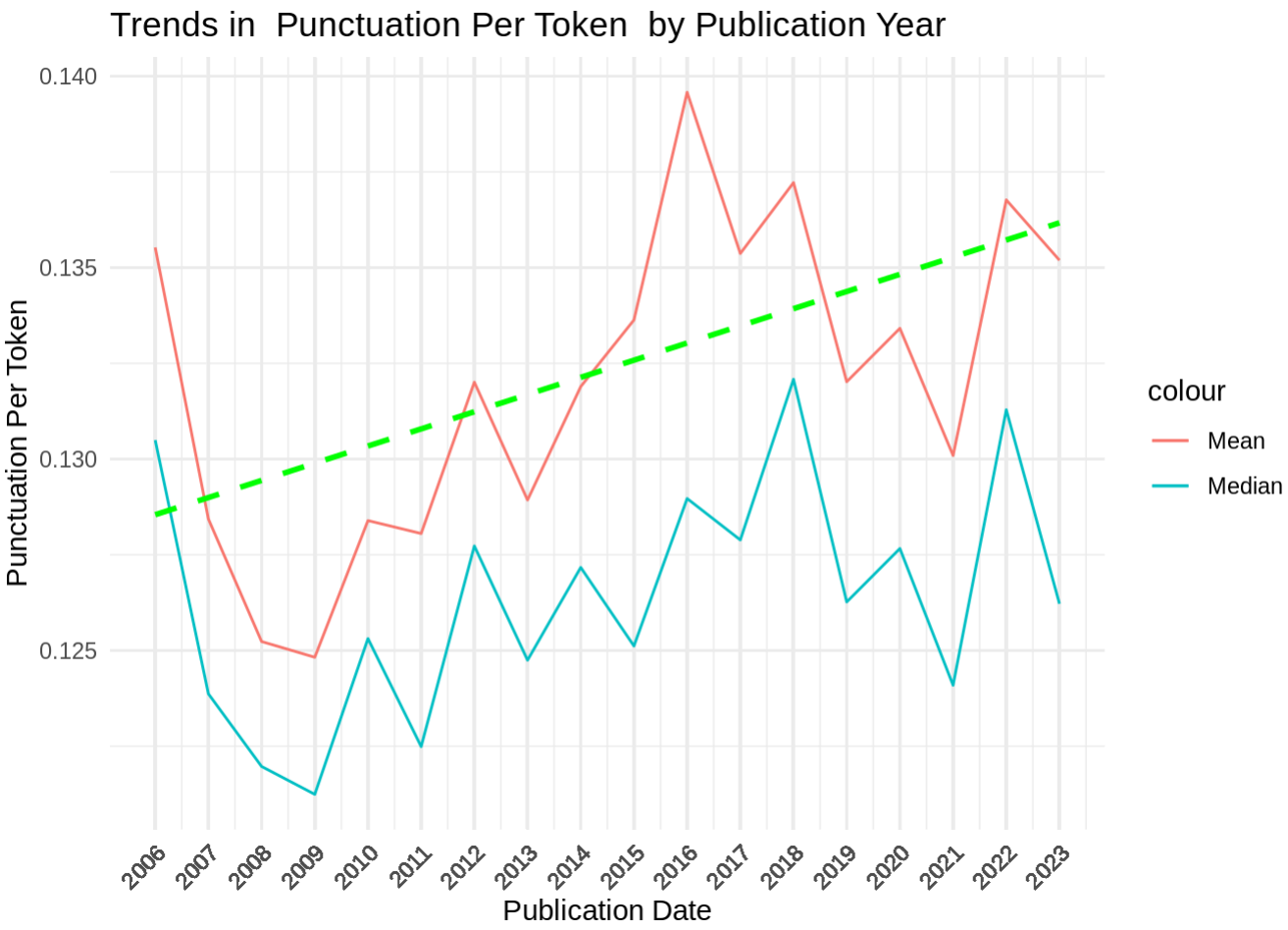

```
makePublicationPlot(is.2$lexRarity, "Lexical Rarity")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

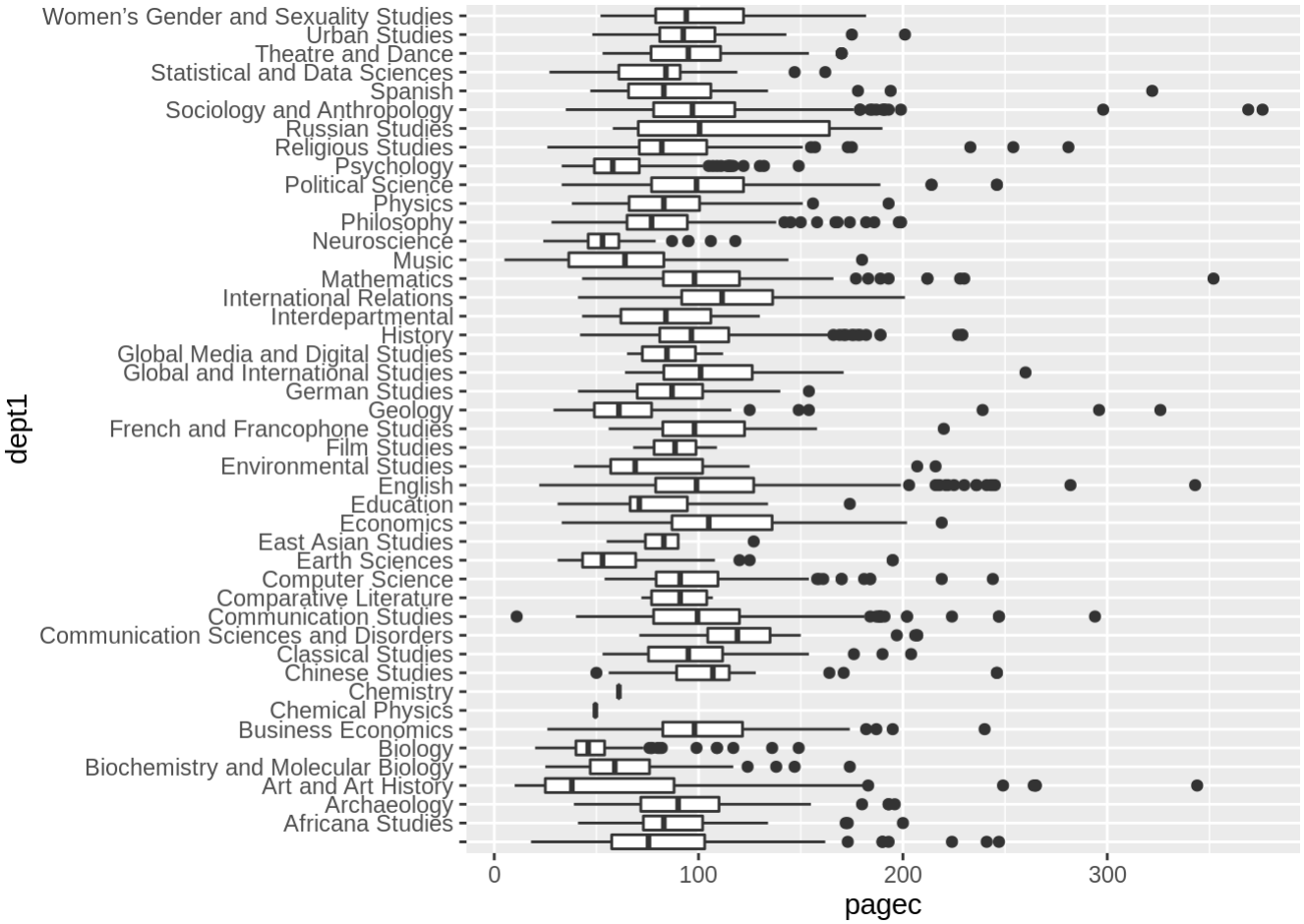



```
makePublicationPlot(is.2$punctPerTok, "Punctuation Per Token")
```

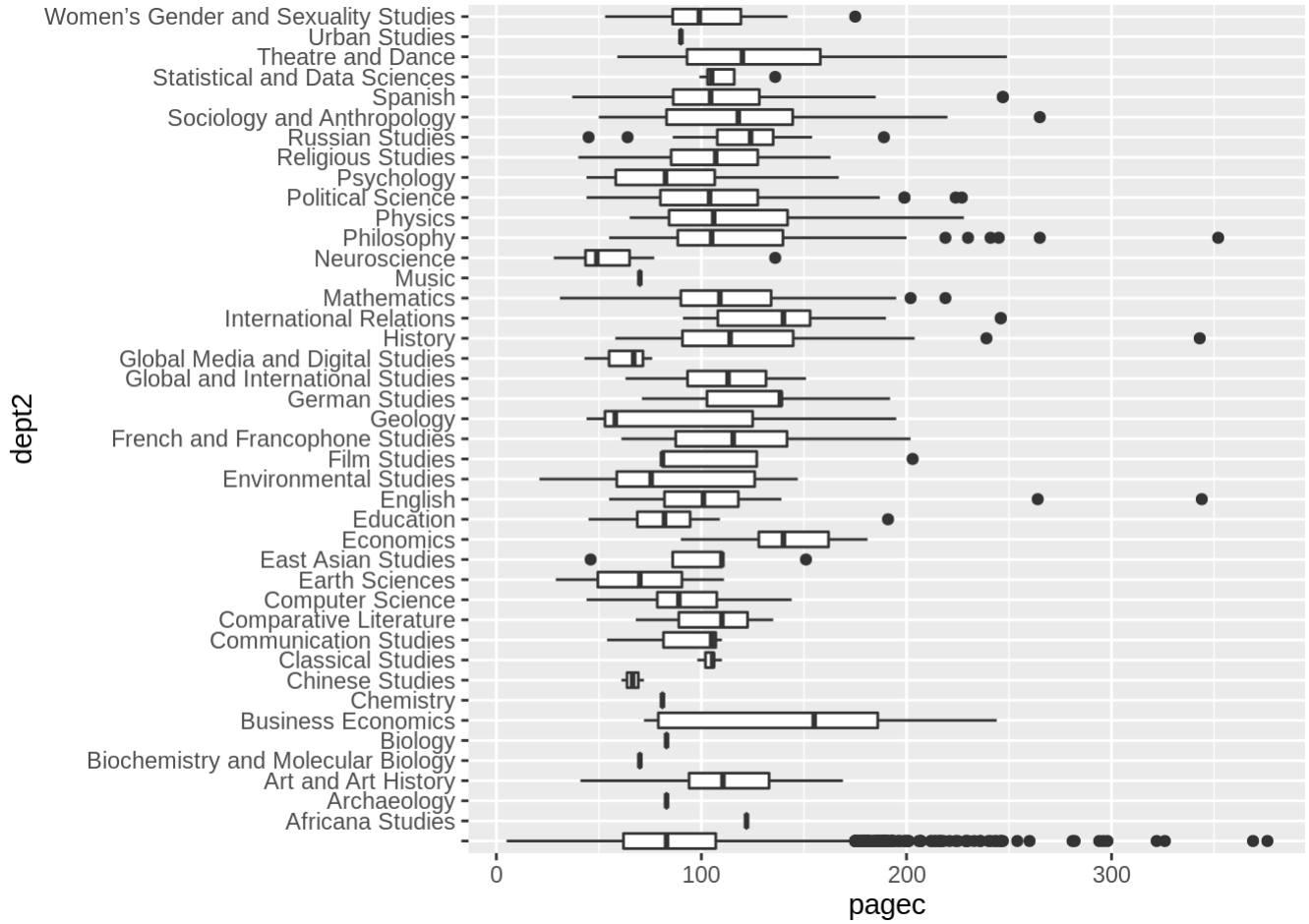
```
## `geom_smooth()` using formula 'y ~ x'
```



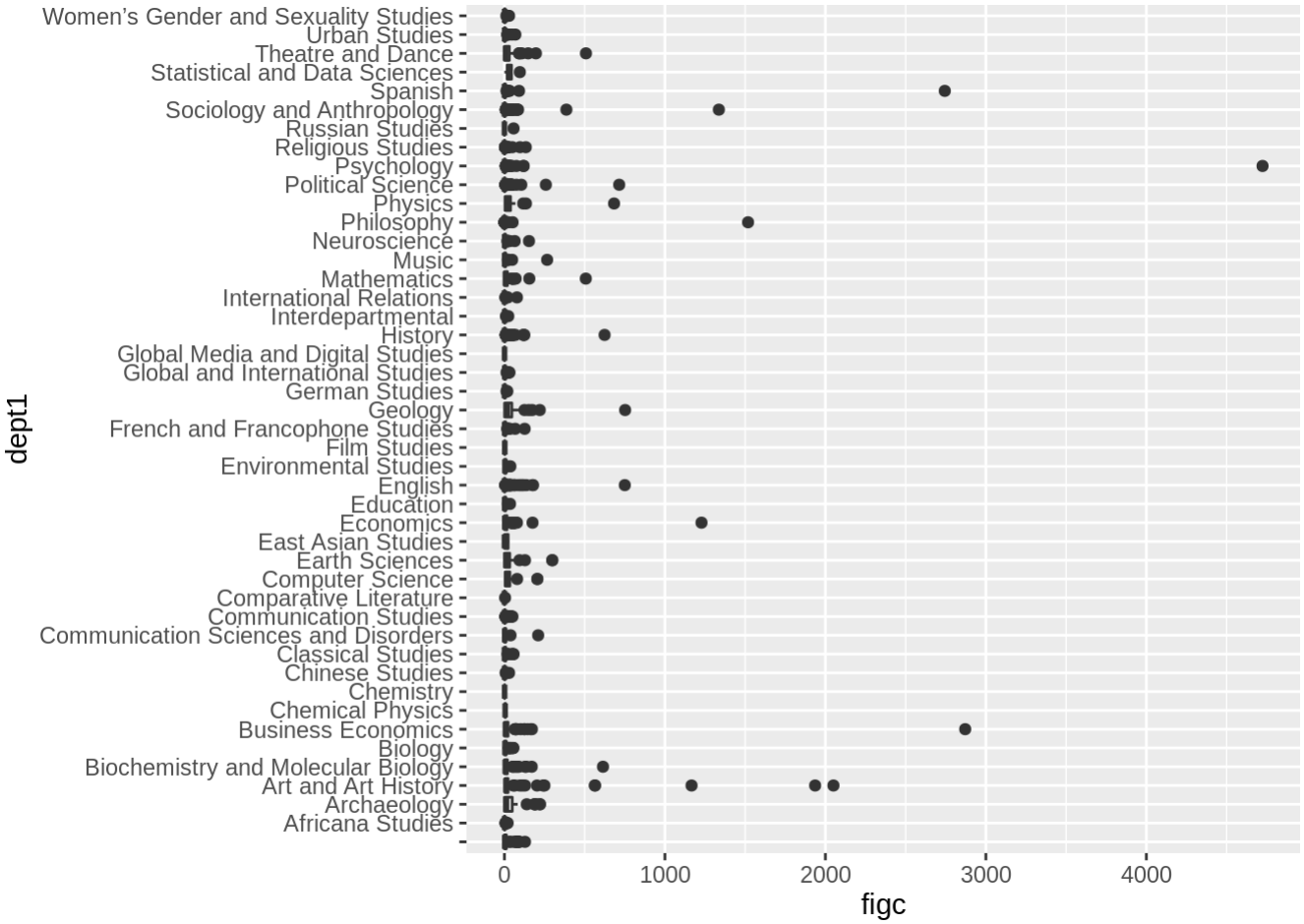

