Appendix 3: Meta-Analyses of Kolmogorov Complexity

Chris Bentz

March 09, 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". For further descriptions of the measure see Ehret & Szmrecsanyi (2016). In order to assess the variance of complexity estimations for different parts of the corpora, 20 chunks of 80 sentences each were created from the original text file of each language.

```
# 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
```

Data Pre-Processing

Scale all numerical columns to make them more commensurable.

```
results.scaled <- cbind(results[1:2], scale(results[3:6]))
```

Density Distributions

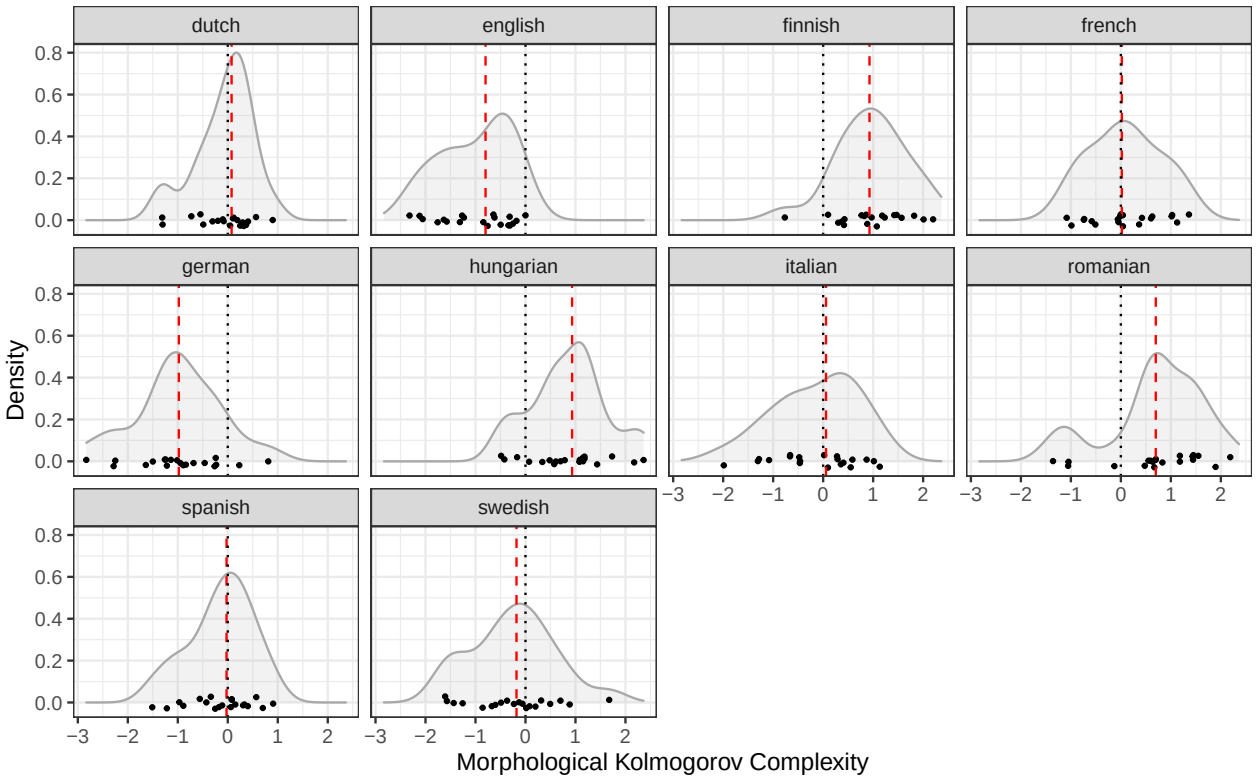
Plot density distributions of complexity measurements by language. Individual measurements are plotted as black dots. The central tendency value (i.e. mean, median, or both) of complexity measurements per language might be indicated as a red dashed line.

Get mean, median, and standard deviation values.

