Appendix 3: Meta-Analyses of Kolmogorov Complexity

Chris Bentz

January 20, 2021

Session Info

Give the session info (reduced).

```
## [1] "R version 3.6.3 (2020-02-29)"
```

```
## [1] "x86_64-pc-linux-gnu"
```

Load Packages

Load packages. If they are not installed yet on your local machine, use install.packages() to install them.

```
library(readr)
library(tidyr)
library(ggplot2)
library(plyr)
library(scales)
library(rstatix)
```

Give the package versions.

```
## rstatix  scales    plyr ggplot2   tidyr   readr
## "0.6.0" "1.1.1" "1.8.6" "3.3.3" "1.1.2" "1.4.0"
```

Load Data

Load data file with morphological complexity estimations. These stem from an earlier project applying measures related to Kolmogorov complexity to parallel texts of "Alice in Wonderland". The data stems from the so-called "semi-parallel" analyses. In order to assess the variance of complexity estimations for different parts of the corpora, 10% of the sentences of each corpus were randomly drawn (this was repeated 1000 times). For further descriptions see Ehret & Szmrecsanyi (2016).

```
# Use the path to the file semialice_morphratios.csv in its raw format.
results <- read_csv("https://raw.githubusercontent.com/IWMLC/complexityMeaning/main/semialice_morphrati
```

Give some simple statistics for this data frame of results.

```
length(unique(results$language)) # i.e. number of different languages
```

```
## [1] 10
```

```
ncol(results)-1 # i.e. number of complexity measurements
```

```
## [1] 1000
```

Data Pre-Processing

Select columns (i.e. number of repeated measurements). We might use all 1000 repeated measurements per language. However, in order to not inflate the number of data points artificially we might also just choose a subset of the measurements.

```
results.short <- results[2:21]
#head(results.short)
```

Scale all numerical columns to make them more commensurable.

```
results.scaled <- cbind(results[1], scale(results.short))
```

Transform data frame from wide format to long format (this is necessary for later plotting and analyses by columns rather than rows).

```
results.long <- gather(results.scaled, key = measure,
                       value = value, morphratio.1:tail(names(results.scaled), 1))
head(results.long)
```

```
##    language      measure      value
## 1     dutch morphratio.1 -1.1980030
## 2   english morphratio.1 -0.8960548
## 3   finnish morphratio.1  1.7294352
## 4    french morphratio.1 -0.5442415
## 5    german morphratio.1 -0.5615801
## 6 hungarian morphratio.1  0.9624459
```

Density Distributions

Plot density distributions of complexity measurements by language. Individual measurements are plotted as black dots. The central tendency value (i.e. mean, median, or both) of complexity measurements per language might be indicated as a red dashed line.

Get mean, median, and standard deviation values.

```
# get mean values for each language
mu <- ddply(results.long, "language", summarise, grp.mean = mean(value, na.rm = T))
 # get median values for each language
med <- ddply(results.long, "language", summarise, grp.median = median(value, na.rm = T))
 # get standard deviation values for each language
sdev <- ddply(results.long, "language", summarise, grp.sd = sd(value, na.rm = T))
```

