Supplementary Material 2: Correlation Matrices

Measuring and assessing indeterminacy and variation in the morphology-syntax distinction

Sandra Auderset Adam J.R. Tallman

In this file we provide the code as well as some additional explanation and plots of the correlation matrices (cf. Section 5 of the paper).

Data preparation

First, we load the necessary packages and set the seed to make the analysis fully reproducible (cf. the Rmd file for details).

```
# set seed
set.seed(200)
```

Next we load data (available at DOI), excluding columns we do not need for the analysis, i.e. we exclude the variables that are summarized in Exponence Complexity and certain meta information about the languages such as affiliation and glottocode.

```
db <- read_csv(here("SM1_Database.csv")) %>%
    select(Language, PMfixed:EXPcomplex)
```

```
## Rows: 767 Columns: 16
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): Language, Glottocode, Family, Morphs, GlossAuthor, ClassAuthorSimp...
## dbl  (5): EXPcomplex, AMsegment, AMupplN, EXPmultN, MSTfossilN
##
## 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.
```

```
glimpse(db)
```

```
## Rows: 767
## Columns: 7
## $ Language   <chr> "AsheninkaPerene", "AsheninkaPerene", "AsheninkaPerene", "A~
## $ PMfixed    <chr> "y", "y", "y", "y", "y", "y", "y", "y", "y", "y", "y", "y",~
## $ FREE       <chr> "n", "n", "n", "y", "n", "n", "y", "n", "n", "y", "n", "n",~
## $ INTERone   <chr> "y", "y", "y", "n", "n", "y", "y", "y", "y", "y", "y", "y",~
## $ PRM        <chr> "both", "both", "both", "both", "both", "both", "both", "bo~
## $ CODelab    <chr> "y", "y", "y", "y", "y", "y", "y", "y", "n", "y", "y", "y",~
## $ EXPcomplex <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
```

For computing correlation matrices we have to convert all variables to a numeric version. We do this by coding more morphology-like values as 1 and more syntax-like values as 0. An overview by variable is provided in table XX in the paper. We also rename the variables for better plotting and exclude columns not relevant for this analysis.

```
db_corr <- db %>%
    rename(EC = EXPcomplex) %>%
```

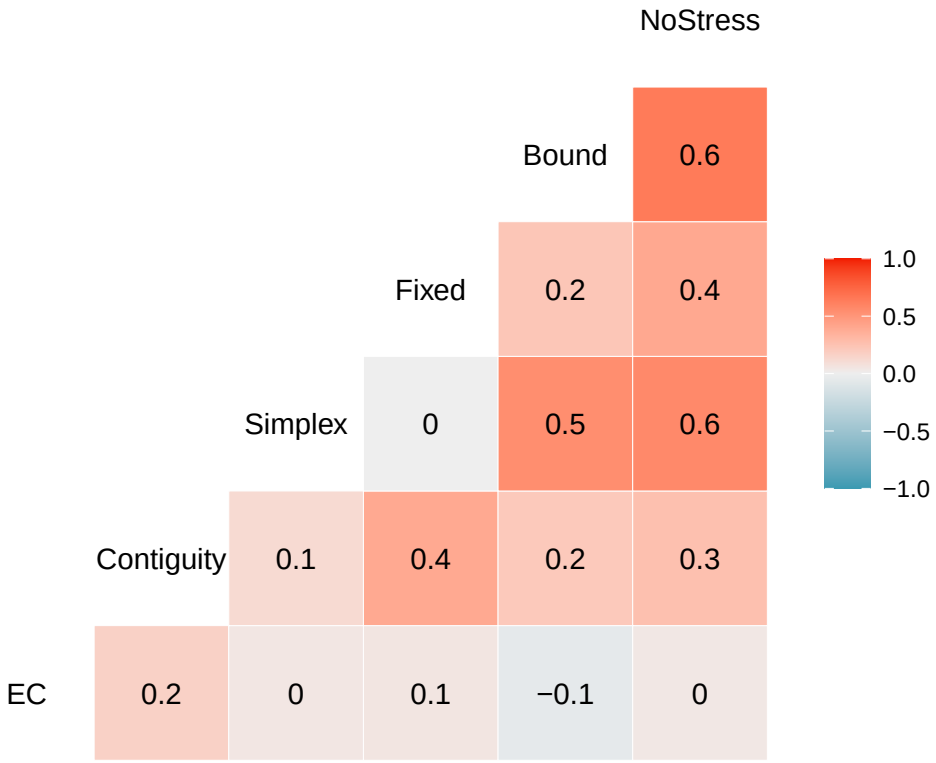
```
    mutate(Contiguity = as.integer(ifelse(INTERone == "y", 0, ifelse(INTERone ==
        "n", 1, NA)))) %>%
    mutate(Simplex = as.integer(ifelse(CODelab == "y", 0, ifelse(CODelab == "n",
        1, NA)))) %>%
    mutate(Fixed = as.integer(ifelse(PMfixed == "y", 1, ifelse(PMfixed == "n", 0,
        NA)))) %>%
    mutate(Bound = as.integer(ifelse(FREE == "y", 0, ifelse(FREE == "n", 1, NA)))) %>%
    mutate(NoStress = as.integer(ifelse(PRM == "y", 0, ifelse(PRM == "both", 1, ifelse(PRM ==
        "n", 2, NA))))) %>%
    select(Language, EC:NoStress)
glimpse(db_corr)
```

```
## Rows: 767
## Columns: 7
## $ Language   <chr> "AsheninkaPerene", "AsheninkaPerene", "AsheninkaPerene", "A~
## $ EC         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ Contiguity <int> 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,~
## $ Simplex    <int> 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0,~
## $ Fixed      <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ Bound      <int> 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0,~
## $ NoStress   <int> 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 2, 1, 1, 1, 1,~
```

