QML Group Project

Packages

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.1     v stringr   1.5.2
v ggplot2   4.0.0     v tibble    3.3.0
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.1.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(brms)
```

```
Loading required package: Rcpp
Loading 'brms' package (version 2.23.0). Useful instructions
can be found by typing help('brms'). A more detailed introduction
to the package is available through vignette('brms_overview').

Attaching package: 'brms'

The following object is masked from 'package:stats':

    ar
```

```r
library(bayesplot)
```

```
This is bayesplot version 1.14.0
- Online documentation and vignettes at mc-stan.org/bayesplot
- bayesplot theme set to bayesplot::theme_default()
   * Does _not_ affect other ggplot2 plots
   * See ?bayesplot_theme_set for details on theme setting

Attaching package: 'bayesplot'

The following object is masked from 'package:brms':

    rhat
```

```r
library(ggdist)
```

```

Attaching package: 'ggdist'

The following objects are masked from 'package:brms':

    dstudent_t, pstudent_t, qstudent_t, rstudent_t
```

```r
library(posterior)
```

```
This is posterior version 1.6.1

Attaching package: 'posterior'

The following object is masked from 'package:bayesplot':

    rhat

The following objects are masked from 'package:stats':

    mad, sd, var

The following objects are masked from 'package:base':

    %in%, match
```

Data Preprocessing

Because the original data was in Croatian and thus difficult for us to interpret, we translated column names and word class values to English. We also dropped any unneeded columns.

Read in Data

```r
original1 = read_tsv("./data/megahr.tsv")
```

```
Rows: 3000 Columns: 62
-- Column specification --------------------------------------------------------
Delimiter: "\t"
chr  (4): leksem, vrsta.riječi, rod, živost
dbl (58): broj.slova, frek, k.N, k.M, k.C, k.STD, k.MIN, k.MAX, k.N.m, k.M.m...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
original2 = read_tsv("./data/megahr.2nd.tsv")
```

```
Rows: 3000 Columns: 62
-- Column specification --------------------------------------------------------
Delimiter: "\t"
chr  (4): leksem, vrsta.riječi, rod, živost
dbl (58): broj.slova, frek, k.N, k.M, k.C, k.STD, k.MIN, k.MAX, k.N.m, k.M.m...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
original <- rbind(original1, original2)
```

Drop Unnecessary Columns

```r
original <- original[-c(7:20,49:62)] #remove unneccessary types
original <- original[-c(7,9:21,23:34)] #remove all but the mean for clarity
```

Translate Column Names to English

```
original <- original|>
  rename(
    token = 'leksem',
    word_class = 'vrsta.riječi',
    gender = 'rod',
    animacy = 'živost',
    number_of_letters = 'broj.slova',
    freq = 'frek'
  )

original <- original|>
  rename(
    subjective_frequency_mean = 'č.M',
    imageability_mean = 'p.M'
  )
```

Translate Word Classes to English

```
original <- original|>
  mutate(word_class = case_when(
    word_class == "Nc" ~ "noun",
    word_class == "Vm" ~ "verb",
    word_class == "Ag" ~ "adj",
    word_class == "Rg" ~ "adv"
  ))
```

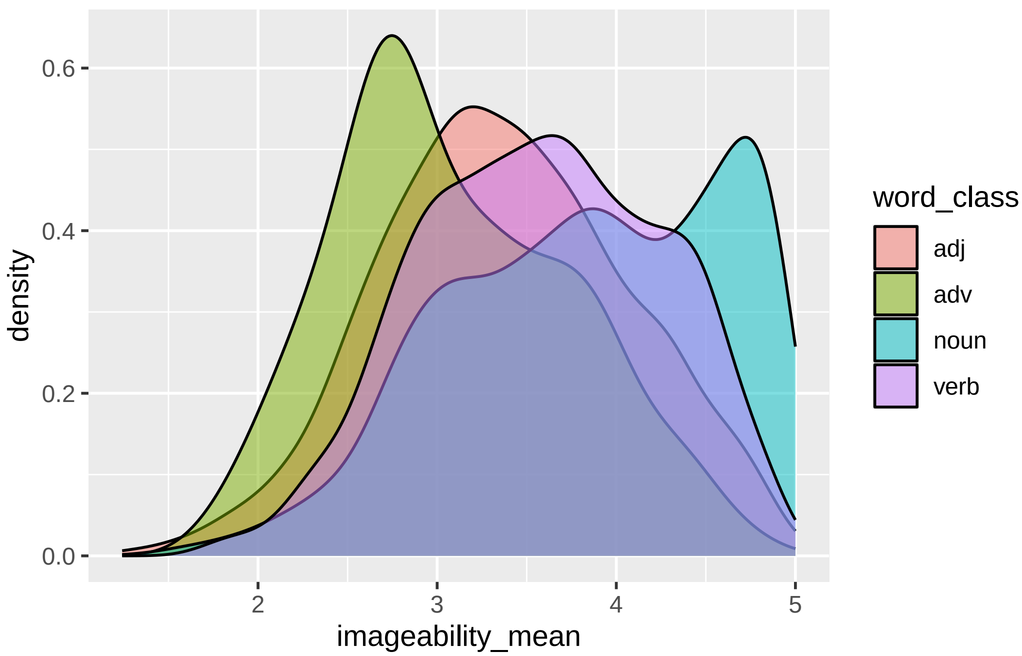
Transform Word Class into Factor