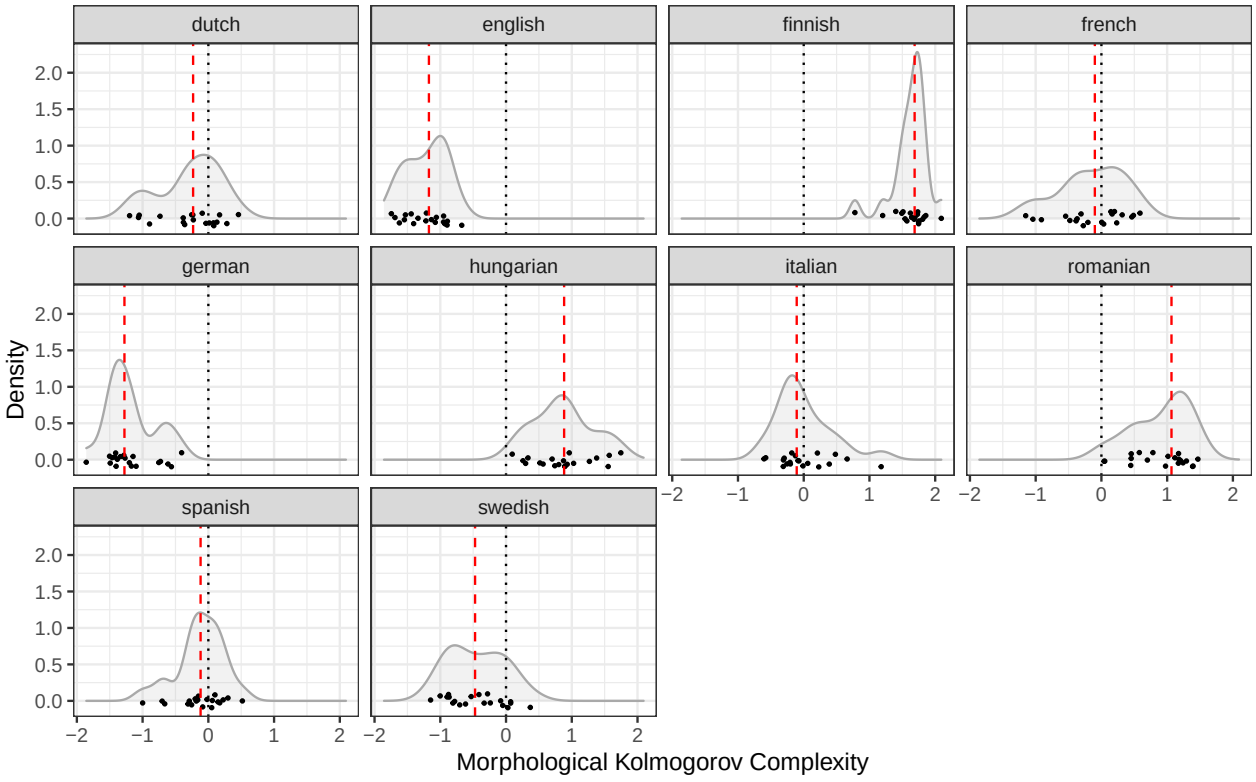
Choose particular languages to plot (if necessary).

```
#selection <- c("french", "spanish", "english")
#results.long.selected <- results.long[results.long$language %in% selection, ]
```

Plot density distributions with indication of central tendency.

```
density.plot <- ggplot(results.long, aes(x = value)) +
 #geom_histogram(aes(y = ..density..), colour = "black", fill = "light grey",
                #binwidth = 0.1) +
 geom_density(alpha = .2, fill = "grey", color = "darkgrey") +
```

```
  geom_jitter(data = results.long, aes(x = value, y = 0),
              size = 0.5, height = 0.1, width = 0) +
  facet_wrap(~ language) +
  #geom_vline(data = mu, aes(xintercept=grp.mean),
  #           linetype = "dotted") +
  geom_vline(data = med, aes(xintercept = grp.median),
               linetype = "dashed", color = "red") +
  geom_vline(aes(xintercept = 0), linetype = "dotted") +
  labs(x = "Morphological Kolmogorov Complexity", y = "Density") +
  theme_bw()
print(density.plot)
```



Safe figure to file.

```
ggsave("Figures/kolmogorov_densities.pdf", density.plot, dpi = 300, scale = 1,
       device = cairo_pdf)
```

```
## Saving 8 x 5 in image
```

Descriptive Statistics

Give an overview of mean, median, and standard deviation values (i.e. values reflecting the location of a distribution).

```
stats.df <- cbind(mu, med[, 2], sdev[, 2])
colnames(stats.df) <- c("language", "mu", "med", "sdev")
stats.df.sorted <- stats.df[order(stats.df$language),]
# round values to two decimal places,  the "-1" excludes column 1
```

```
stats.df.sorted[, -1] <- round(stats.df.sorted[, -1], 2)
print(stats.df.sorted)
```

```
##     language    mu   med sdev
## 1      dutch -0.28 -0.23 0.48
## 2    english -1.21 -1.17 0.31
## 3    finnish  1.62  1.69 0.27
## 4     french -0.15 -0.10 0.50
## 5     german -1.17 -1.28 0.38
## 6  hungarian  0.87  0.88 0.46
## 7    italian  0.00 -0.11 0.43
## 8   romanian  0.91  1.07 0.44
## 9    spanish -0.12 -0.12 0.36
## 10   swedish -0.45 -0.47 0.43
```