Correlation matrix across all languages

We compute a correlation matrix of all variables across all languages. We do this using Kendall's Tau and pairwise comparison and plot the result.

```
cm_all <- db_corr %>%
    select(-Language) %>%
    ggcorr(., method = c("pairwise", "kendall"), label = TRUE, hjust = 0.5)
cm_all
```



Correlation matrices per language

Next we compute one correlation matrix per language to inspect cross-linguistic differences. To do this efficiently, we use a custom function that takes one variable, namely the language, and generates the correlation matrix as well as the overall correlation coefficient. We apply it to all the languages of the sample.

```
# function for correlation matrix and coefficient per language
corr.lang <- function(x) {
    lang_corr <- db_corr %>%
        filter(Language == x) %>%
        select(-Language) %>%
        ggcorr(., method = c("pairwise", "kendall"), label = "TRUE", hjust = 0.7) +
        ggplot2::labs(title = x)
    mean_corr <- round(mean(lang_corr$data$coefficient), 2)
    returnlist <- list(corr_matrix = lang_corr, corr_coeff = mean_corr)
}
# apply to each language and display plots
mov_corr <- corr.lang("Movima")
mov_corr
```

```
## $corr_matrix
```

Movima


```
##
## $corr_coeff
## [1] -0.03
```

```
pui_corr <- corr.lang("Puinave")
pui_corr
```

```
## $corr_matrix
```

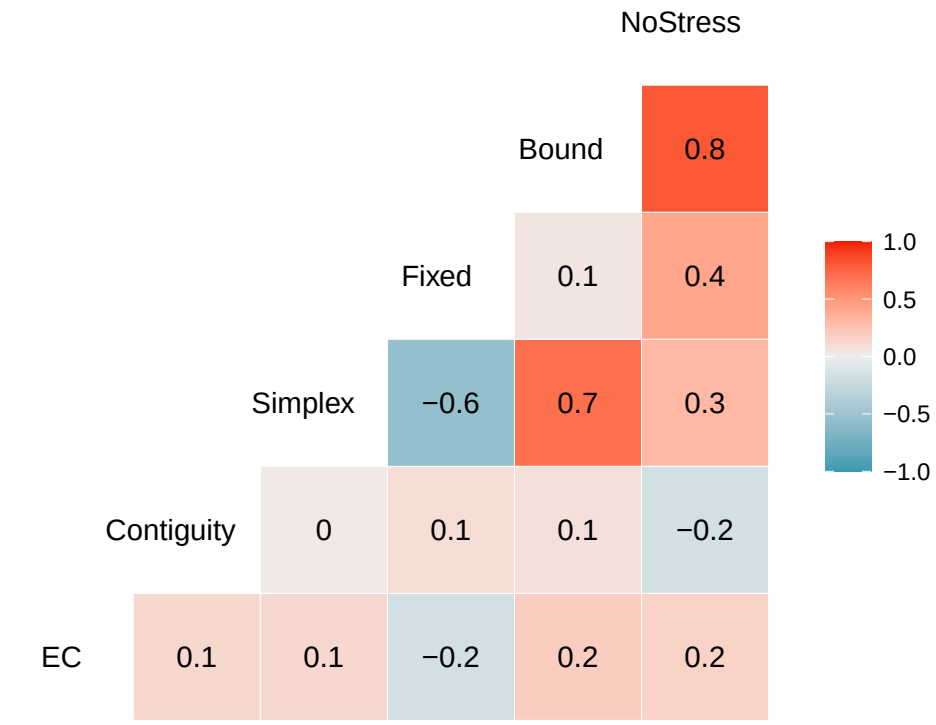
Puinave


```
##
## $corr_coeff
## [1] 0.24
```

```
tar_corr <- corr.lang("Tariana")
tar_corr
```

```
## $corr_matrix
```