```
original <- original|>
  mutate(word_class = as.factor(word_class))
```

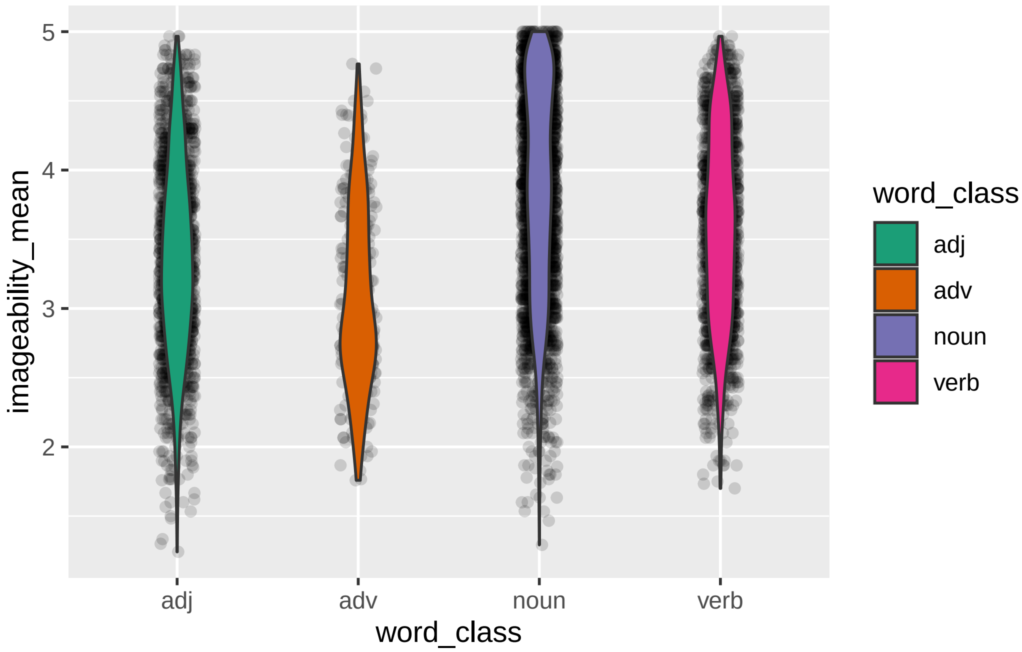
Effect of Word Class on Imageability

Data Plots

```
original |>
  ggplot(aes(imageability_mean, fill = word_class)) +
  geom_density(alpha = 0.5)
```




```
original |>
  ggplot(aes(word_class, imageability_mean, fill = word_class)) +
  geom_jitter(alpha = 0.15, width = 0.1) +
  geom_violin(width = 0.2) +
  scale_fill_brewer(palette = "Dark2")
```



Summary of Imageability Score Means by Word Class

```
original_summ <- original |>
  group_by(word_class) |>
  summarise(
    mean(imageability_mean), median(imageability_mean), sd(imageability_mean)
  )

original_summ
```

```
# A tibble: 4 x 4
  word_class `mean(imageability_mean)` `median(imageability_mean)`
  <fct>                          <dbl>                       <dbl>
1 adj                             3.37                        3.33
2 adv                             3.09                        2.97
3 noun                            3.84                        3.9
4 verb                            3.58                        3.6
# i 1 more variable: `sd(imageability_mean)` <dbl>
```

Fit Regression Model