```
gf_boxplot(dept1 ~ pagec, data = is.2)
```



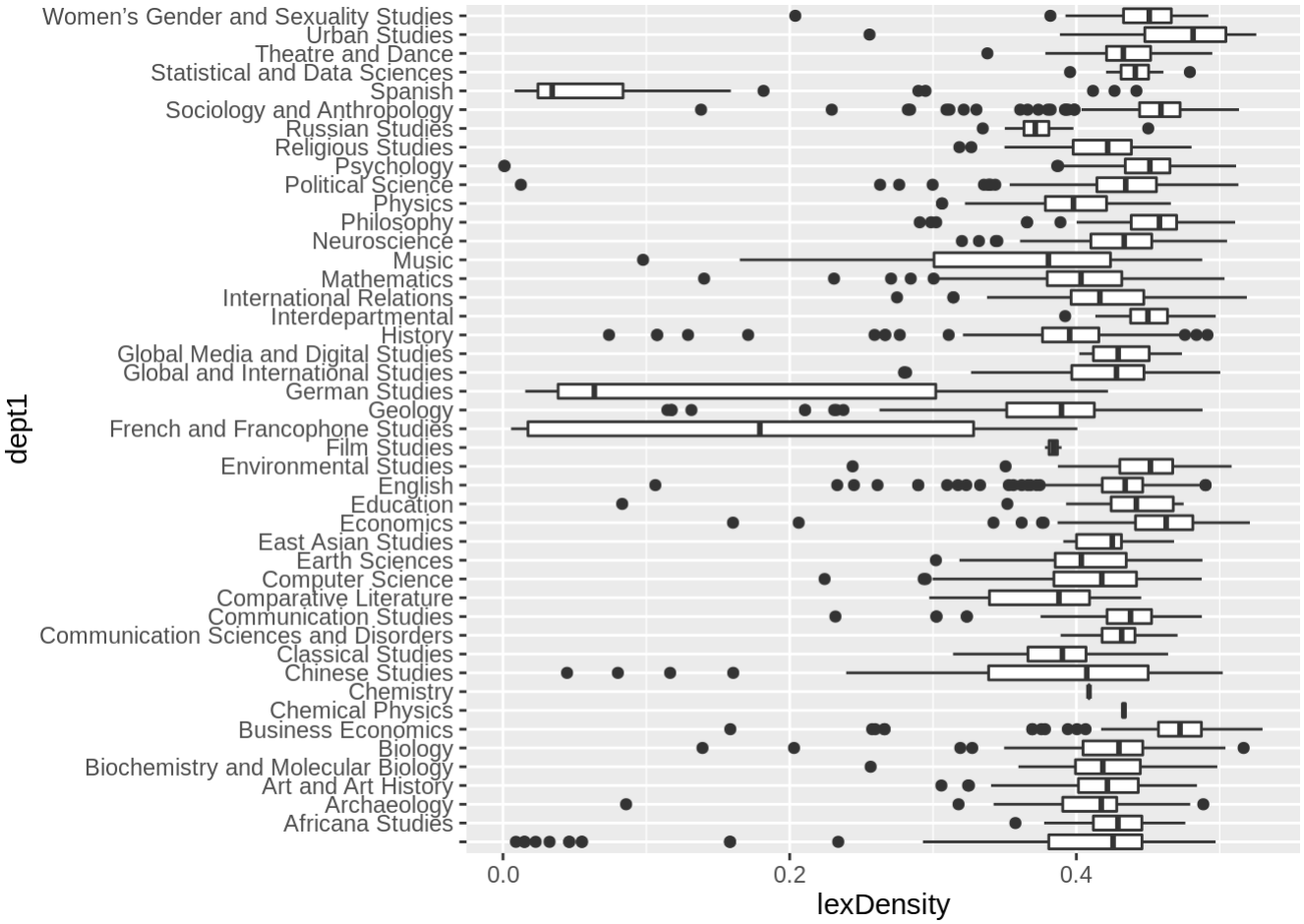

```
gf_boxplot(dept2 ~ pagec, data = is.2)
```



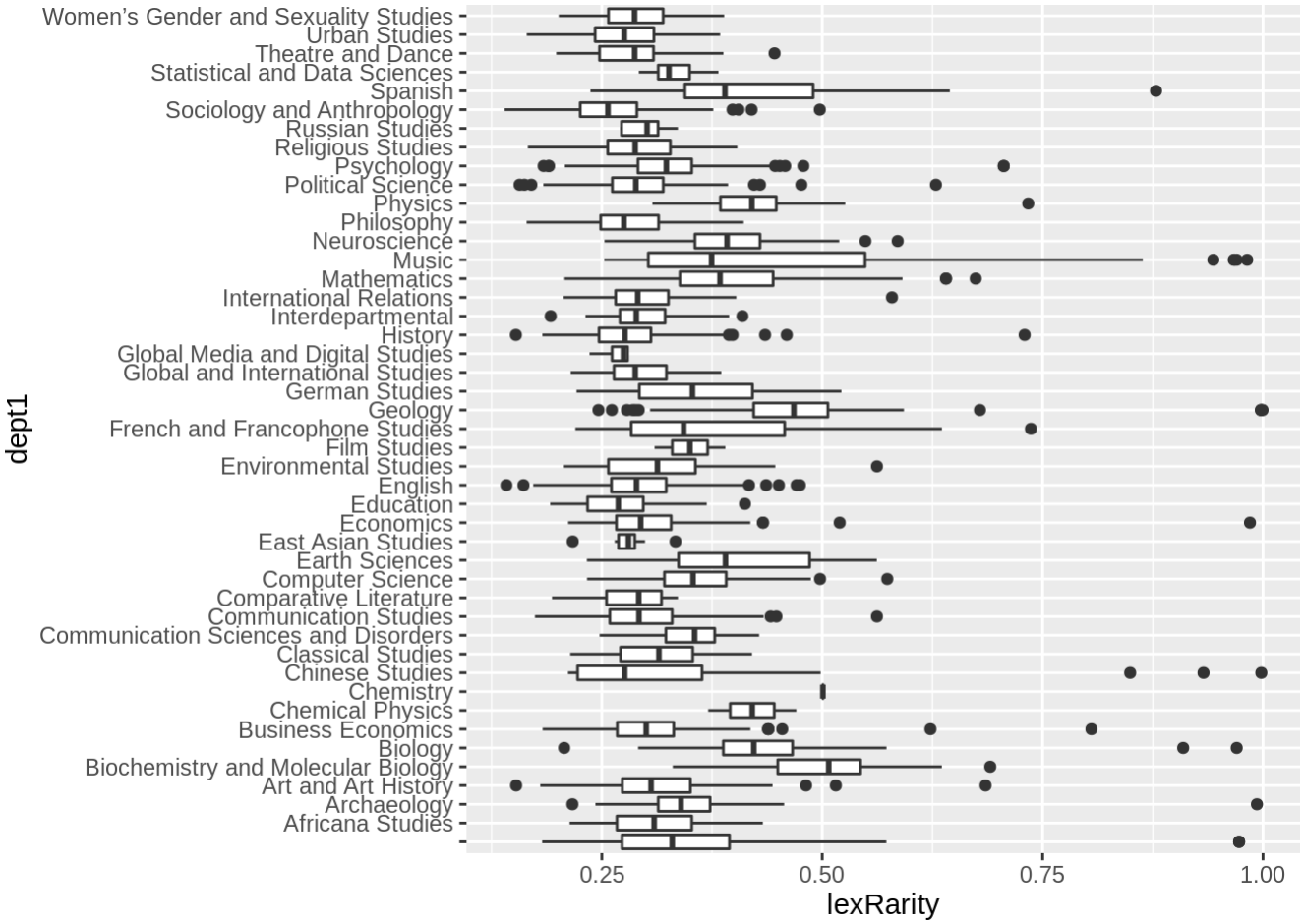

```
gf_boxplot(dept1 ~ figc, data = is.2)
```



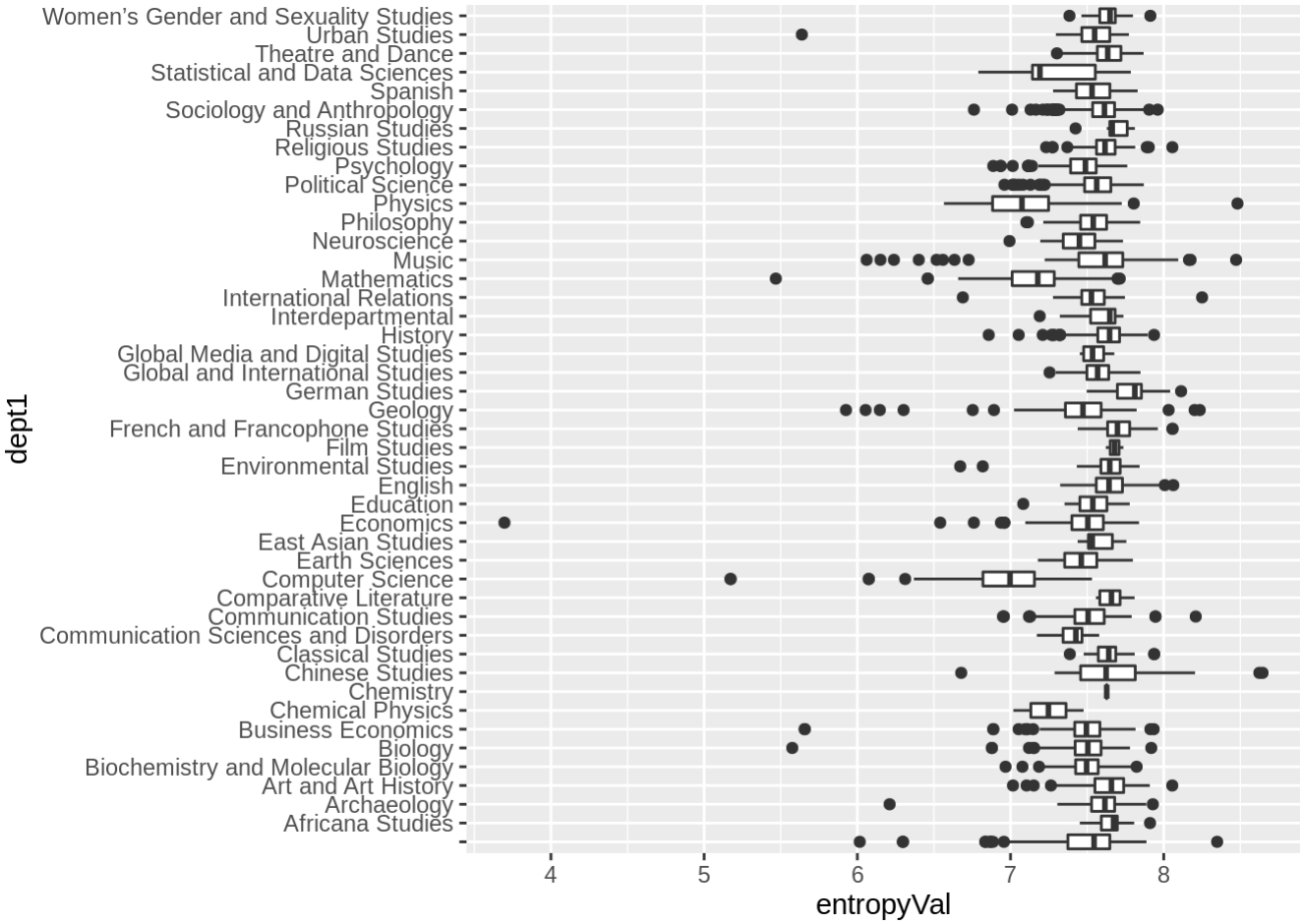

```
gf_boxplot(dept1 ~ lexDensity, data = is.2)
```



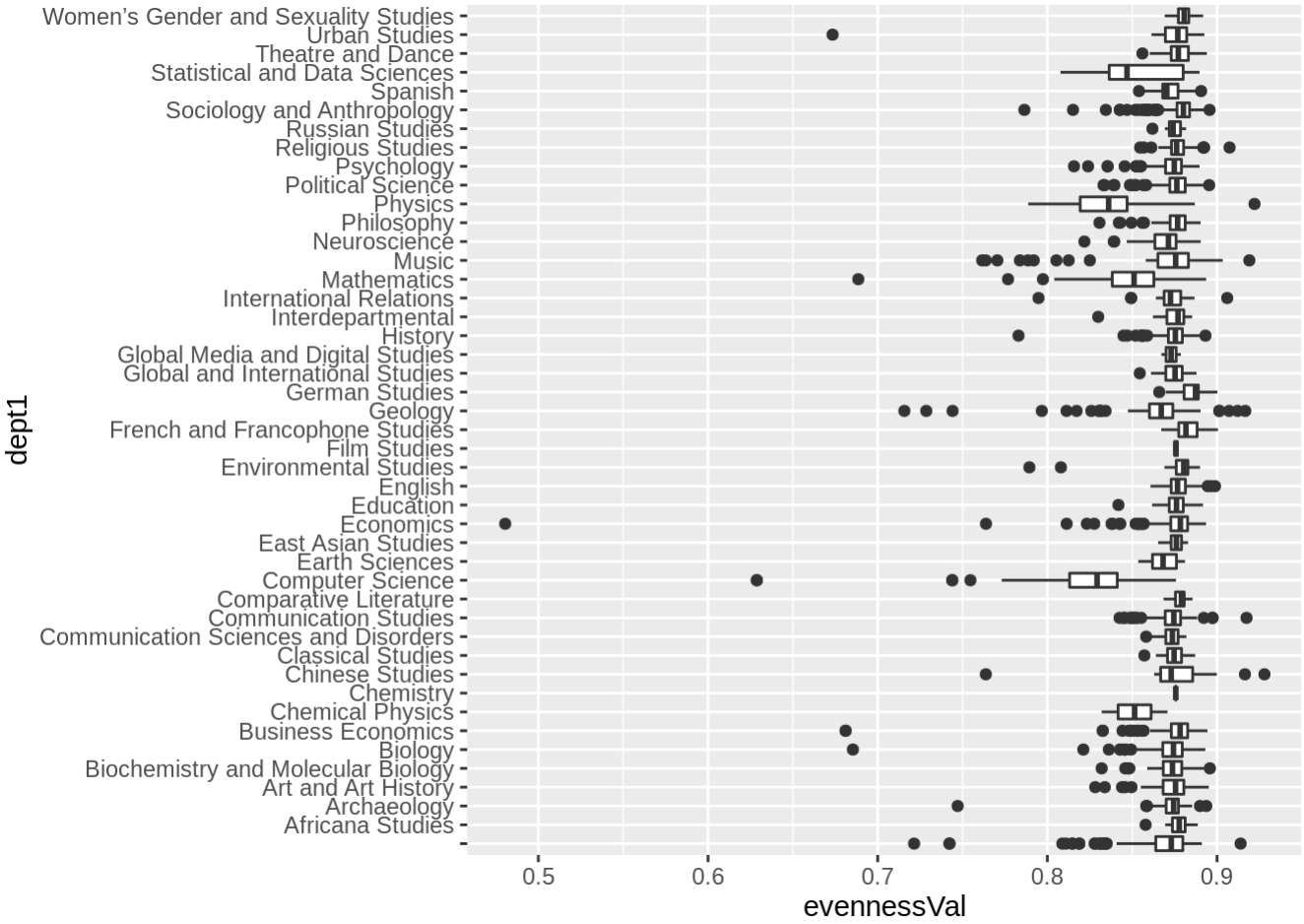