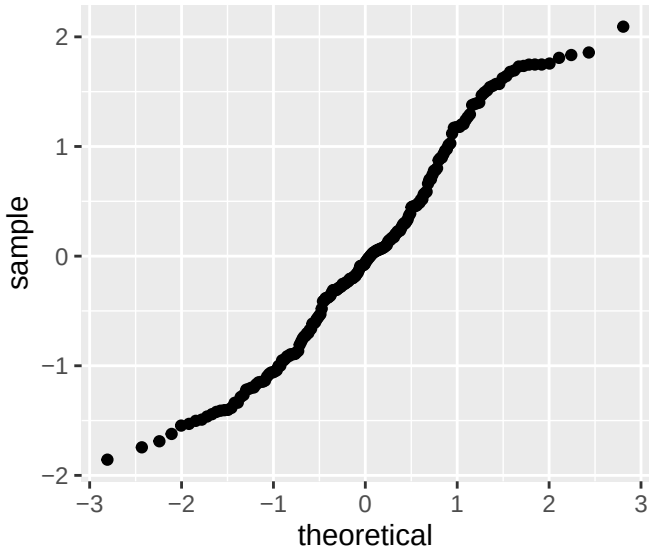
Output data frame as csv file.

```
write.csv(stats.df.sorted, file = "Tables/kolmogorov_descriptiveStats.csv", row.names = F)
```

Normality

The assumption that the tested data stems from a normally distributed population is often necessary for the mathematical proofs underlying standard statistical techniques. We might apply normality tests to check for this assumption (e.g. Baayen 2008, p. 73), but some statisticians advice against such pre-tests, since they are often too sensitive (MacDonald 2014, p. 133-136, Rasch et al. (2020), p. 67). In fact, Rasch et al. (2020, p. xi) argue based on earlier simulation studies that almost all standard statistical tests are fairly robust against deviations from normality. In a similar vein, Lumley et al. (2009) argue that non-normality of the data is a negligible issue with the t-test, at least for larger sample sizes, e.g. >= 100. However, especially for smaller sample sizes, it is still advisable to check for gross deviations from normality in the data. One common way of doing this is quantile-quantile plots. The points should here roughly follow a straight line (Crawley 2007, p. 281).

```
ggplot(results.long, aes(sample = value)) +
  stat_qq()
```



Statistical tests

Select a statistical test: Standard t-tests can be used to assess significant differences in the means of the pseudo-complexity distributions, if we assume that the underlying population distributions are normal. Wilcoxon tests are a non-parametric alternative, i.e. they do not make assumptions about the underlying population distribution, e.g. normality (Crawley 2007, p. 283; Baayen 2008, p. 77). Since there are some deviations from normality visible in the QQ-Plot above, we here choose a Wilcoxon test. If we supply two data vectors, then by default the command wilcox.test() runs a Wilcoxon rank sum test (for unpaired samples), and with "paired = T" a Wilcoxon signed rank test. Note that the same measurement procedure was used here across different languages (and corresponding text samples) to assess morphological complexity. We thus consider the resulting vectors "paired". A more general term is "related samples", which are defined as "two sets of data where a data point in one set has a pairwise relationship to a point in the other set of data" (Cahusac 2021, p. 56). Note that pairwise.wilcox.test() is a function of the R-core stats package which runs multiple tests, i.e. for all groups in the "language" column in our case.

P-value adjustment for multiple comparisons: In case of multiple testing, we should account for the fact that the likelihood of finding a significant result by chance increases with the number of statistical tests. One of the most conservative methods to account for this is the so-called Bonferroni correction, i.e. multiplying the p-values with the number of tests. This method assumes that tests are independent of one another (MacDonald 2014, p. 254-260). Since we here run pairwise tests by languages, our tests are not independent (the same language is tested against others multiple times). We therefore apply the so-called Holm-Bonferroni method, which is less conservative. It does not assume independence between tests (see the descriptions in the vignette invoked by the command "?p.adjust()") .

```
p.values <- pairwise.wilcox.test(results.long$value, results.long$language,
                    paired = T, p.adjust.method = "holm")
p.values
```

```
##
##  Pairwise comparisons using Wilcoxon signed rank test
##
## data:  results.long$value and results.long$language
##
##           dutch   english finnish french  german  hungarian italian romanian
## english   0.00065 -       -       -       -       -         -       -
## finnish   8.6e-05 8.6e-05 -       -       -       -         -       -
## french    1.00000 8.6e-05 8.6e-05 -       -       -         -       -
## german    0.00101 1.00000 8.6e-05 0.00053 -       -         -       -
## hungarian 0.00011 8.6e-05 0.00419 0.00025 8.6e-05 -         -       -
## italian   1.00000 8.6e-05 8.6e-05 1.00000 0.00011 0.00200   -       -
## romanian  0.00011 8.6e-05 0.00042 0.00042 8.6e-05 1.00000   0.00065 -
## spanish   1.00000 0.00011 8.6e-05 1.00000 8.6e-05 0.00031   1.00000 0.00025
## swedish   1.00000 0.00053 8.6e-05 0.82550 0.00200 8.6e-05   0.07668 0.00025
##           spanish
## english   -
## finnish   -
## french    -
## german    -
## hungarian -
## italian   -
## romanian  -
## spanish   -
## swedish   0.64084
##
## P value adjustment method: holm
```