```r
img_bm <- brm(
  imageability_mean ~ word_class,
  family = gaussian,
  data = original,
  seed = 6725,
  file = "cache/img_bm"
)
```

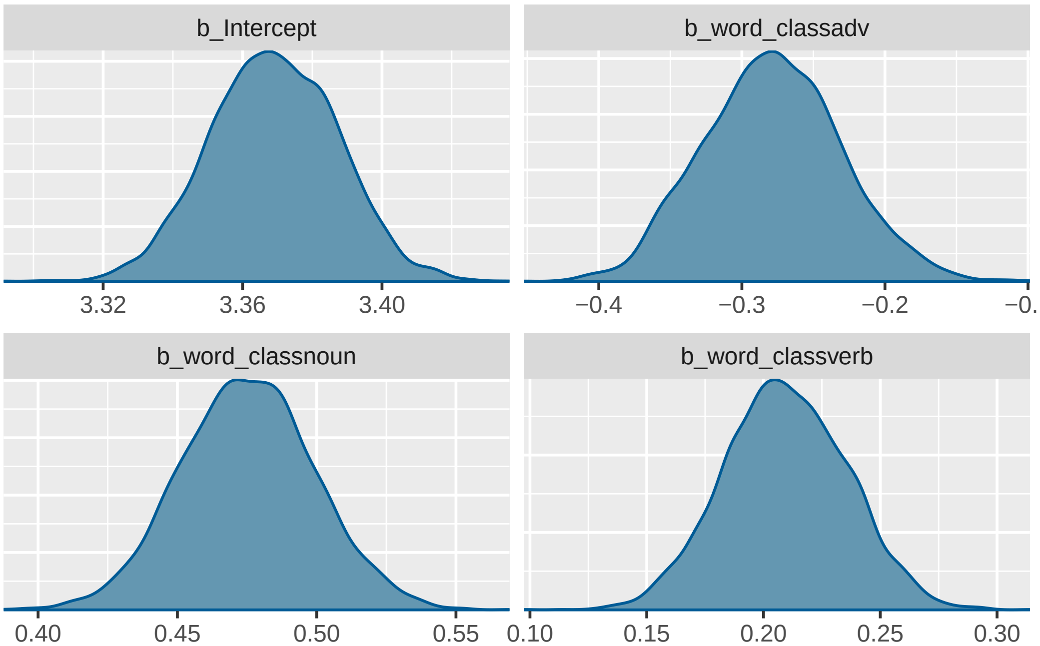
```r
summary(img_bm)
```

```
 Family: gaussian
  Links: mu = identity
Formula: imageability_mean ~ word_class
   Data: original (Number of observations: 6000)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept          3.37      0.02     3.33     3.40 1.00     3013     2934
word_classadv     -0.28      0.05    -0.37    -0.18 1.00     3927     3418
word_classnoun     0.48      0.02     0.43     0.52 1.00     3065     2655
word_classverb     0.21      0.03     0.16     0.26 1.00     3189     2813

Further Distributional Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma     0.72      0.01     0.70     0.73 1.00     4447     2826

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

Results

Posterior Plots

```r
mcmc_dens(img_bm, pars = vars(starts_with("b_")))
```



Get Draws

```r
img_bm_draws <- as_draws_df(img_bm) |>
  mutate(
    adj = b_Intercept,
    adv = b_Intercept + b_word_classadv,
    noun = b_Intercept + b_word_classnoun,
    verb = b_Intercept + b_word_classverb
  )

img_bm_long <- img_bm_draws |>
  select(adj:verb) |>
  pivot_longer(everything(), names_to = "word_class", values_to = "pred")
```

```
Warning: Dropping 'draws_df' class as required metadata was removed.
```

```r
img_bm_long
```

```
# A tibble: 16,000 x 2
   word_class  pred
   <chr>      <dbl>
 1 adj         3.38
 2 adv         3.07
 3 noun        3.84
 4 verb        3.55
 5 adj         3.38
 6 adv         3.05
 7 noun        3.82
 8 verb        3.59
 9 adj         3.37
10 adv         3.07
# i 15,990 more rows
```