```
gf_boxplot(dept1 ~ lexRarity, data = is.2)
```



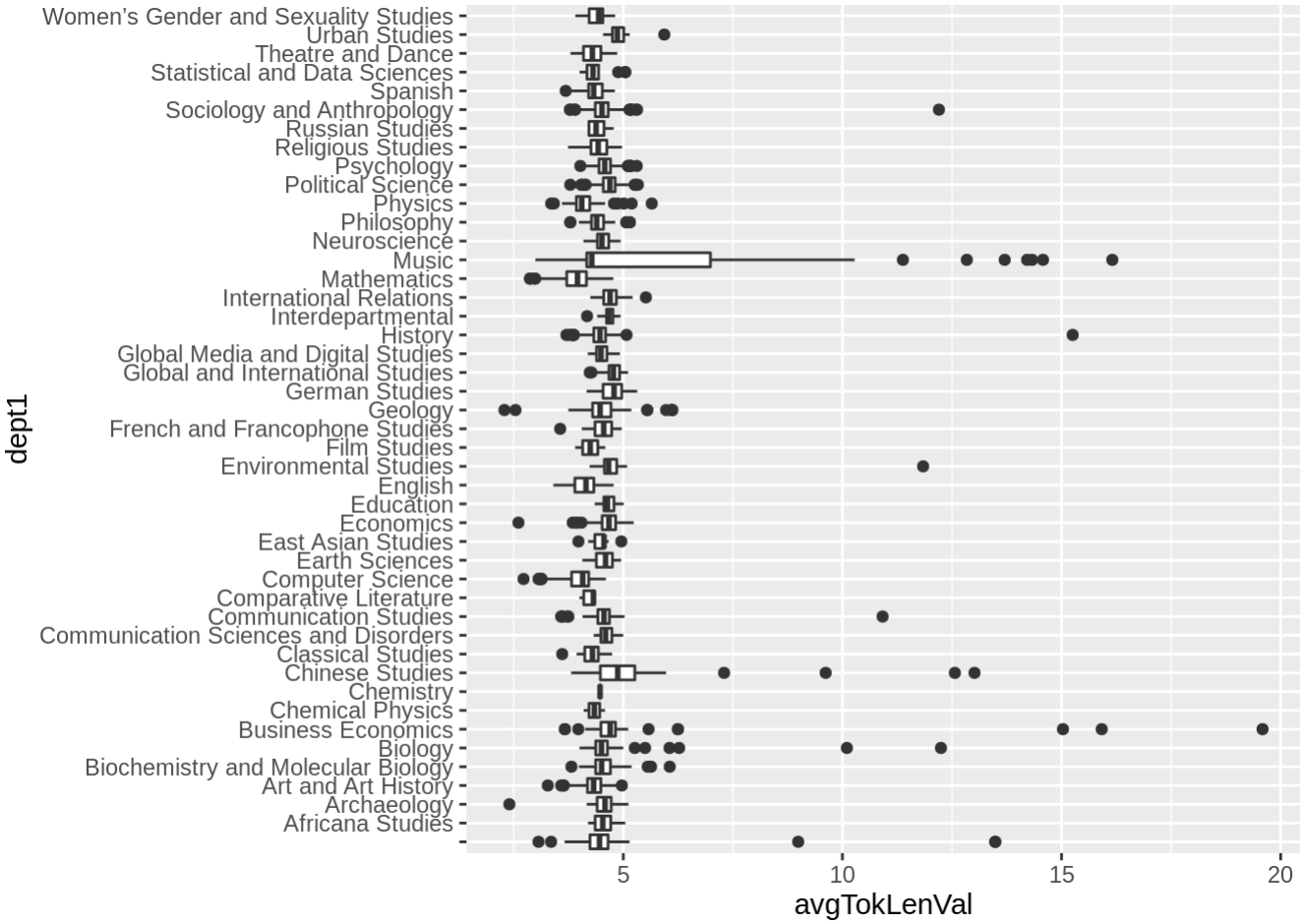

```
gf_boxplot(dept1 ~ entropyVal, data = is.2)
```



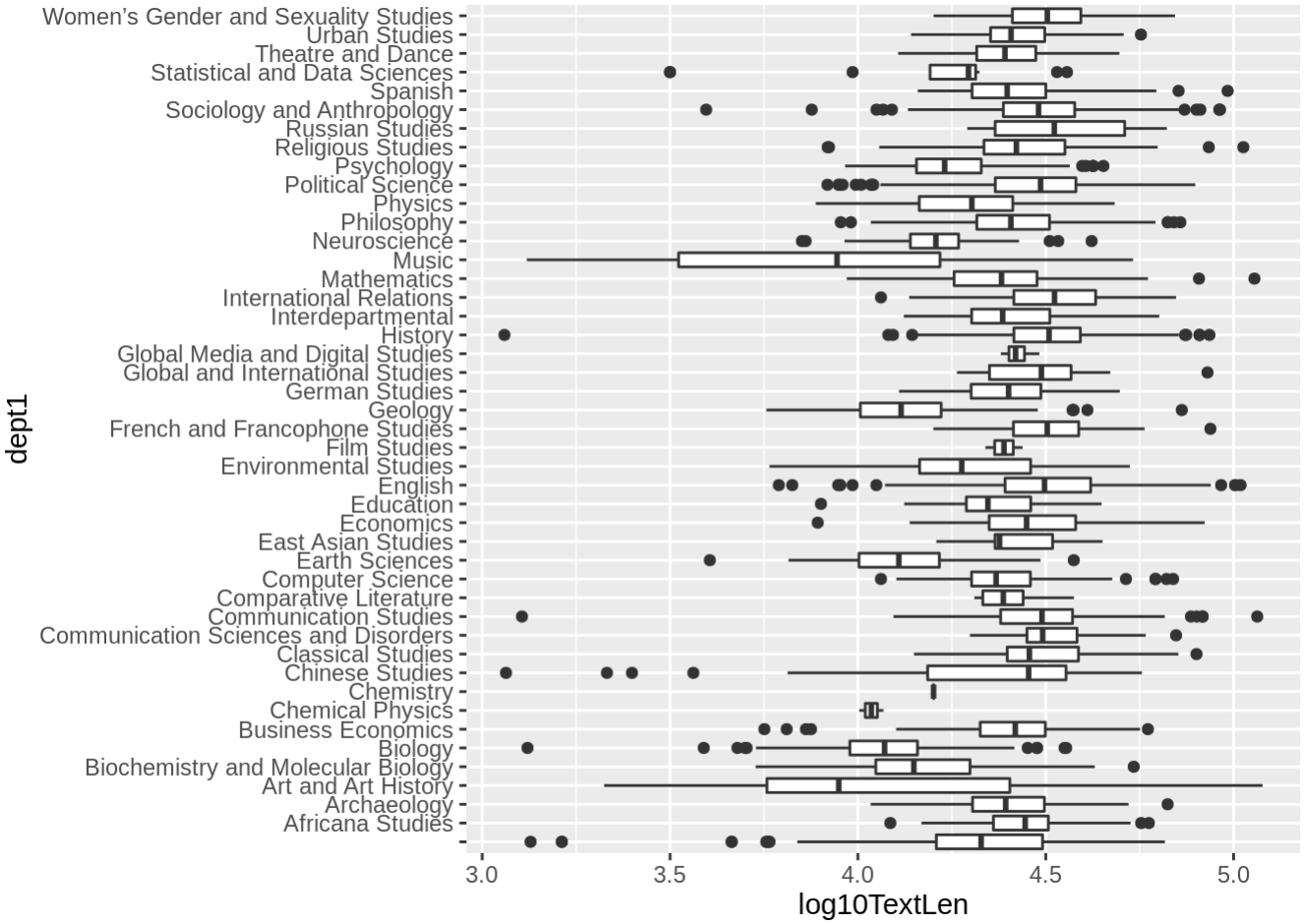

```
gf_boxplot(dept1 ~ evennessVal, data = is.2)
```



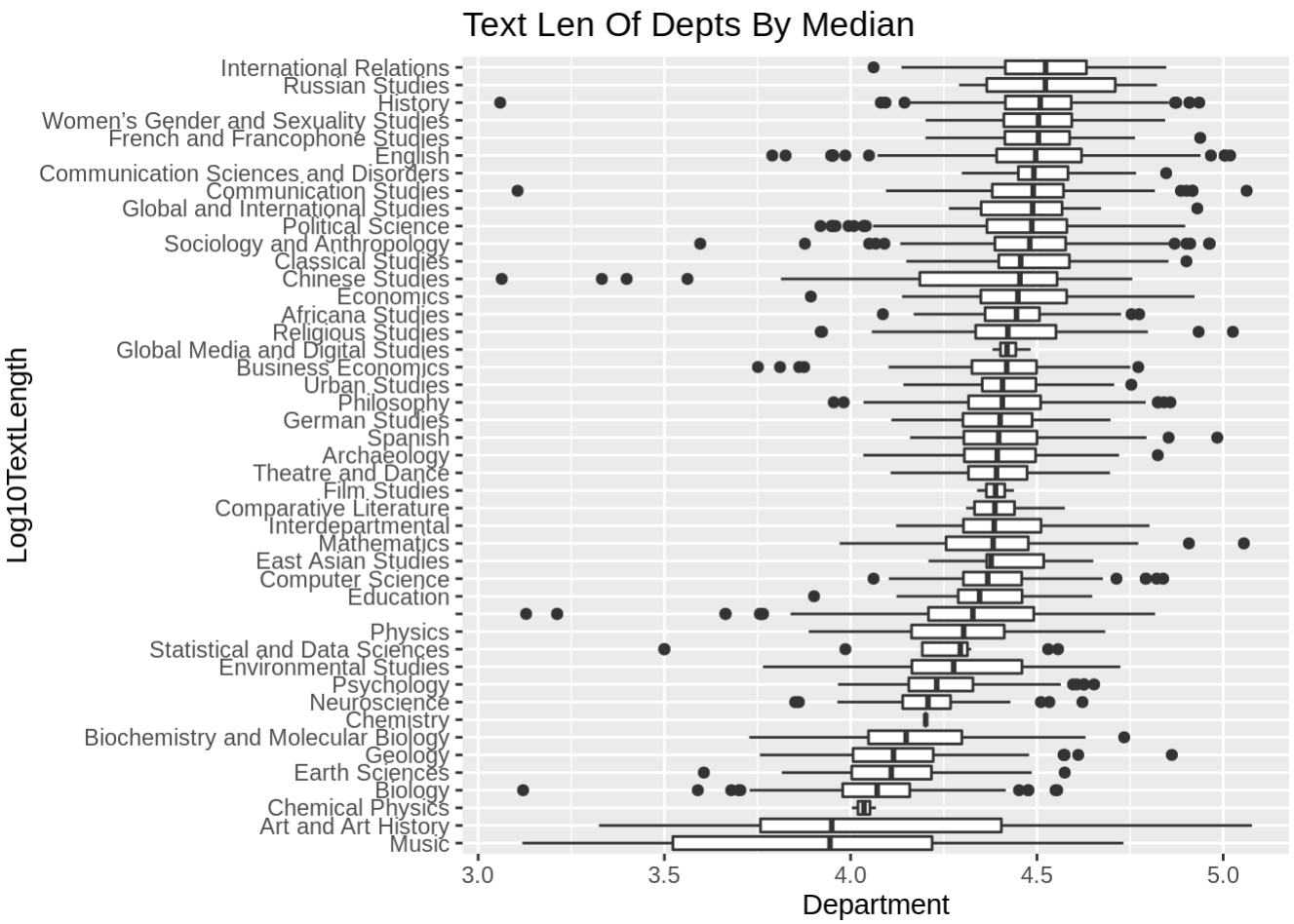

```
gf_boxplot(dept1 ~ avgTokLenVal, data = is.2)
```




```
gf_boxplot(dept1 ~ log10TextLen, data = is.2)
```



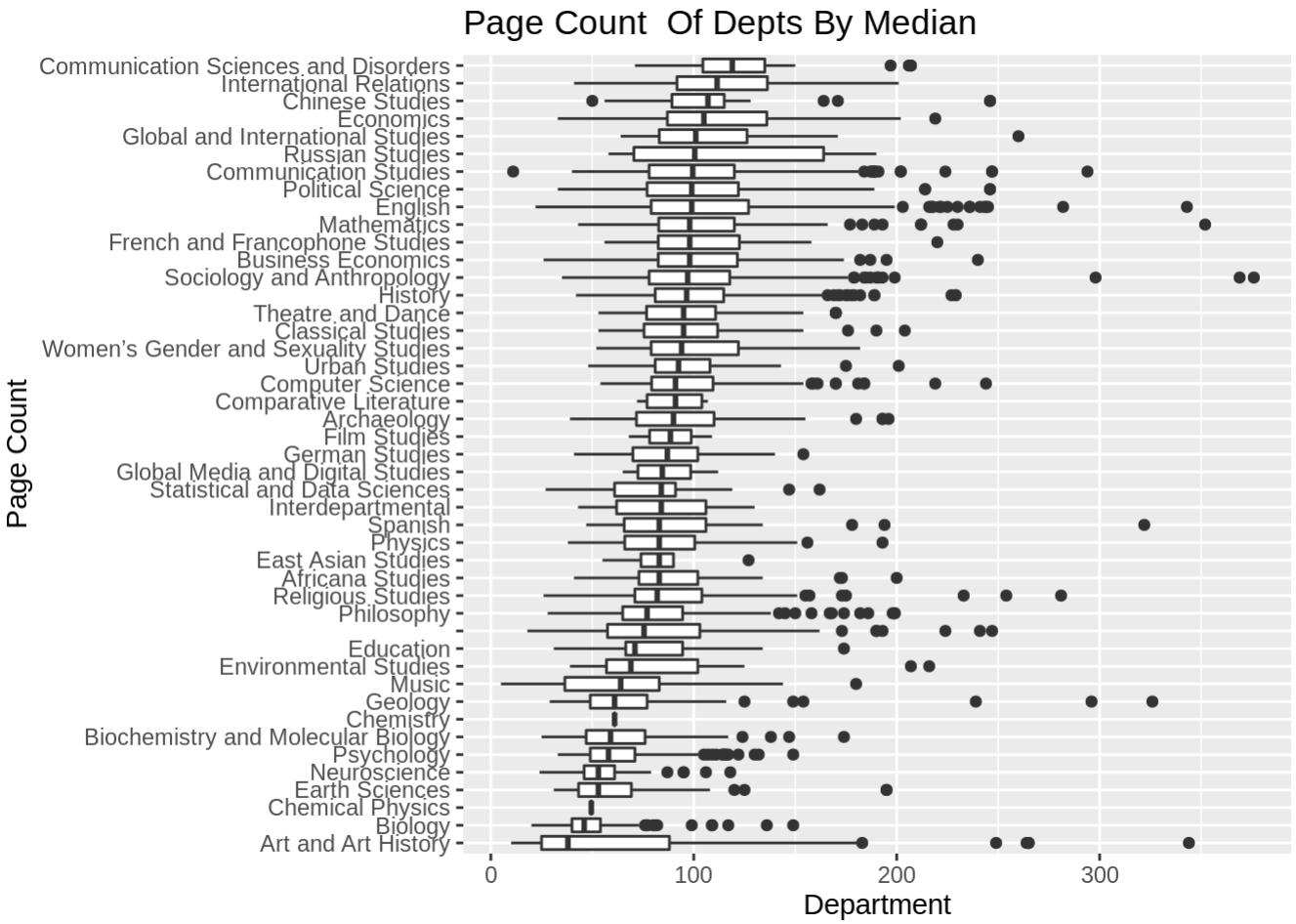