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

Plot density distributions with indication of central tendency.

```
density.plot <- ggplot(results.scaled, aes(x = morphratio)) +
 #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.scaled, aes(x = morphratio, y = 0),
             size = 0.7, height = 0.03, 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)
```



Save 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 -0.07  0.08 0.57
## 2    english -1.00 -0.80 0.72
## 3    finnish  0.96  0.93 0.71
## 4     french  0.09  0.02 0.72
## 5     german -0.99 -0.98 0.87
## 6  hungarian  0.85  0.93 0.77
## 7    italian -0.14  0.06 0.85
## 8   romanian  0.70  0.70 0.96
## 9    spanish -0.14 -0.03 0.64
```

```
## 10   swedish -0.25 -0.18 0.86
```

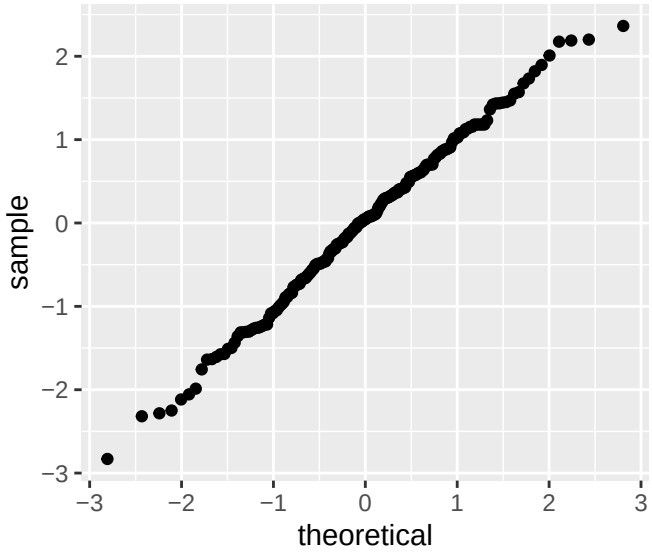
Output data frame as csv file.

```
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.scaled, aes(sample = morphratio)) +
  stat_qq()
```



Statistical tests

Select a statistical test: Standard t-tests can be used to assess significant differences in the means of 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 few deviations from normality visible in the QQ-plot above, we here run pairwise t-tests. If we supply two data vectors, then by default the function pairwise.t.test() runs a Welch two sample t-test (for unpaired samples); with the argument "paired = T" a paired t-test is invoked. Here our data consists of several samples (i.e. by language) which are linked via the same measurement procedure (i.e. Kolmogorov morphological complexity), and we hence consider them "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).

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.t.test(results.scaled$morphratio, results.scaled$language,
                    paired = T, p.adjust.method = "holm")
p.values
```

```
##
##  Pairwise comparisons using paired t tests
##
## data:  results.scaled$morphratio and results.scaled$language
##
##           dutch   english finnish french  german  hungarian italian romanian
## english   0.00072 -       -       -       -       -         -       -
## finnish   0.00080 1.8e-07 -       -       -       -         -       -
## french    1.00000 0.00739 9.7e-05 -       -       -         -       -
## german    0.00202 1.00000 3.0e-07 0.00198 -       -         -       -
## hungarian 0.00980 1.4e-05 1.00000 0.00202 3.3e-06 -         -       -
## italian   1.00000 0.02776 1.9e-07 1.00000 0.00517 0.00163   -       -
## romanian  0.09679 1.8e-05 1.00000 0.19572 6.7e-05 1.00000   0.02503 -
## spanish   1.00000 0.01603 9.3e-05 1.00000 0.02772 0.00517   1.00000 0.03635
## swedish   1.00000 0.05563 9.2e-06 0.83257 0.05790 0.00044   1.00000 0.02433
##           spanish
## english   -
## finnish   -
## french    -
## german    -
## hungarian -
## italian   -
## romanian  -
## spanish   -
## swedish   1.00000
##
## 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 t-test we here use Cohen's d (i.e. function cohens_d() of the "rstatix" package). (Note: the d estimate given by this function can be negative. However, the sign is not relevant here, only the absolute value.)