Table 1: Imageability Mean, SD, and CrIs by Word Class

```r
img_bm_tab <- img_bm_long |>
  group_by(word_class) |>
  summarise(
    mean = mean(pred), sd = sd(pred),
    `99%` = paste0("[", paste(quantile2(pred, c(0.005, 0.995)) |> round(2), collapse = ", ")
    `95%` = paste0("[", paste(quantile2(pred, c(0.025, 0.975)) |> round(2), collapse = ", ")
    `75%` = paste0("[", paste(quantile2(pred, c(0.125, 0.875)) |> round(2), collapse = ", ")
  )

img_bm_tab
```

```
# A tibble: 4 x 6
  word_class  mean     sd `99%`        `95%`        `75%`
  <chr>      <dbl>  <dbl> <chr>        <chr>        <chr>
1 adj         3.37 0.0182 [3.32, 3.42] [3.33, 3.4]  [3.35, 3.39]
2 adv         3.09 0.0443 [2.98, 3.2]  [3.01, 3.18] [3.04, 3.14]
3 noun        3.84 0.0146 [3.81, 3.88] [3.82, 3.87] [3.83, 3.86]
4 verb        3.58 0.0184 [3.53, 3.63] [3.54, 3.62] [3.56, 3.6]
```

```r
img_bm_tab |>
  knitr::kable(
    col.names = c("", "Mean", "SD", "99% CrI", "95% CrI", "60% CrI"),
    digits = 1, align = c("rcccc")
  )
```

	Mean	SD	99% CrI	95% CrI	60% CrI
adj	3.4	0	[3.32, 3.42]	[3.33, 3.4]	[3.35, 3.39]
adv	3.1	0	[2.98, 3.2]	[3.01, 3.18]	[3.04, 3.14]
noun	3.8	0	[3.81, 3.88]	[3.82, 3.87]	[3.83, 3.86]
verb	3.6	0	[3.53, 3.63]	[3.54, 3.62]	[3.56, 3.6]

Reporting

We fitted a Bayesian regression model to the mean imageability score of Croatian words. We used a Gaussian distribution for the outcome and word class (adjective, adverb, noun, verb) as the only predictor. Word class was coded using the default treatment coding.

According to the model, the mean imageability score of adjectives is between 3.33 and 3.40, of adverbs is between 3.01 and 3.18, of nouns is between 3.82 and 3.87, and of verbs is between 3.54 and 3.62, at 95% probability.

Table 1 reports mean, SD, and 99% and 75% CrIs.

When comparing adverbs, nouns, and verbs to adjectives, at 95% confidence, the mean imageability score of adverbs is 0.18 to 0.37 points lower than that of adjectives (mean = 0.28, SD = 0.05), while the mean imageability score of nouns is 0.43 to 0.52 points higher than that of adjectives (mean = 0.48, SD = 0.02), and the mean imageability score of verbs is 0.16 to 0.26 points higher than that of adjectives (mean = 0.21, SD = 0.03)