```
# playing around with reorderings
ggplot(is.2, aes(y = reorder(dept1, log10TextLen, FUN = median), x = log10TextLen)) +
  geom_boxplot() +
  labs(title = "Text Len Of Depts By Median", x = "Department", y = "Log10TextLength")
```



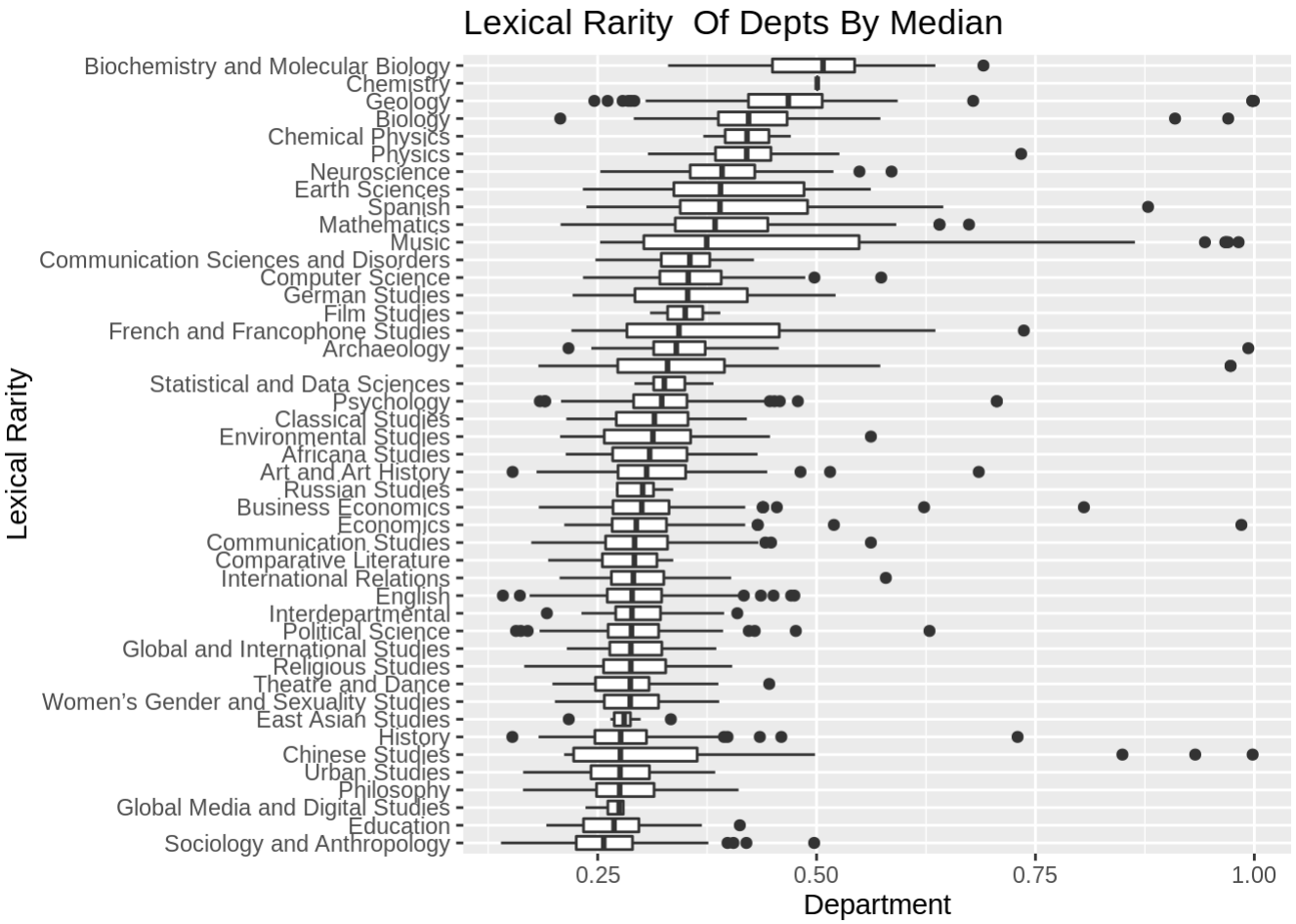

```
#functional-ize it
medSortBP <- function(sortVal, title) {
 ggplot(is.2, aes(y = reorder(dept1, sortVal, FUN = median), x = sortVal)) +
  geom_boxplot() +
  labs(title = paste(title, " Of Depts By Median"), x = "Department", y = title)
}
```

By Department Boxplot Visualizations

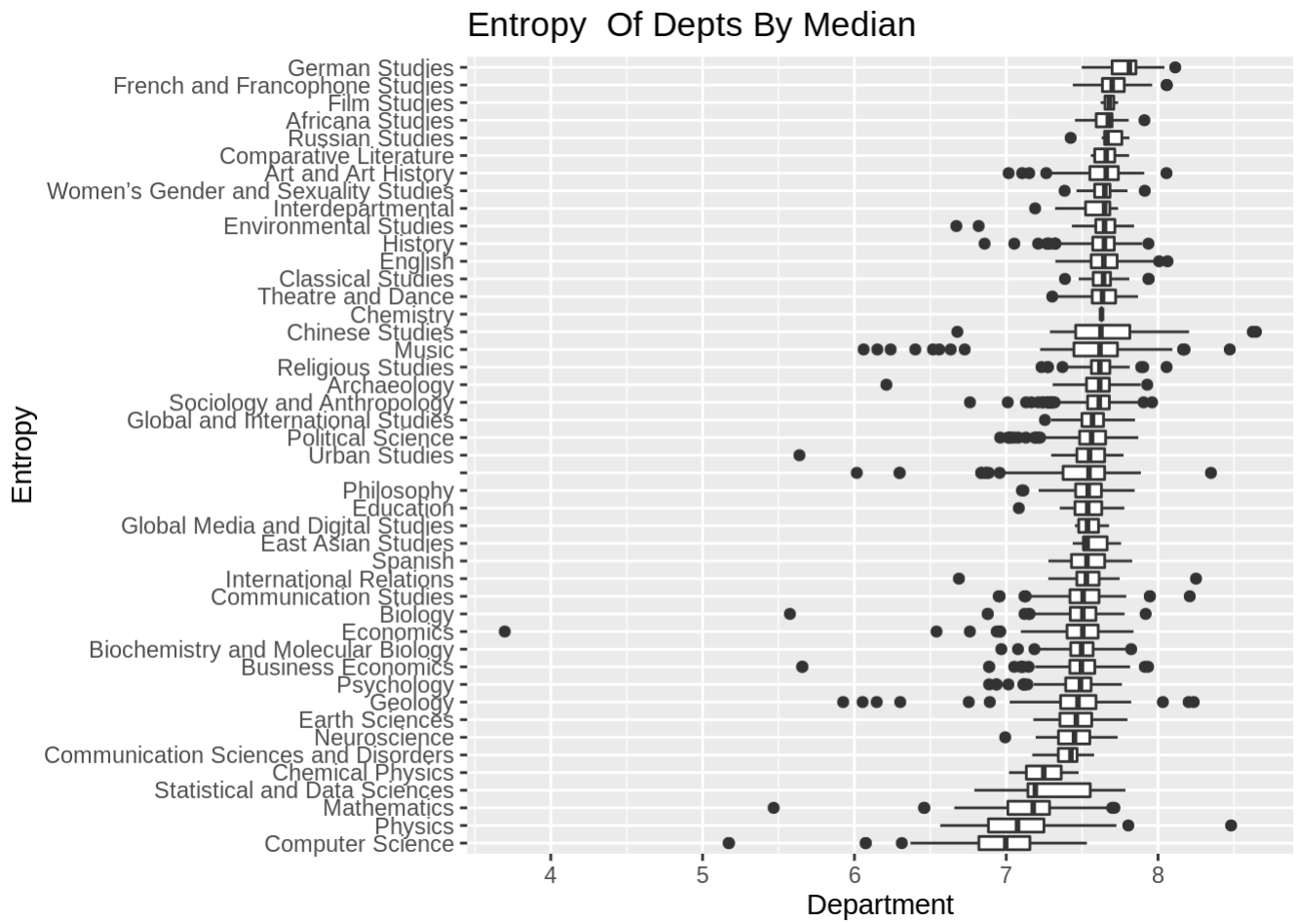
```
medSortBP(is.2$pagec, "Page Count")
```



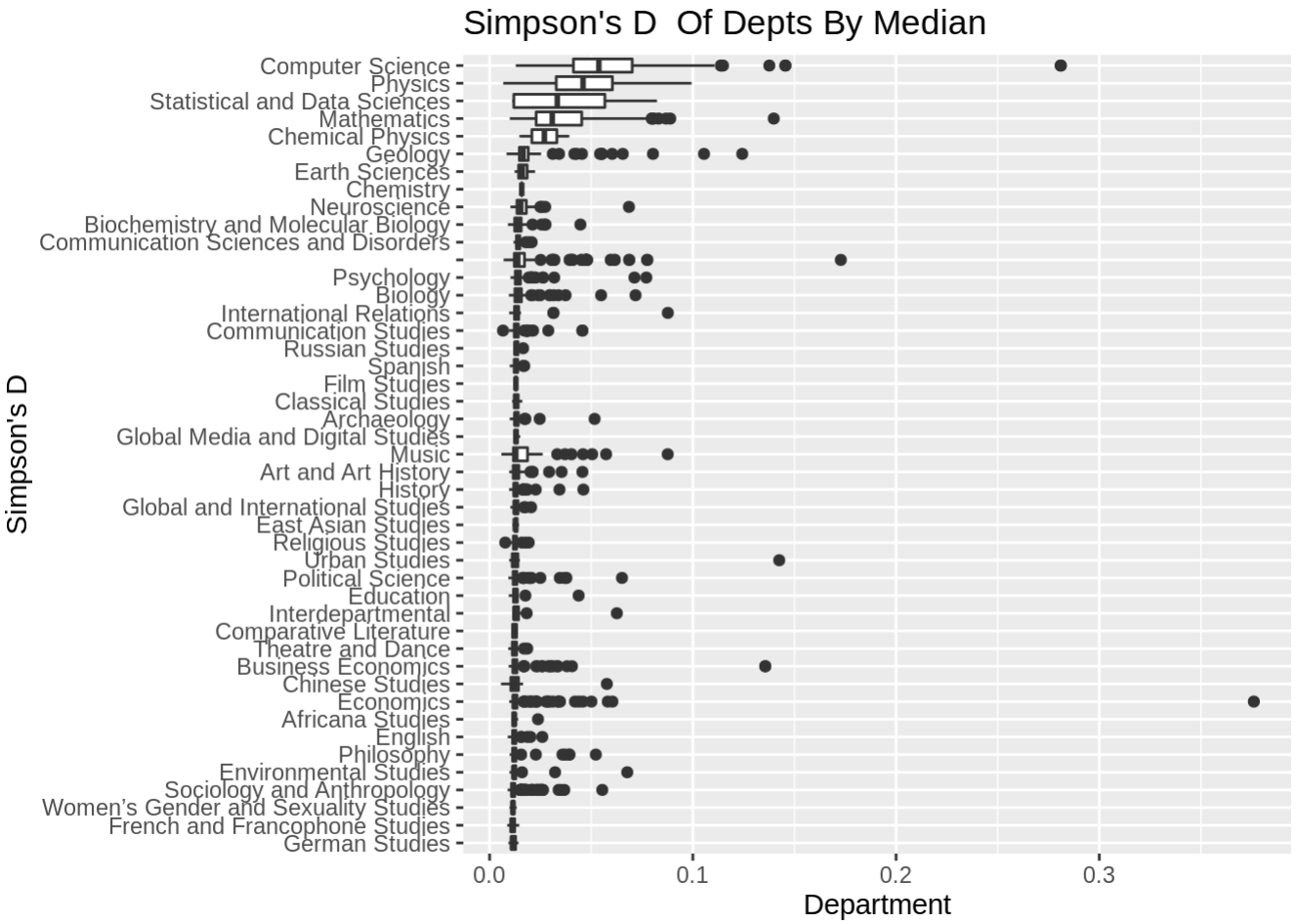