Effect Size

Statistical significance is only one part of the story. For instance, a difference in complexity values might be statistically significant, but so small that it is negligible for any theorizing. In fact, it is sometimes argued that effect sizes - rather than p-values - should be the aim of statistical inquiry (Cahusac 2020, p. 12-15). An overview of effect size measures per statistical test is given in Patil (2020). In conjunction with the Wilcoxon signed rank test we here use the statistic r (i.e. function wilcox_effsize() of the "rstatix" package).

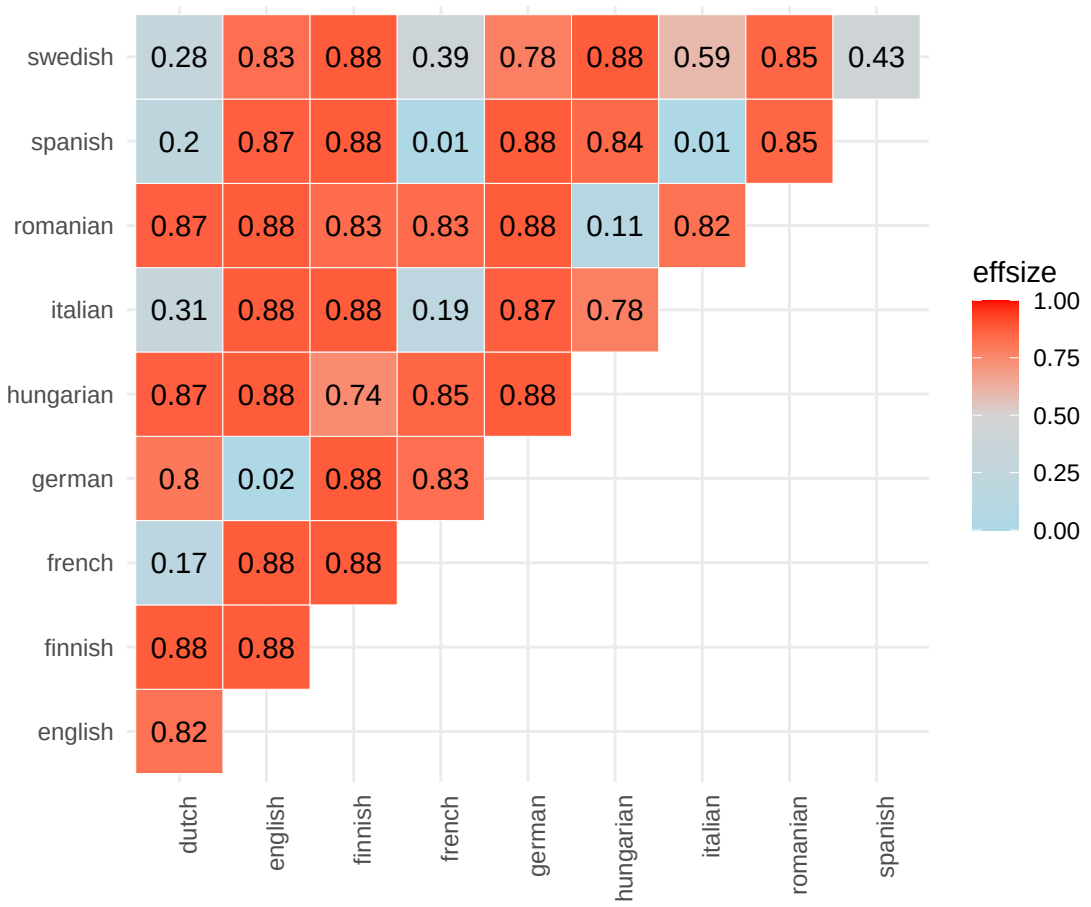
```
effect.sizes <- wilcox_effsize(results.long, value ~ language, paired = T)
print(effect.sizes)
```

```
## # A tibble: 45 x 7
##    .y.   group1  group2    effsize    n1    n2 magnitude
##  * <chr> <chr>   <chr>       <dbl> <int> <int> <ord>
##  1 value dutch   english     0.818    20    20 large
##  2 value dutch   finnish     0.877    20    20 large
##  3 value dutch   french      0.167    20    20 small
##  4 value dutch   german      0.801    20    20 large
##  5 value dutch   hungarian   0.868    20    20 large
##  6 value dutch   italian     0.309    20    20 moderate
##  7 value dutch   romanian    0.868    20    20 large
##  8 value dutch   spanish     0.200    20    20 small
##  9 value dutch   swedish     0.284    20    20 small
## 10 value english finnish     0.877    20    20 large
## # ... with 35 more rows
```