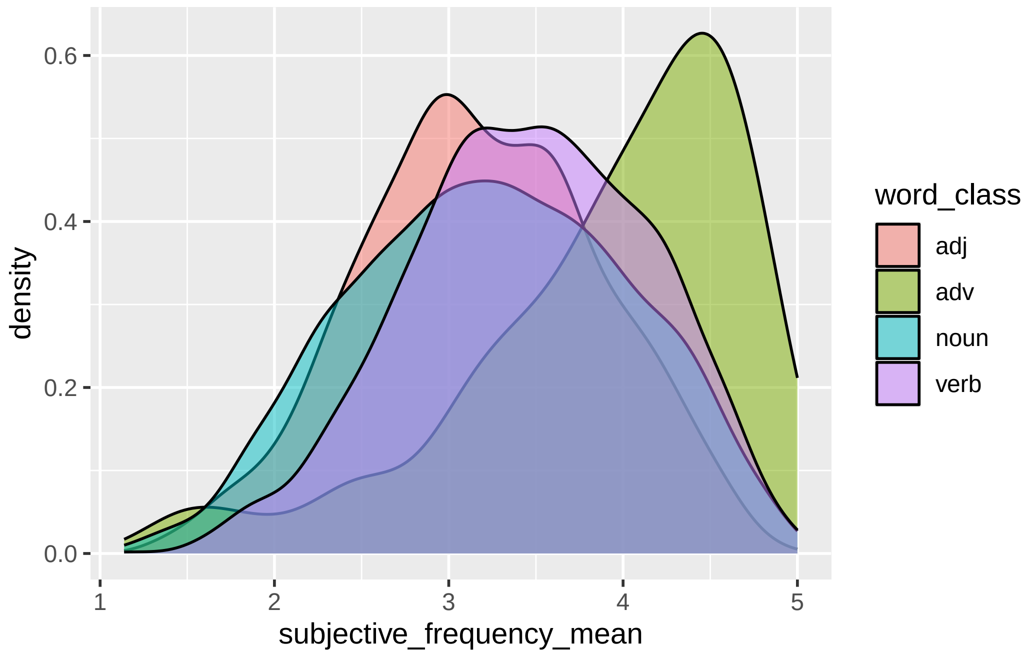
Comparison to Original Study

The findings from this Bayesian analysis were similar to the findings from a frequentist analysis of the same data in Peti-Stantić et al. (2021). As in the original study, we found evidence that mean imageability score is highest for nouns, followed by verbs, then adjectives, then adverbs. As the 99% credible intervals for mean imageability score do not overlap, the posteriors suggest differences between word classes.

Effect of Word Class on Subjective Frequency

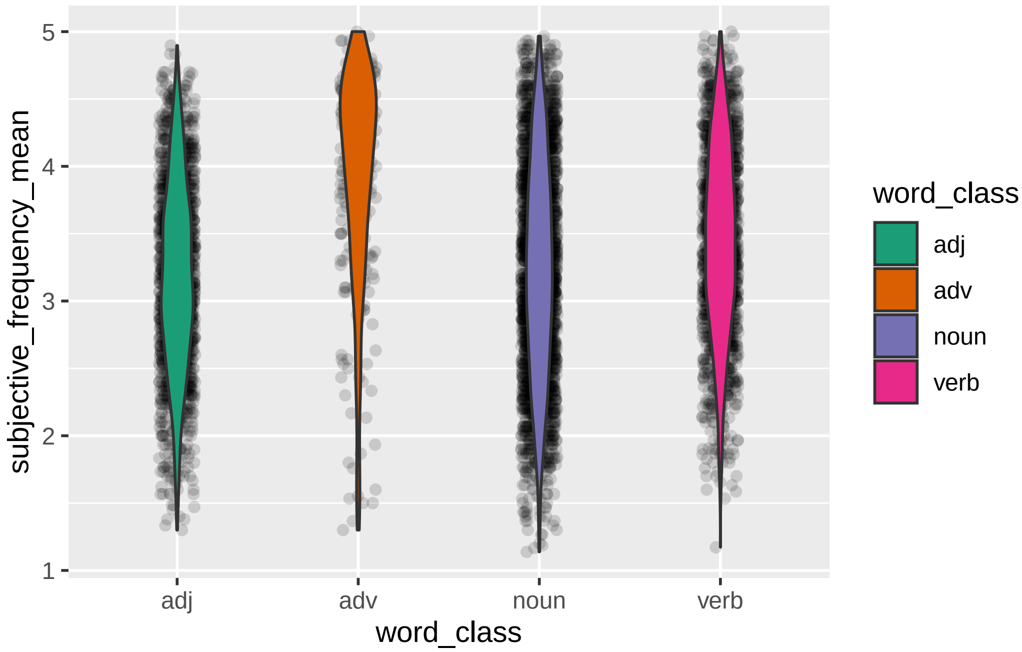
Plotting the Types

```
original |>
  drop_na(word_class) |>
  ggplot(aes(subjective_frequency_mean, fill = word_class)) +
  geom_density(alpha = 0.5)
```



Examining the Data Structure

```
original |>
  ggplot(aes(word_class, subjective_frequency_mean, fill = word_class)) +
  geom_jitter(alpha = 0.15, width = 0.1) +
  geom_violin(width = 0.2) +
  scale_fill_brewer(palette = "Dark2")
```



Summarising the Data by Word Class

```
original_summ <- original |>
  group_by(word_class) |>
  summarise(
    mean(subjective_frequency_mean)
  )

original_summ
```

```
# A tibble: 4 x 2
  word_class `mean(subjective_frequency_mean)`
  <fct>                                  <dbl>
1 adj                                     3.18
2 adv                                     3.91
3 noun                                    3.24
4 verb                                    3.45
```

Creating a Regression Model of the Data

```r
subj_freq_bm <- brm(
  subjective_frequency_mean ~ word_class,
  family = gaussian,
  data = original,
  seed = 6725,
  file = "cache/subj_freq_bm"
)
```