```
medSortBP(is.2$lexRarity, "Lexical Rarity")
```



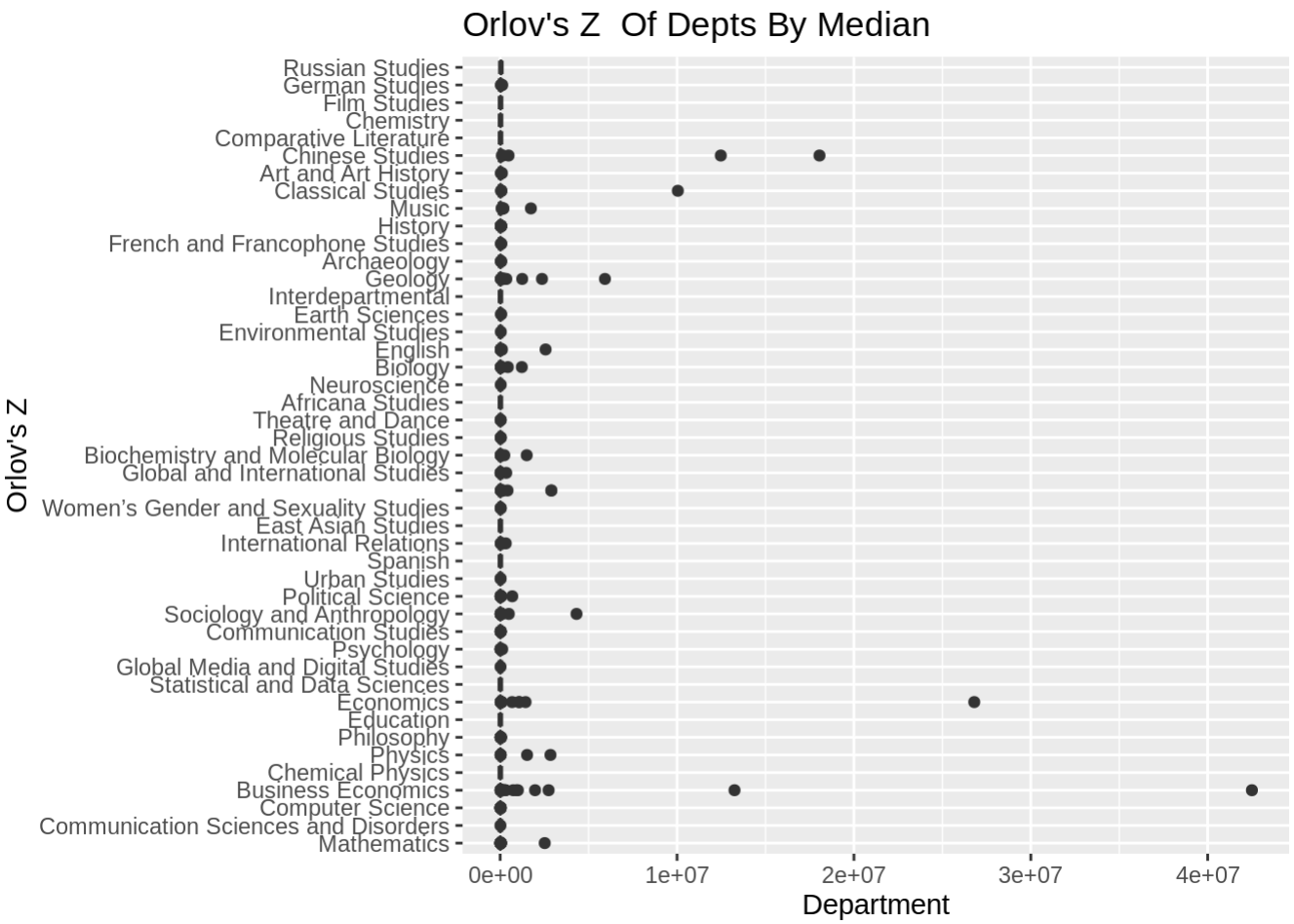

```
medSortBP(is.2$entropyVal, "Entropy")
```



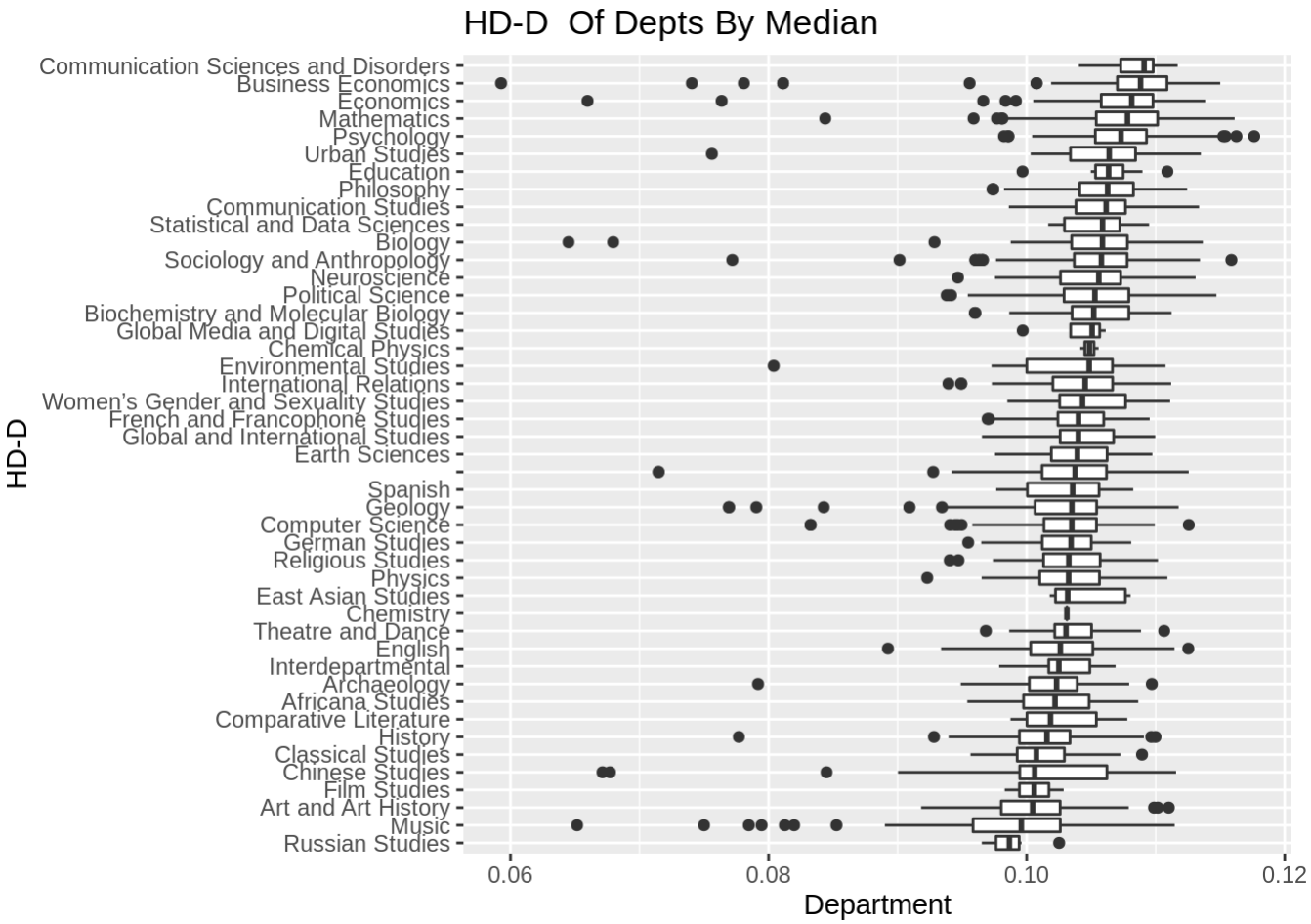

```
medSortBP(is.2$simpsonDVal, "Simpson's D")
```



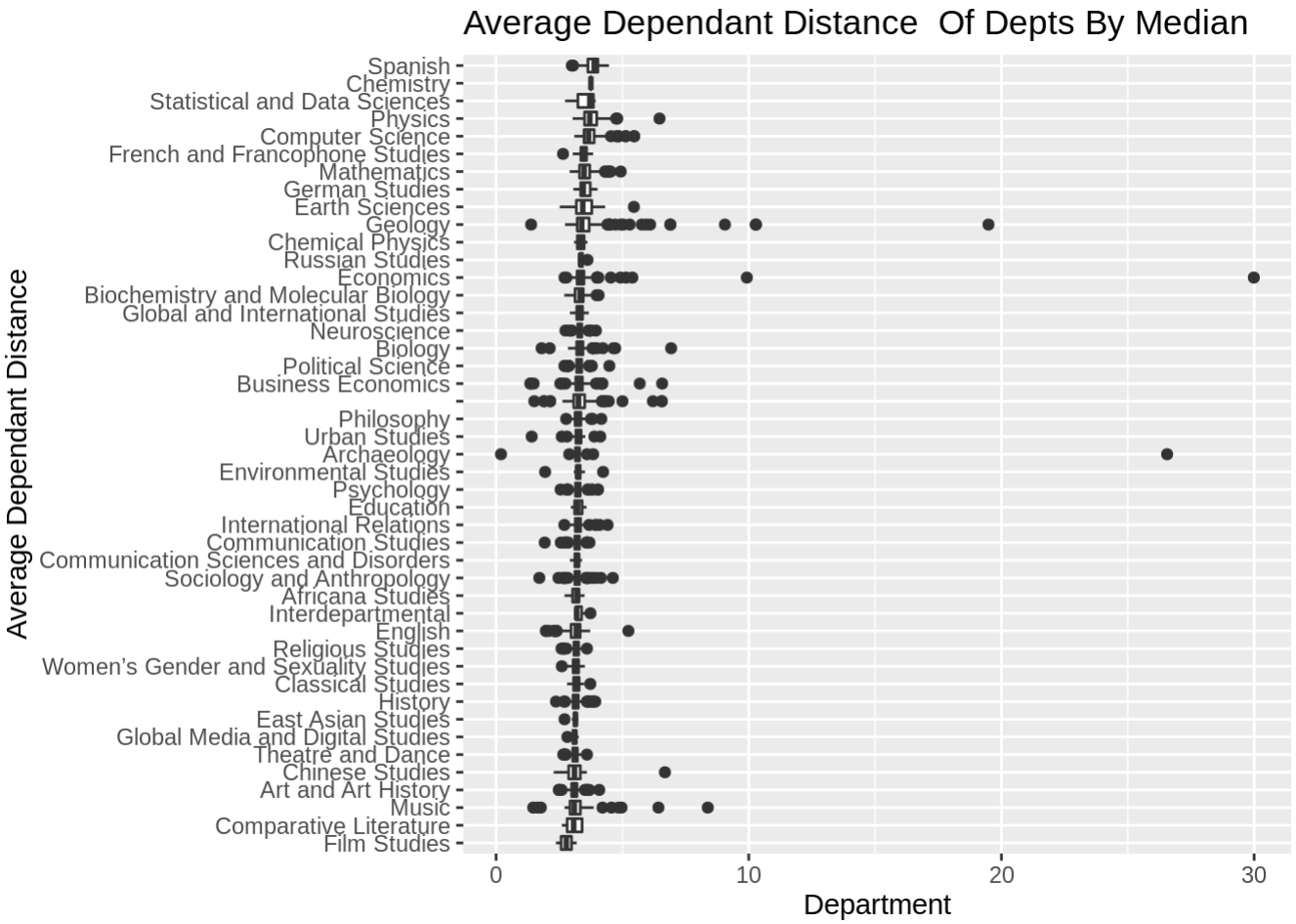

```
medSortBP(is.2$orlovZVal, "Orlov's Z")
```



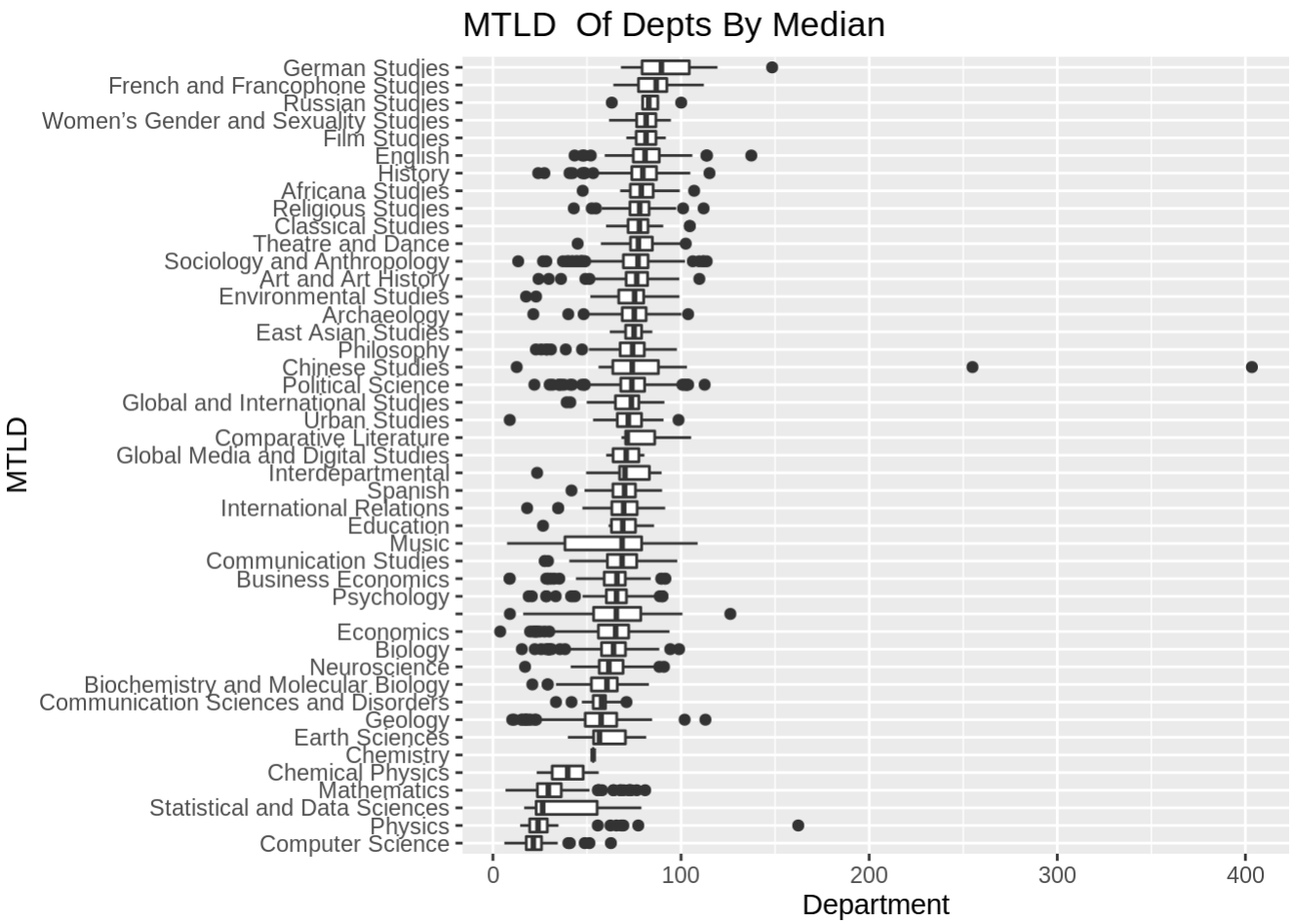