```
effect.sizes <- cohens_d(results.scaled, morphratio ~ language, paired = T)
print(effect.sizes)
```

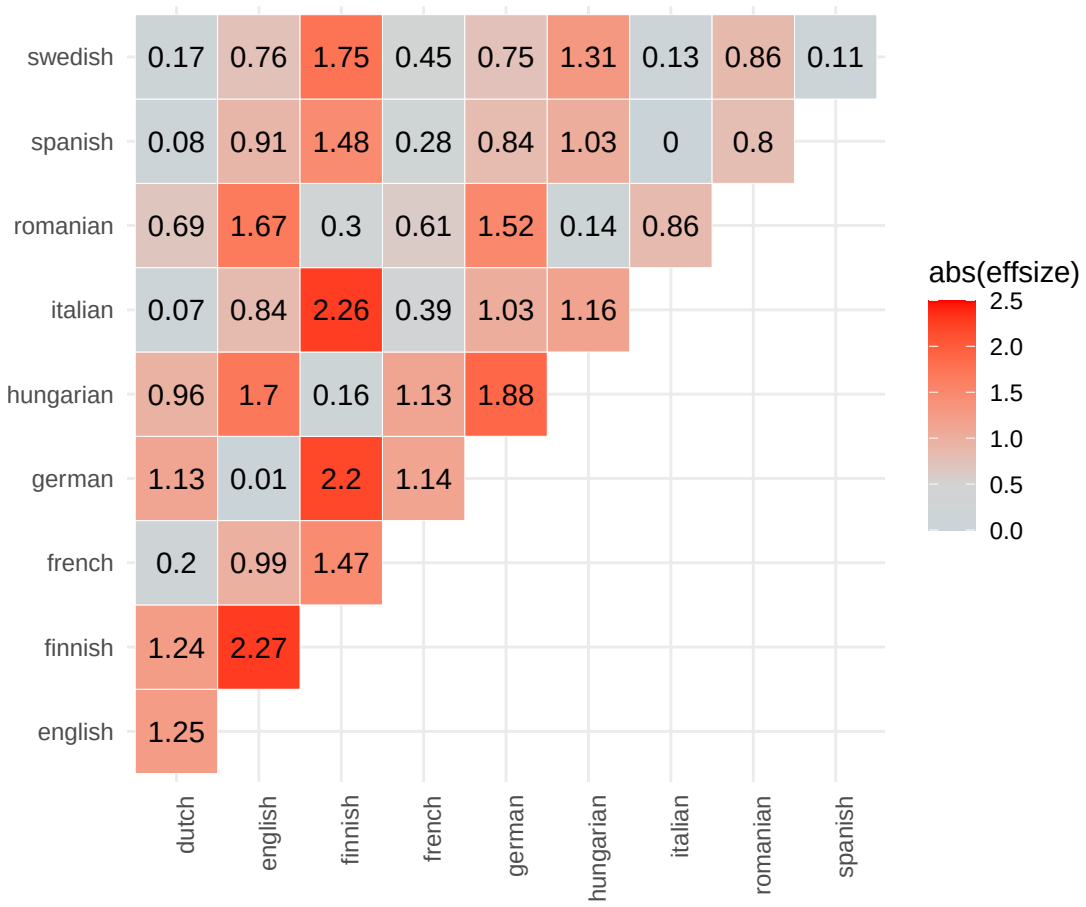
```
## # A tibble: 45 x 7
##    .y.        group1  group2    effsize    n1    n2 magnitude
```

```
##   * <chr>      <chr>   <chr>       <dbl> <int> <int> <ord>
##   1 morphratio dutch   english    1.25      20    20 large
##   2 morphratio dutch   finnish   -1.24      20    20 large
##   3 morphratio dutch   french    -0.198     20    20 negligible
##   4 morphratio dutch   german     1.13      20    20 large
##   5 morphratio dutch   hungarian -0.960     20    20 large
##   6 morphratio dutch   italian    0.0705    20    20 negligible
##   7 morphratio dutch   romanian  -0.691     20    20 moderate
##   8 morphratio dutch   spanish    0.0779    20    20 negligible
##   9 morphratio dutch   swedish    0.173     20    20 negligible
##  10 morphratio english finnish   -2.27      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 = abs(effsize)), color = "white") +
  scale_fill_gradient2(low = "light blue", mid = "light grey", high = "red",
                       midpoint = 0.5, limit = c(0, 2.5)) +
  geom_text(aes(label = round(abs(effsize), 2))) +
  labs(x = "", y = "") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
effect.sizes.plot
```



Save 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 languages with the highest median morphological complexities in this sample are Hungarian and Finnish. The languages with the lowest median values are English and German. All other languages range in between these two extremes.

Statistical significance

For most pairwise t-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. The dependence of p-values on the number of data points is sometimes taken as an

argument against so-called "frequentist" statistics. However, this issue can also be encountered by including measures of effect size.

Effect size

Using Cohen's d as a measure of effect size, an effect is typically considered "small" when $d < 0.2$, "medium" when $0.2 < d < 0.8$, and "large" when $d > 0.8$. Sometimes "very large" is attributed to $d > 1.3$ (Cahusac 2021, p. 14). In our empirical data of morphological complexity measurements we find everything from negligible to very large effect sizes. For example, the difference between Italian and Spanish has an effect size of virtually zero, while the effect size is small between Swedish and Dutch (0.17), medium between French and Swedish (0.45), large between Spanish and Hungarian (1.03), and very large between English and Finnish (2.27).

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.

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.