Tariana

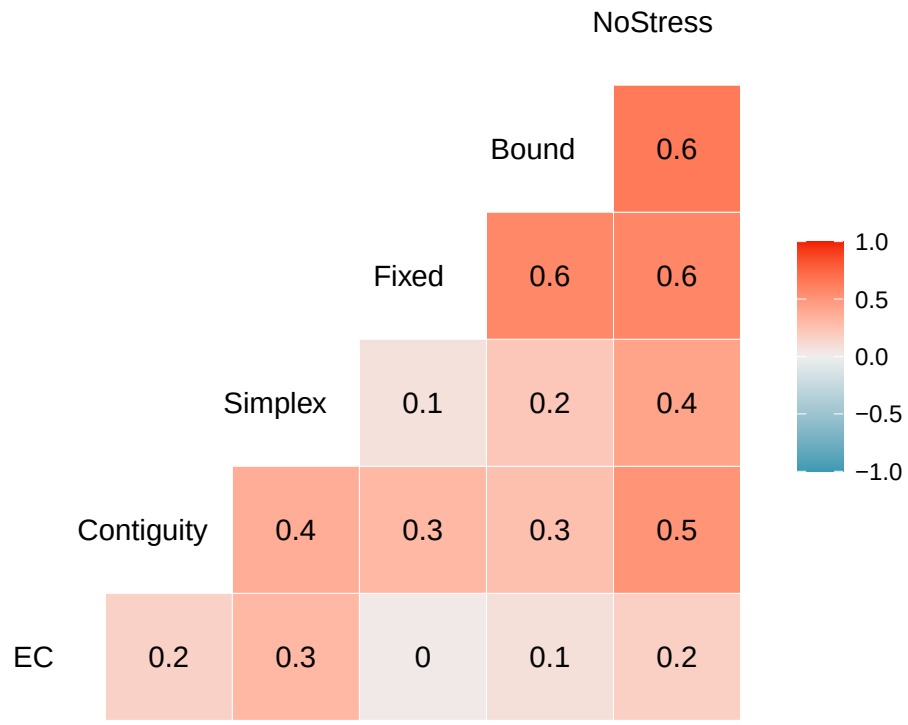



```
##
## $corr_coeff
## [1] 0.15
```

```
ash_corr <- corr.lang("AsheninkaPerene")
ash_corr
```

```
## $corr_matrix
```

AsheninkaPerene




```
##
## $corr_coeff
## [1] 0.32
```

```
cha_corr <- corr.lang("Chacobo")
cha_corr
```

```
## $corr_matrix
```