```
medSortBP(is.2$hd.dVal, "HD-D")
```



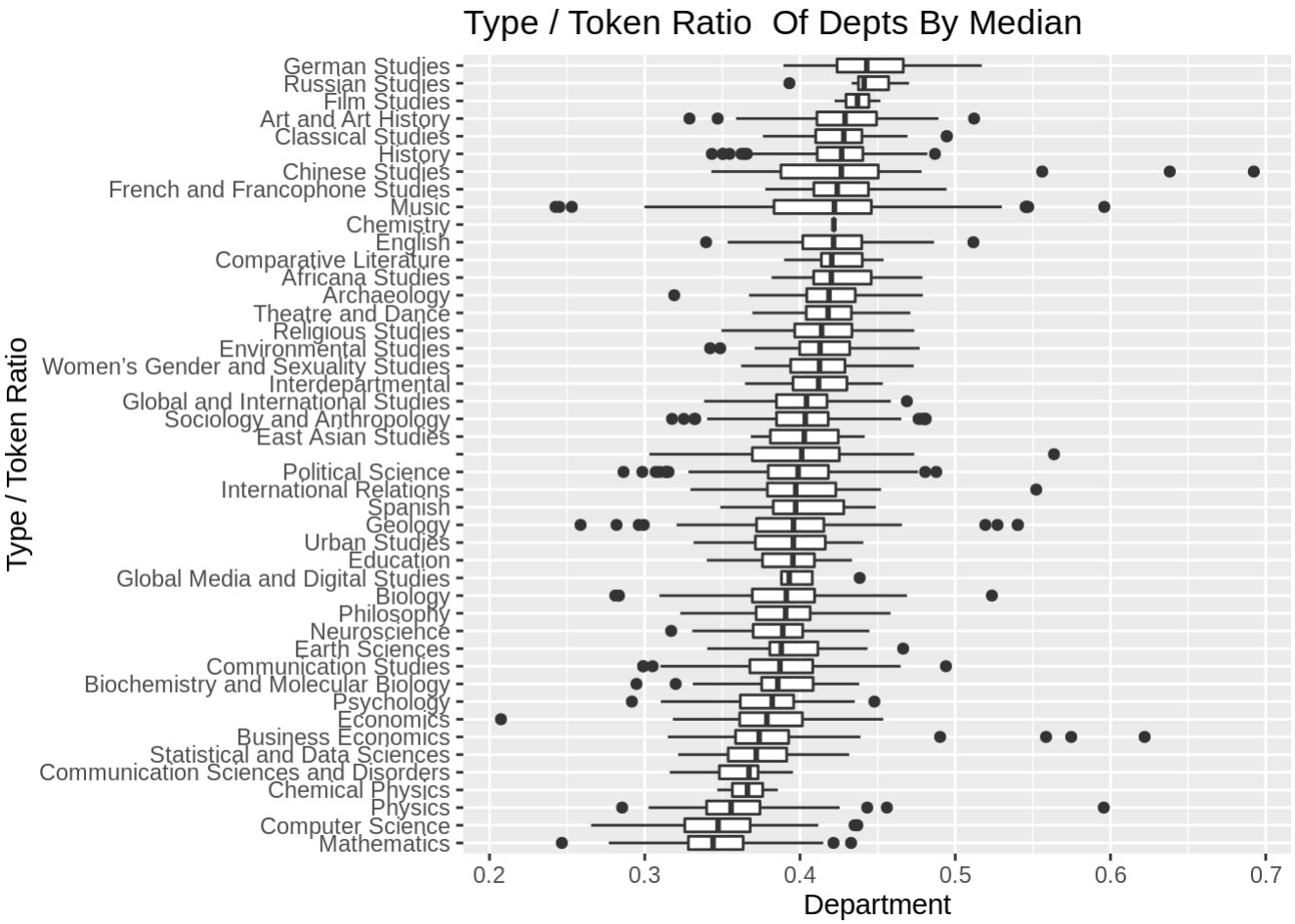

```
medSortBP(is.2$avgDepDistVal, "Average Dependant Distance")
```



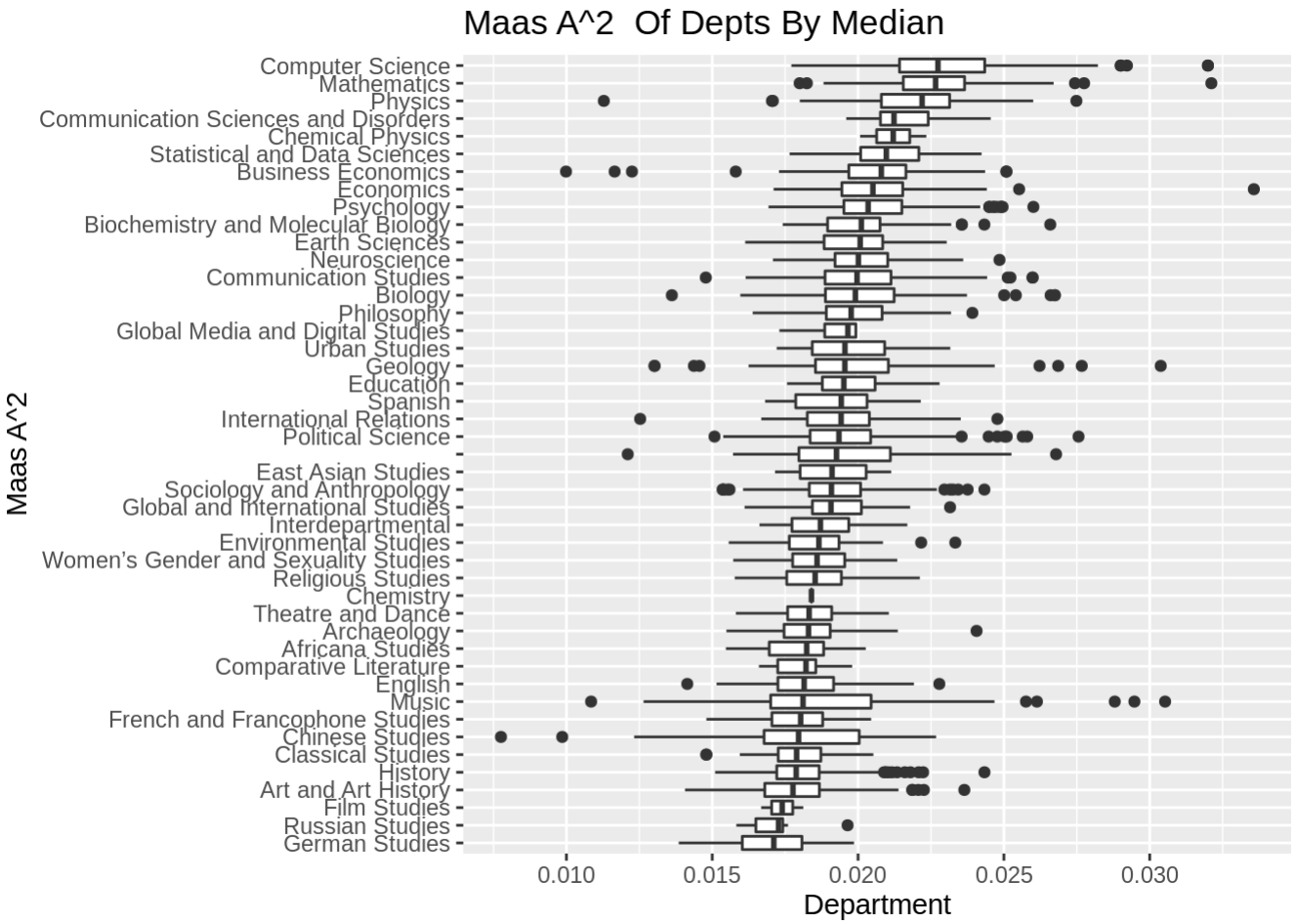

```
medSortBP(is.2$mtld, "MTLD")
```



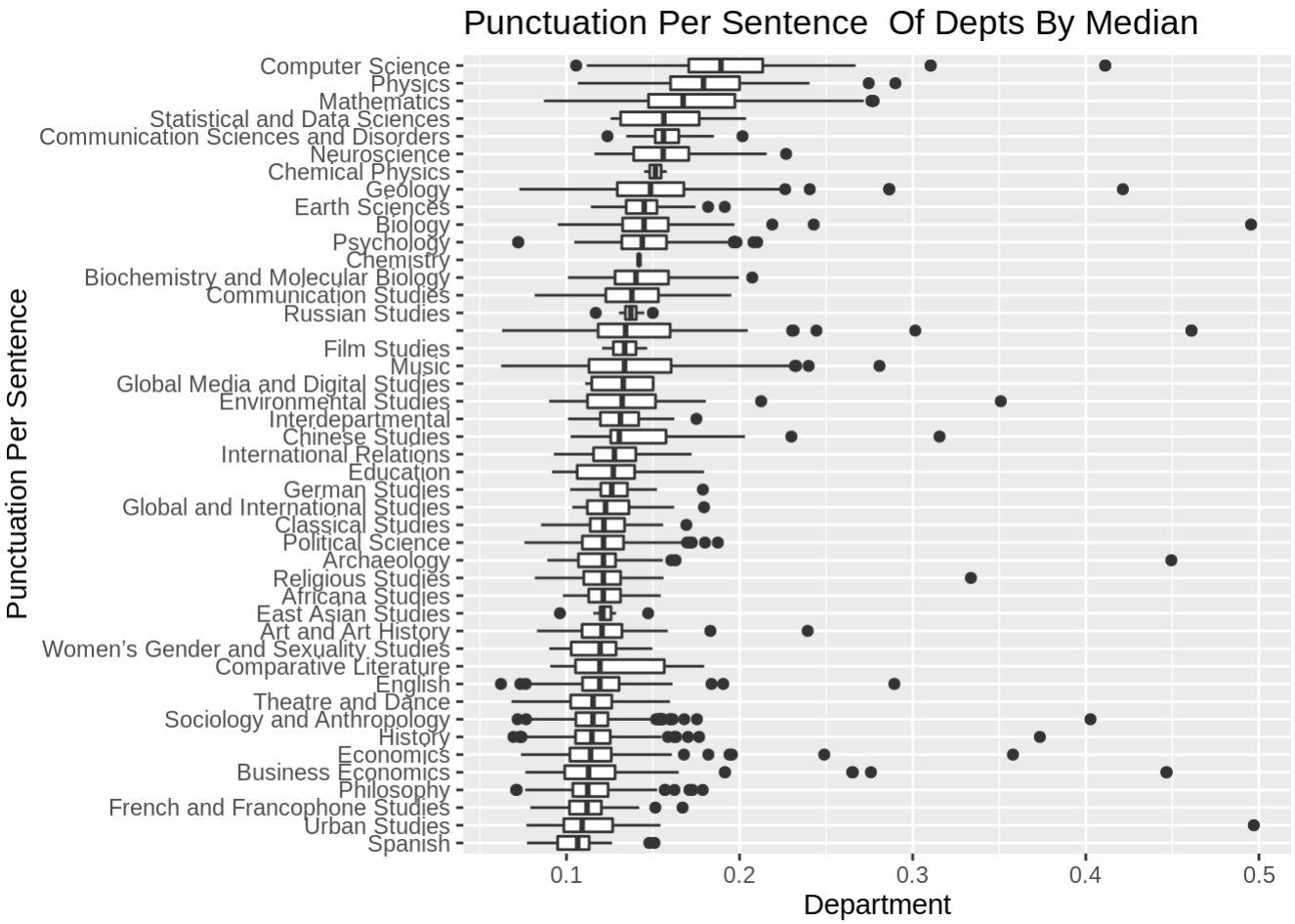

```
medSortBP(is.2$typeTokenRatioVal, "Type / Token Ratio")
```



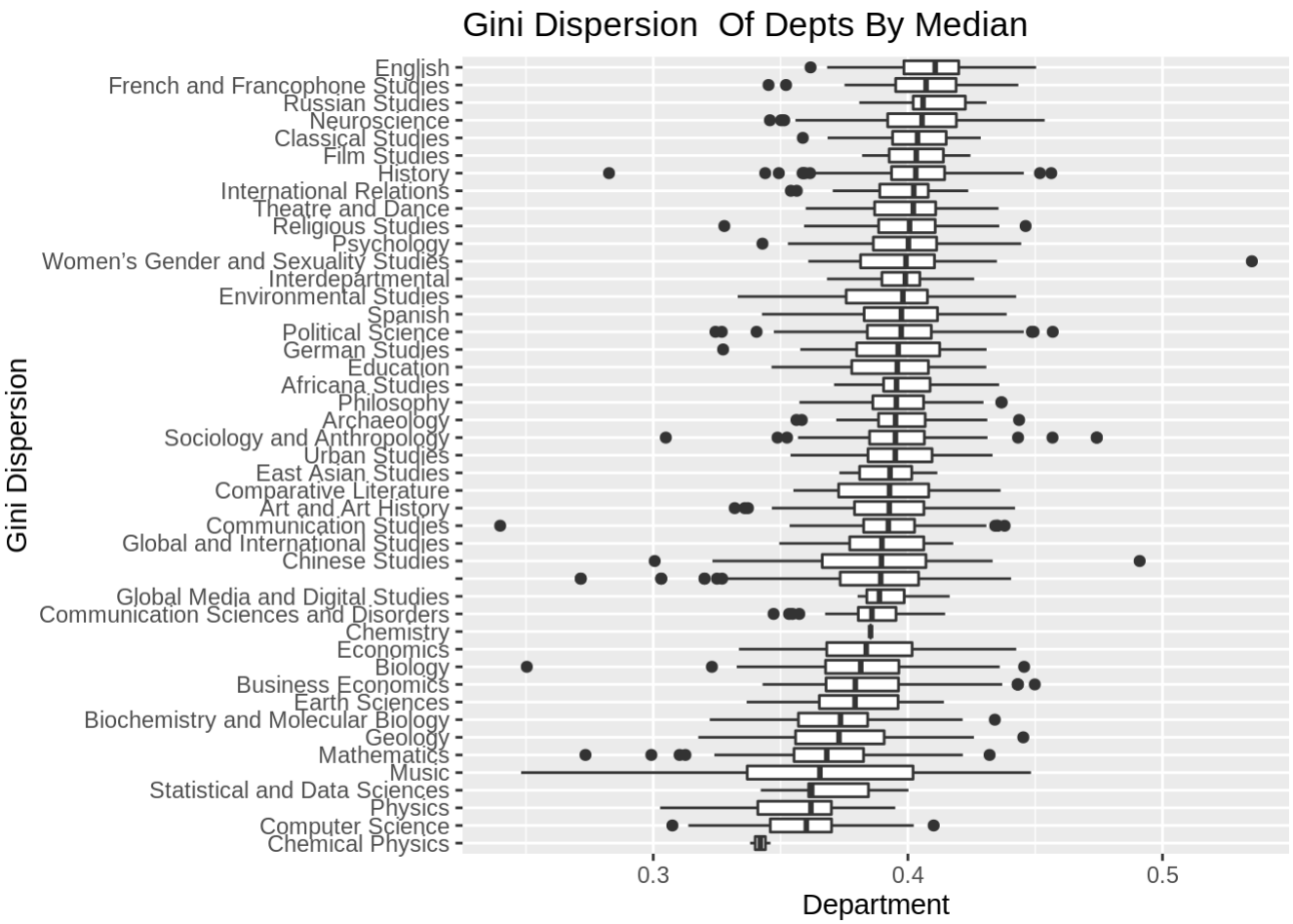