Show Results of Model

```r
summary(subj_freq_bm)
```

```
 Family: gaussian
  Links: mu = identity
Formula: subjective_frequency_mean ~ word_class
   Data: original (Number of observations: 6000)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
               Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept          3.18      0.02     3.14     3.22 1.00     3760     2848
word_classadv      0.73      0.05     0.64     0.83 1.00     3969     2738
word_classnoun     0.06      0.02     0.02     0.11 1.00     3999     3329
word_classverb     0.28      0.03     0.22     0.33 1.00     3583     3274

Further Distributional Parameters:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
sigma     0.73      0.01     0.72     0.75 1.00     4466     3017

Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
and Tail_ESS are effective sample size measures, and Rhat is the potential
scale reduction factor on split chains (at convergence, Rhat = 1).
```

Mutate the Draws

```r
subj_freq_bm_draws <- as_draws_df(subj_freq_bm) |>
  mutate(
    adj = b_Intercept,
    adv = b_Intercept + b_word_classadv,
    noun = b_Intercept + b_word_classnoun,
    verb = b_Intercept + b_word_classverb
  )
```

Table Summarization

```r
subj_freq_bm_long <- subj_freq_bm_draws |>
  select(adj:verb) |>
  pivot_longer(everything(), names_to = "word_class", values_to = "pred")
```

```
Warning: Dropping 'draws_df' class as required metadata was removed.
```

```r
subj_freq_bm_tab <- subj_freq_bm_long |>
  group_by(word_class) |>
  summarise(
    mean = mean(pred), sd = sd(pred),
    `99%` = paste0("[", paste(quantile2(pred, c(0.005, 0.995)) |> round(2), collapse = ", ")
    `95%` = paste0("[", paste(quantile2(pred, c(0.025, 0.975)) |> round(2), collapse = ", ")
    `75%` = paste0("[", paste(quantile2(pred, c(0.125, 0.875)) |> round(2), collapse = ", ")
  )

subj_freq_bm_tab
```

```
# A tibble: 4 x 6
  word_class  mean     sd `99%`        `95%`        `75%`
  <chr>      <dbl>  <dbl> <chr>        <chr>        <chr>
1 adj         3.18 0.0191 [3.13, 3.23] [3.14, 3.22] [3.16, 3.2]
2 adv         3.91 0.0457 [3.8, 4.03]  [3.82, 4]    [3.86, 3.96]
3 noun        3.24 0.0141 [3.2, 3.28]  [3.21, 3.27] [3.23, 3.26]
4 verb        3.46 0.0189 [3.41, 3.5]  [3.42, 3.49] [3.43, 3.48]
```

Nicer Table

```r
subj_freq_bm_tab |>
  knitr::kable(
    col.names = c("", "Mean", "SD", "99% CrI", "95% CrI", "75% CrI"),
    digits = 2, align = c("rccccc")
  )
```

	Mean	SD	99% CrI	95% CrI	75% CrI
adj	3.18	0.02	[3.13, 3.23]	[3.14, 3.22]	[3.16, 3.2]
adv	3.91	0.05	[3.8, 4.03]	[3.82, 4]	[3.86, 3.96]
noun	3.24	0.01	[3.2, 3.28]	[3.21, 3.27]	[3.23, 3.26]
verb	3.46	0.02	[3.41, 3.5]	[3.42, 3.49]	[3.43, 3.48]

Reporting

We fitted a Bayesian regression model to Mean Subjective Frequency (MSF) of Croatian words, measured on a Likert scale between 1 and 5. We used a Gaussian distribution for the outcome and word class (adjective, adverb, noun, verb) as the only predictor. Word class was treated as a default factored input.

According to the model, the mean MSF of adjectives is between 3.14 and 3.22, of adverbs is between 3.82 and 4.00, of nouns is between 3.21 and 3.27, and of verbs is between 3.42 and 3.49, at 95% probability. The table above reports mean, SD, 99%, 95% and 75% CrIs. When comparing the word classes at 95% confidence, adverbs were rated 0.64-0.83 more frequently used (mean = 3.91, SD = 0.05) than adjectives (mean = 3.18, SD = 0.02). Additionally, nouns were rated 0.02-0.11 more frequently used (mean = 3.24, SD = 0.02) and verbs were rated 0.22-0.33 more frequently used (mean = 3.46, SD = 0.03) than adjectives.

Compared to Peti-Stantić et al. (2021), we found evidence of similar trends in the mean of subjective frequency, with there being a large difference in the means of adverbs and verbs as compared to the mean of adjectives. However, in contrast to the study, which found there to be no ‘significant’ difference in the means in subjective frequency of nouns and adjectives, we can conclude with 95% confidence that there is a small but positive difference in the mean of nouns as compared to adjectives.