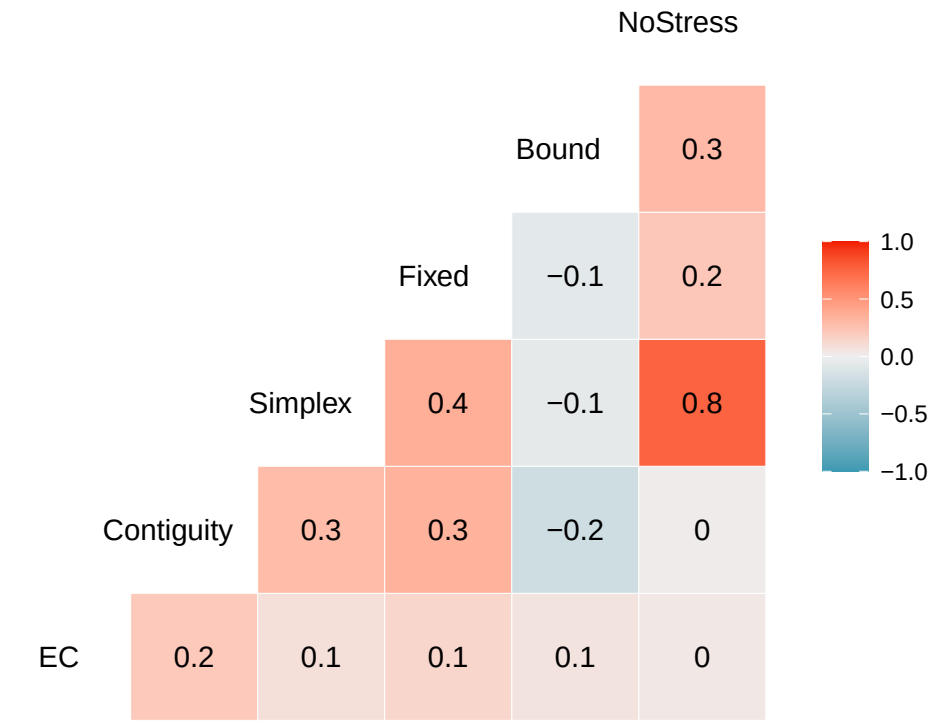
Chacobo


```
##
## $corr_coeff
## [1] 0.55
```

```
cav_corr <- corr.lang("Cavineña")
cav_corr
```

```
## $corr_matrix
```

Cavineña




```
##
## $corr_coeff
## [1] 0.17
```

```
hup_corr <- corr.lang("Hup")
hup_corr
```

```
## $corr_matrix
```

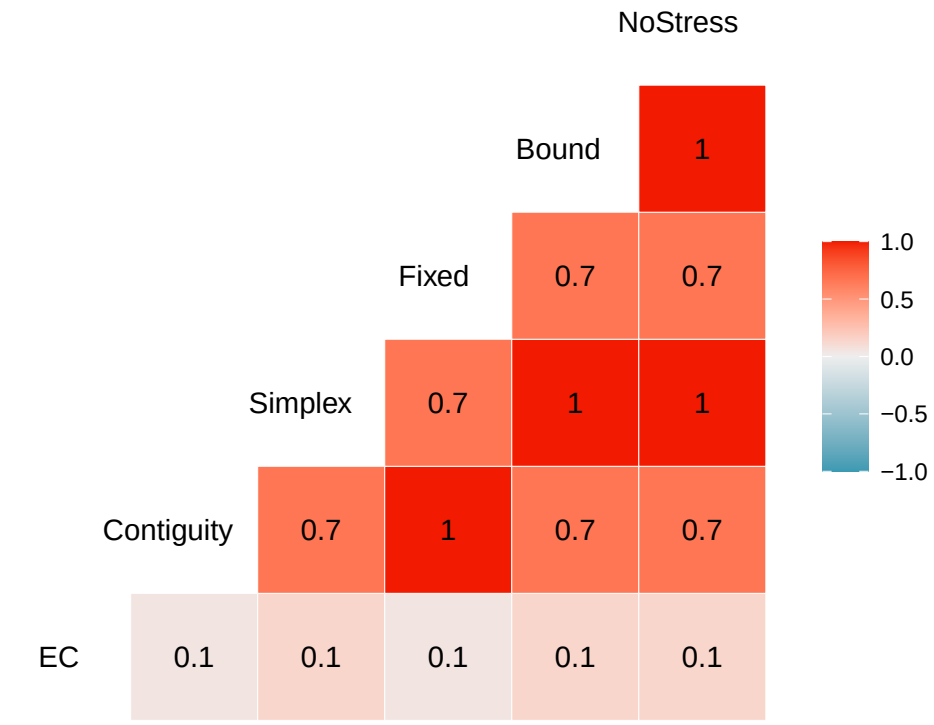
Hup


```
##
## $corr_coeff
## [1] 0.13
```

```
cay_corr <- corr.lang("CentralAlaskanYupik")
cay_corr
```

```
## $corr_matrix
```

CentralAlaskanYupik




```
##
## $corr_coeff
## [1] 0.57
```