```
medSortBP(is.2$maasAsqVal, "Maas A^2")
```



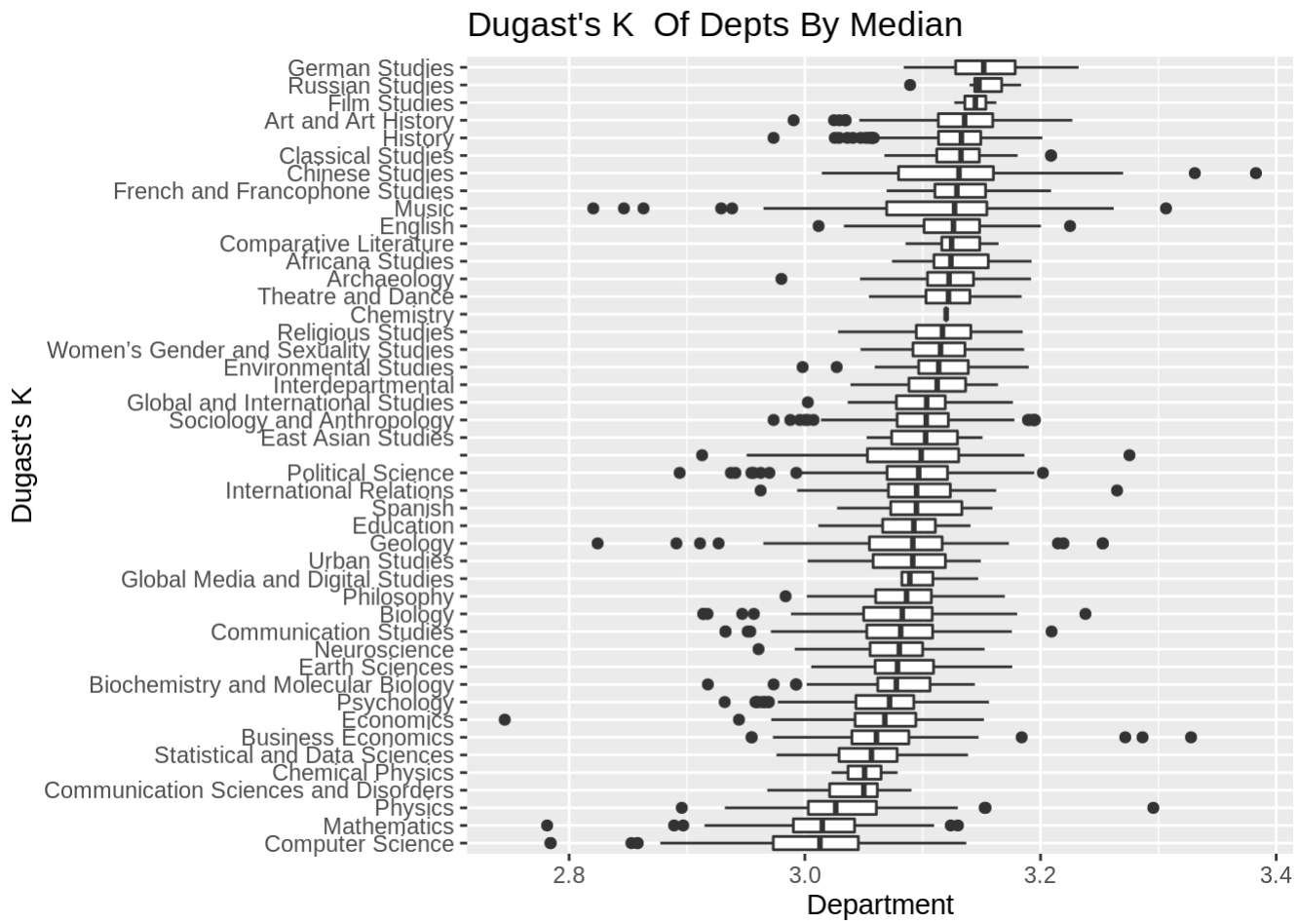

```
medSortBP(is.2$punctPerSent, "Punctuation Per Sentence")
```



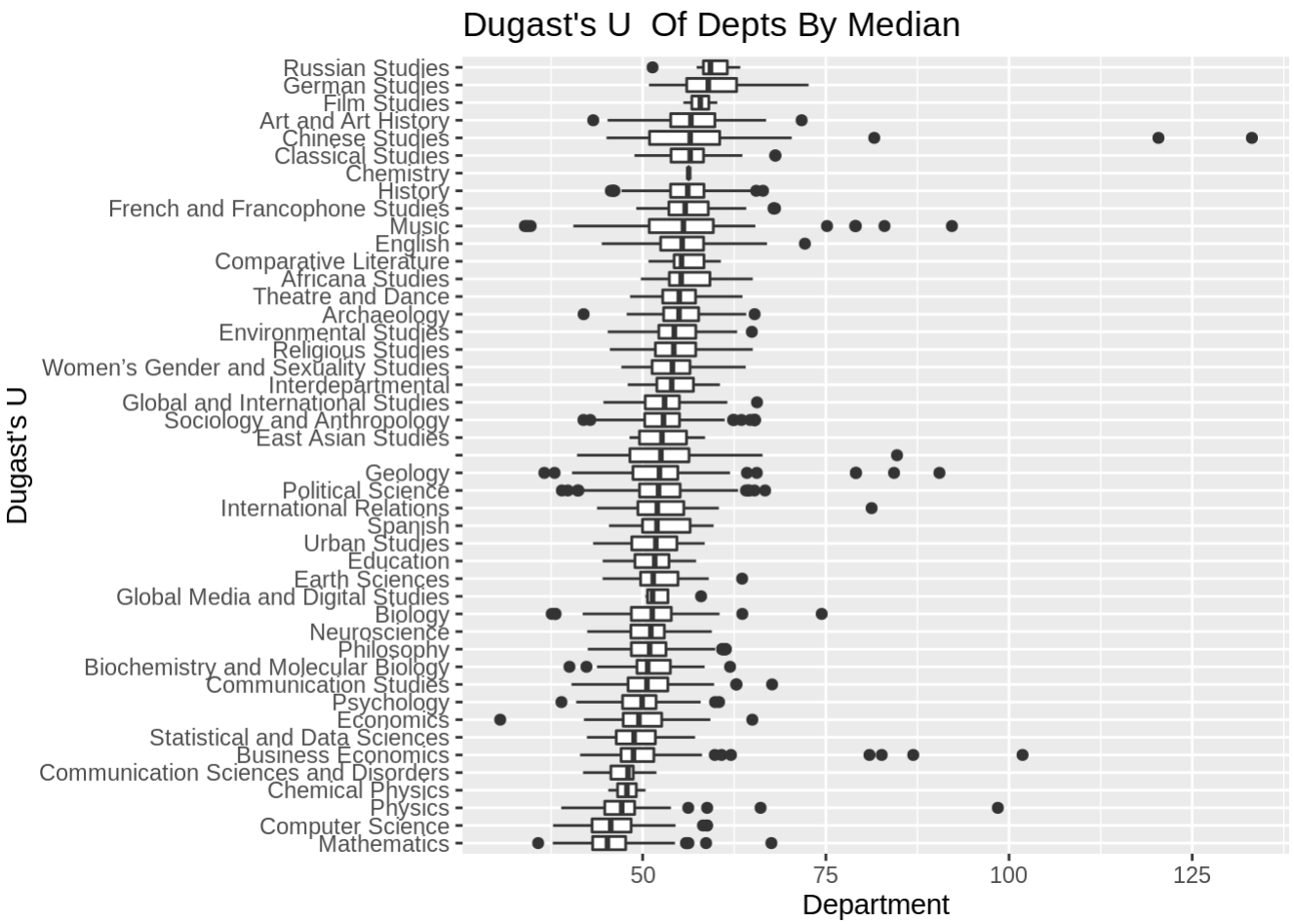

```
# medSortBP(is.2$punctPerTok, "Punctuation Per Token") # same as prior?
medSortBP(is.2$giniDispVal, "Gini Dispersion")
```



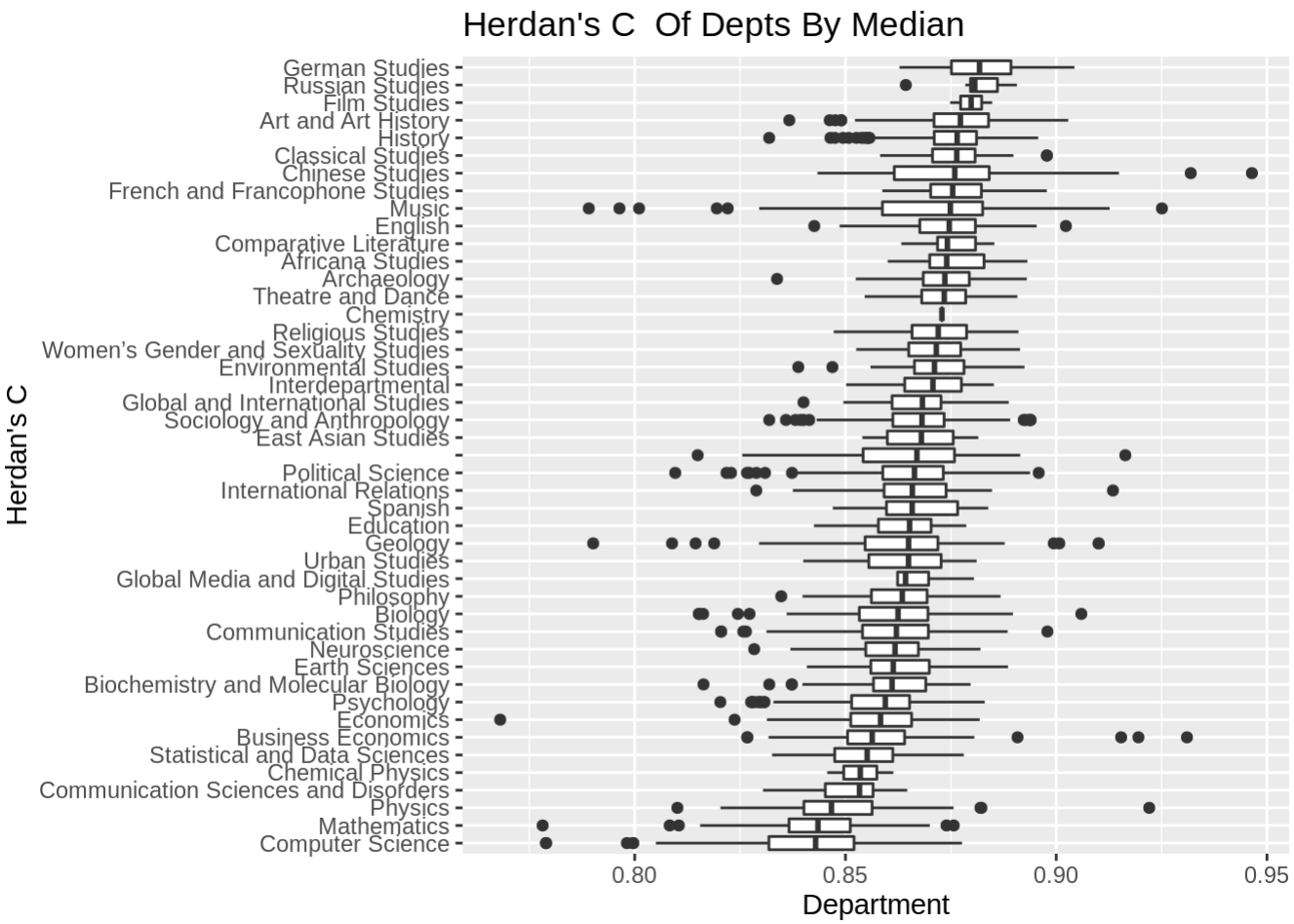

```
medSortBP(is.2$dugastsKVal, "Dugast's K")
```



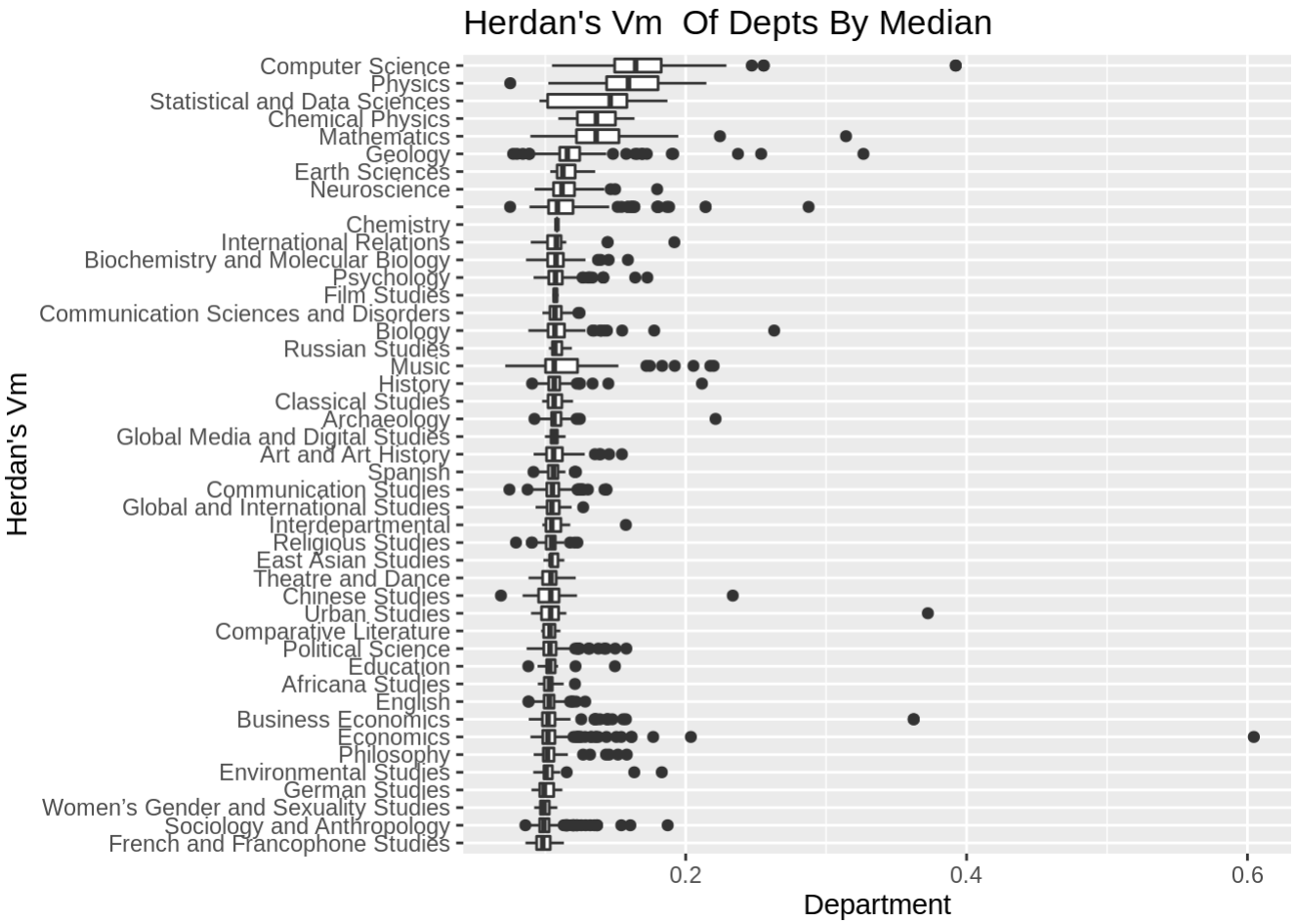