Effect Size Heatmap

Plot a heatmap with effect sizes to get a better overview.

```
effect.sizes.plot <- ggplot(as.data.frame(effect.sizes), aes(group1, group2)) +
  geom_tile(aes(fill = effsize), color = "white") +
  scale_fill_gradient2(low = "light blue", mid = "light grey", high = "red",
                       midpoint = 0.5, limit = c(0,1)) +
  geom_text(aes(label = round(effsize, 2))) +
  labs(x = "", y = "") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
effect.sizes.plot
```



Safe figure to file.

```
ggsave("Figures/kolmogorov_effectSizes.pdf", effect.sizes.plot, dpi = 300, scale = 1,
       device = cairo_pdf)
```

```
## Saving 6 x 5 in image
```

Interpretation

Descriptive Statistics

The language with highest median morphological complexity in this sample is Finnish. The language with the lowest median value is German. All other languages range in between these two extremes, with French, Italian, and Spanish having the smallest difference in median values.

Statistical significance

For most pairwise Wilcoxon tests we find $p < 0.05$. Some exceptions are the tests between Spanish, French and Italian, the test between German and English, as well as the test between Hungarian and Romanian. In other words, based on our data we should reject the null hypothesis that pairwise complexity differences are 0 in most cases. This holds for this particular number of data points ($n = 20$). In fact, if we included more data points then the pairwise differences could become significantly different from 0 also for the as yet non-significant pairs (as you can test by just adding more columns in the pre-processing code above). The

dependence of p-values on the number of data points is sometimes taken as an argument against so-called "frequentist" statistics. However, this is why it is important to include a measure of effect size.

Effect size

The r measure of effect size for many pairwise comparisons is 0.88 (which suggests that this number is a practical upper limit somewhat lower than the theoretical upper limit of 1). For Spanish compared to Italian and French it is lowest with 0.01 (we can disregard the minus sign here and look at the absolute values). According to Patil (2020) the r measure is in the range [0,1], and the yardstick for interpreting effect sizes is: 0.1-0.3 "small", 0.3-0.5 "medium", and > 0.5 "large". We would thus conclude that the difference between morphological complexities is large for most pairs of languages, while for some it is small or medium.

Alternative statistical approaches

We here used so-called "frequentist" statistical tests. A common alternative are Bayesian statistics. For the t-test, for instance, there is a Bayesian alternative proposed in Kruschke (2012). Another, less widespread alternative, are tests in the framework of "evidence-based" statistics which are based on likelihood ratios for competing hypotheses (see Cahusac, 2021, pp. 7 for a discussion). While different researchers might prefer different statistical approaches, Cahusac (2021, p. 8) states that: "If the collected data are not strongly influenced by prior considerations, it is somewhat reassuring that the three approaches usually reach the same conclusion."

References

Baayen, R. H. (2008). Analyzing linguistic data: A practical introduction using statistics in R. Cambridge University Press.

Cahusac, P. M. B. (2021). Evidence-based statistics. John Wiley & Sons.

Crawley, M. J. (2007). The R book. John Wiley & Sons Ltd.

Ehret, K. and B. Szmrecsanyi (2016). An information-theoretic approach to assess linguistic complexity. In: R. Baechler & G. Seiler (eds.), Complexity, Isolation, and Variation, 71-94. Berlin: de Gruyter.

Kruschke, J. K. (2012). Bayesian estimation supersedes the t test. Journal of Experimental Psychology.

Lumley et al. (2002). The importants of the normality assumption in large public health data sets. Annu. Rev. Public Health.

McDonald, J.H. (2014). Handbook of Biological Statistics (3rd ed.). Sparky House Publishing, Baltimore, Maryland. online at http://www.biostathandbook.com

Patil, I. (2020). Test and effect size details. online at https://cran.r-project.org/web/packages/statsExpressions/vignettes/stats_details.html.

Rasch, D., Verdooren, R., and J. Pilz (2020). Applied statistics. Theory and problem solutions with R. John Wiley & Sons Ltd.