```
medSortBP(is.2$dugastsUVal, "Dugast's U")
```



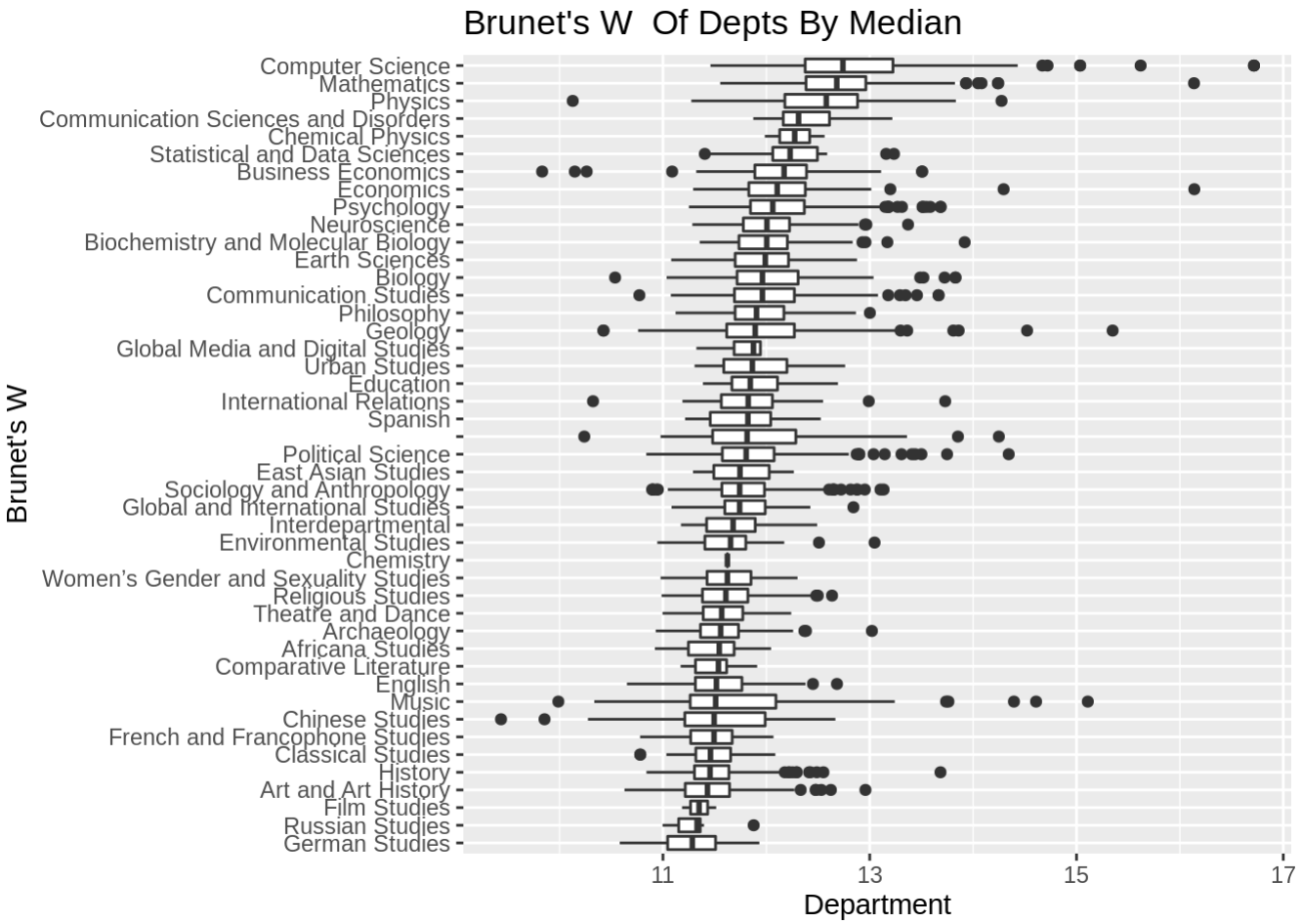

```
medSortBP(is.2$herdansCVal, "Herdan's C")
```




```
medSortBP(is.2$herdanVmVal, "Herdan's Vm")
```




```
medSortBP(is.2$brunetsWVal, "Brunet's W")
```



Art and Music published ISes seem to have more occurances of shorter papers, which makes sense with the idea that a large part of their work is non-written.

Some Further Tests

Found Kruskal-Wallis test as an option for checking if multiple groups have a different outcome.

```
testDiffs <- function(indep, dep) {
  kruskal.test(dep ~ indep, data =is.2)
}

testDiffs(is.2$isexemplar, is.2$dept1)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dep by indep
## Kruskal-Wallis chi-squared = 0.17271, df = 1, p-value = 0.6777
```

```
aov(lexRarity ~ dept1, data = is.2)
```

```
## Call:
##    aov(formula = lexRarity ~ dept1, data = is.2)
##
## Terms:
##                     dept1 Residuals
## Sum of Squares  17.48366  24.72974
## Deg. of Freedom       44      5244
##
## Residual standard error: 0.0686718
## Estimated effects may be unbalanced
```

```
testDiffs(is.2$isexemplar, is.2$pubdate)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dep by indep
## Kruskal-Wallis chi-squared = 51.922, df = 1, p-value = 5.775e-13
```

```
testDiffs(is.2$isexemplar, is.2$dept2)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dep by indep
## Kruskal-Wallis chi-squared = 4.1164, df = 1, p-value = 0.04247
```

Sling Some Models Around

```
# seemingly important obs. from EDA
m1 <- glm(as.factor(isexemplar) ~ lexDensity + lexRarity + pubdate +pagec, data = is.2, family =
binomial)
summary(m1)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ lexDensity + lexRarity +
##     pubdate + pagec, family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2234  -0.3898  -0.3165  -0.2462   2.8297
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.313e+02  2.847e+01  -8.123 4.55e-16 ***
## lexDensity  -2.837e+00  7.096e-01  -3.998 6.38e-05 ***
## lexRarity    1.061e+00  6.597e-01   1.608    0.108
## pubdate      1.133e-01  1.411e-02   8.031 9.68e-16 ***
## pagec        9.036e-03  1.247e-03   7.247 4.25e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470  on 5288  degrees of freedom
## Residual deviance: 2335  on 5284  degrees of freedom
## AIC: 2345
##
## Number of Fisher Scoring iterations: 6
```

```
# mix in departments of authors?
m2 <- glm(as.factor(isexemplar) ~ dept1 + dept2, data = is.2, family = binomial)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
#summary(m2) # can be a pretty big thing
```

```
# just simple statistics
m.simples <- glm(as.factor(isexemplar) ~ figc + wordc + pagec + len1 + len2 + len3 + len4 + len5
+ len6 + len7 + len8 + len9 + len10 + len11 + len12 + len13 + len14 + len15, data = is.2, family
= binomial)
summary(m.simples) # all simple statistics
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ figc + wordc + pagec +
##     len1 + len2 + len3 + len4 + len5 + len6 + len7 + len8 + len9 +
##     len10 + len11 + len12 + len13 + len14 + len15, family = binomial,
##     data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5708  -0.3740  -0.3182  -0.2777   3.5875
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.598e+00  1.485e-01 -24.234  < 2e-16 ***
## figc        -2.469e-04  7.643e-04  -0.323 0.746679
## wordc       -4.796e-05  1.707e-05  -2.809 0.004965 **
## pagec        9.071e-03  2.643e-03   3.433 0.000598 ***
## len1         1.887e-04  4.469e-05   4.224  2.4e-05 ***
## len2         2.946e-04  7.610e-05   3.872 0.000108 ***
## len3        -1.064e-04  7.941e-05  -1.340 0.180372
## len4        -2.602e-04  1.226e-04  -2.122 0.033874 *
## len5        -1.381e-04  1.801e-04  -0.767 0.442955
## len6         3.507e-04  1.944e-04   1.804 0.071222 .
## len7        -7.130e-05  2.303e-04  -0.310 0.756876
## len8         2.916e-04  2.532e-04   1.151 0.249565
## len9         3.484e-04  3.372e-04   1.033 0.301509
## len10       -1.211e-04  4.078e-04  -0.297 0.766463
## len11        6.209e-05  4.677e-04   0.133 0.894379
## len12        3.863e-04  6.374e-04   0.606 0.544505
## len13        6.503e-04  6.707e-04   0.970 0.332238
## len14        1.185e-03  1.125e-03   1.054 0.291961
## len15       -6.014e-03  1.994e-03  -3.016 0.002559 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2360.6  on 5270  degrees of freedom
## AIC: 2398.6
##
## Number of Fisher Scoring iterations: 6
```

```
# rough reduction
m.simples2 <- glm(as.factor(isexemplar) ~ wordc + pagec + len1 + len2 + len3, data = is.2, famil
y = binomial)
summary(m.simples2)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ wordc + pagec + len1 +
##     len2 + len3, family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6498  -0.3737  -0.3324  -0.2997   2.7716
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.536e+00  1.412e-01 -25.049  < 2e-16 ***
## wordc       -1.294e-05  1.110e-05  -1.167  0.24339
## pagec        9.885e-03  2.260e-03   4.375 1.22e-05 ***
## len1         1.131e-04  4.236e-05   2.670  0.00758 **
## len2         2.244e-04  7.119e-05   3.152  0.00162 **
## len3        -1.911e-04  6.938e-05  -2.754  0.00588 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2406.5  on 5283  degrees of freedom
## AIC: 2418.5
##
## Number of Fisher Scoring iterations: 5
```

```
# based of token values
m.tokens <- glm(as.factor(isexemplar) ~ punctPerTok + typeTokenRatio + avgToksSentVal + typeToke
nRatioVal + avgTokLenVal, data = is.2, family = binomial)
summary(m.tokens)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ punctPerTok + typeTokenRatio +
##     avgToksSentVal + typeTokenRatioVal + avgTokLenVal, family = binomial,
##     data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1861  -0.3722  -0.3474  -0.3246   2.5686
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -4.089e+00  7.822e-01  -5.228 1.71e-07 ***
## punctPerTok        7.218e+00  1.525e+00   4.734 2.20e-06 ***
## typeTokenRatio     1.670e+01  1.186e+01   1.408    0.159
## avgToksSentVal     8.701e-04  3.956e-03   0.220    0.826
## typeTokenRatioVal -1.408e+01  1.222e+01  -1.153    0.249
## avgTokLenVal      -1.412e-01  8.695e-02  -1.624    0.104
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470  on 5288  degrees of freedom
## Residual deviance: 2445  on 5283  degrees of freedom
## AIC: 2457
##
## Number of Fisher Scoring iterations: 5
```

```
# Part-Of-Speech Measures
m.pos <- glm(as.factor(isexemplar) ~ lexDensity + lexRarity, data = is.2, family = binomial)
summary(m.pos)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ lexDensity + lexRarity,
##     family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6982  -0.3615  -0.3447  -0.3306   2.4814
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.6415     0.4243  -3.869 0.000109 ***
## lexDensity   -2.9565     0.6786  -4.356 1.32e-05 ***
## lexRarity     0.4365     0.6297   0.693 0.488139
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2445.1  on 5286  degrees of freedom
## AIC: 2451.1
##
## Number of Fisher Scoring iterations: 5
```

```
# Dispersion
m.disp <- glm(as.factor(isexemplar) ~ evenDispVal + giniDispVal, data = is.2, family = binomial)
summary(m.disp)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ evenDispVal + giniDispVal,
##     family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.4345  -0.3663  -0.3588  -0.3493   2.4373
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.9653     2.6800  -1.480    0.139
## evenDispVal   0.4766    11.8317   0.040    0.968
## giniDispVal   2.4171    13.2275   0.183    0.855
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2468.5  on 5286  degrees of freedom
## AIC: 2474.5
##
## Number of Fisher Scoring iterations: 5
```

```
# Frequency Spectrum
m.freqspec <- glm(as.factor(isexemplar) ~ entropyVal + evennessVal + simpsonDVal, data = is.2, f
amily = binomial)
summary(m.freqspec)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ entropyVal + evennessVal +
##     simpsonDVal, family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3989  -0.3618  -0.3528  -0.3436   2.4290
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.8090     6.0027   0.135    0.893
## entropyVal    0.6551     0.5404   1.212    0.225
## evennessVal  -9.8012     9.5105  -1.031    0.303
## simpsonDVal   5.2803     7.6808   0.687    0.492
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2459.8  on 5285  degrees of freedom
## AIC: 2467.8
##
## Number of Fisher Scoring iterations: 5
```

Final Model, Misclassification, etc.

```
# Combine some stuff that worked
m.comb <- glm(as.factor(isexemplar) ~ lexDensity + pagec + pubdate + punctPerSent, data = is.2,
family = binomial)
summary(m.comb)
```

```
##
## Call:
## glm(formula = as.factor(isexemplar) ~ lexDensity + pagec + pubdate +
##     punctPerSent, family = binomial, data = is.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1854  -0.3894  -0.3138  -0.2443   2.8392
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.273e+02  2.852e+01  -7.969 1.60e-15 ***
## lexDensity   -3.154e+00  6.371e-01  -4.950 7.43e-07 ***
## pagec         8.825e-03  1.246e-03   7.084 1.40e-12 ***
## pubdate       1.113e-01  1.414e-02   7.869 3.57e-15 ***
## punctPerSent  4.266e+00  1.327e+00   3.214  0.00131 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2470.0  on 5288  degrees of freedom
## Residual deviance: 2328.1  on 5284  degrees of freedom
## AIC: 2338.1
##
## Number of Fisher Scoring iterations: 6
```

```
# m.comb is... good enough... for me!

cis <- confint(m.comb, level = 0.95)
```

```
## Waiting for profiling to be done...
```

```
exp(cis)
```

```
##                      2.5 %       97.5 %
## (Intercept)  5.105186e-124 1.931355e-75
## lexDensity    1.278327e-02 1.566313e-01
## pagec         1.006378e+00 1.011314e+00
## pubdate       1.087513e+00 1.149539e+00
## punctPerSent  4.906841e+00 9.137919e+02
```

```
m.succ <- ifelse(fitted(m.comb) > 0.5, 1, 0)
tally(~ m.succ + m.comb$y, data = is.2, format = "proportion")
```

```
##       m.comb$y
## m.succ            0           1
##      0 0.9374172811 0.0623936472
##      1 0.0001890717 0.0000000000
```

Multinomial Regression for Primary Major Prediction

Attempted a multinomial regression model to fit towards the IS's primary major, to no success – hence abandoned.

```
#library(nnet)
# hmmm... This ran for many minutes and concluded with an AIC val > 20,000, so giving up on this part of the analysis
#maj.guess <- multinom(dept1 ~ lexDensity + lexRarity + pubdate +pagec, data = is.2)
#summary(maj.guess)
```