Under the magnifying glass. Dimensions of variation in the contemporary Timok variety

Documentation

Teodora Vuković

2021-10-15

Contents

Introduction

The present document is appendix to the manuscript Under the magnifying glass. Dimensions of variation in the contemporary Timok variety.

The manuscript deals with morphosyntactic and socio-geographic variation in a South Slavic Timok variety spoken in Southeast Serbia. Four linguistic features are analysed in the context of variation between East South Slavic/Standard Serbian on the one side, and Balkan Slavic/non-standard on the other. The features selected for the analysis are:

- marking of indirect object and possessor
- post-positive demonstratives
- dative reflexive *si* as a particle
- auxiliary omission in the perfect tense

The present document follows the analysis presented in the paper and provides data and methodological processes used. It thus orderly refers to the sections and subsections from the manuscript.

For the purposes of the present paper, corpus files were searched using Python. The published online version of the corpus might provide different search options. Should the search be repeated on the uploaded version of the corpus, due to potential fine-grained changes in the data, the tendencies presented in the paper will not change, but the absolute numbers might, as well as the overall number of examples.

Note that in the present document, some pieces of code have been hidden to make it more readable. The entire code is available in the source script with the .Rmd extension.

3. Facets of variation

3.1 The analysis of morphosyntactic factors

3.1.1 Marking of indirect object and possessor

The analysis is based on the following variables:

- Dependent variable: type of marking (*na* + general oblique case vs. inflectional dative)
- Independent variables: function (indirect object, possessor), part-of-speech (nouns, pronouns, 'other'), nominal categories (proper/common nouns, grammatical number, grammatical gender, animacy)

The data used in the analysis is stored in the file 1_data.xlsx. The data was extracted from the corpus semi-automatically. Firstly Python script was used to extract all the instances of dative or *na* + noun/pronoun patterns.

00_IO_na_search.py

00_IO_dative_search.py

Noun forms were approximated using word endings for inflected and non-inflected forms. For pronouns, a list of all pronominal forms was used (see in scripts). The list of verbs was added as an additional means to enable better search and ensure that particular verbs will be retrieved (see in scripts). The obtained examples of IO are not just based on the pre-defined list of verbs, other contexts were included as well.

This data was then filtered manually example, by example. The final list of examples was labelled manually for the perametres included in the analysis. The filtered data was further segmented by focusing on particular criteria for each analysis. The overall number of examples is 895.

Frequencies of *na* 'on' + general oblique case and inflectional dative are normalized with regard to the overall number of relevant parts of speech and nominal categories retrieved from the corpus and multiplied with 10,000 in case of the PoS, gender and number, but with 1,000 in case of type of noun and animacy.

The file 1_marking_examples.xlsx is organized in sheets as follows:

1. Case, PoS, Function - rows contain examples extracted from the corpus. Columns contain information about Case, Function, PoS for each example (manually annotated)
2. IO PoS RAW - data from Case, PoS, Function, only for IO. It contains also a summary table with absolute frequencies regarding PoS.
3. POSS PoS RAW - data from Case, PoS, Function, only for POSS. It contains also a summary table with absolute frequencies regarding PoS.
4. Freq PoS tabele - repeated summary tables from 2. IO PoS RAW and 3. POSS PoS RAW, with calculated percentages, normalized per total number of the respective category.
5. Nominal categories RAW data - (for nouns only!) rows contain examples extracted from the corpus. Columns contain informtaion about nominal categories: Type of Noun (proper, common), Gender (masculine, feminine, neuter), Number (singular, plural), Animacy (animate, inanimate).
6. % for Nominal categories - Summary table based on data from 5. Nominal categories RAW data, with percentages and normalized frequencies per total number of nouns of each type/gender/number/animacy. The data for Type of Nouns is marked in yellow. The final table used for Figure 3 is highlited in red.
7. corpus_PoS_frequencies - frequencies extracted from the corpus for each PoS and nominal categories. The last row shows total frequency for each column.

In what follows analyses are presented as they appear in the paper.

Chi square test is used to compare analysed observations of analytic vs. inflectional marking in the whole sample. The test is performed using the data in the file 1_analytic_synthetic_marking.csv' which contains all examples of IO and POSS extracted from the corpus, labelled for the type of marking: analytic=0, inflectional=1 (from the file 1_data.xlsx, sheet 1. Case, PoS, Function, column Case). The values were relabelled below 0="NA+OBL", 1="DAT" here for clearer representation.

```r
head(marking_function_pos)
```

```
##   ID                 File   Case Function     PoS
## 1  1 TOR_C_00028_tagged.txt na-OBL       IO    NOUN
## 2  4  TOR_C_0080_tagged.txt na-OBL       IO    NOUN
## 3  6  TOR_C_0060_tagged.txt na-OBL       IO    NOUN
## 4  7  TOR_C_0085_tagged.txt na-OBL       IO PRONOUN
## 5  9 TOR_C_00030_tagged.txt na-OBL       IO    NOUN
## 6 12 TOR_C_00032_tagged.txt na-OBL       IO    NOUN
```

The sum of each category is used as input for Chi-square test.

```r
head(analytic_synthetic_marking_chisq)
```

```
##
##    DAT na-OBL
##    108    567
```

```r
chisq.test(analytic_synthetic_marking_chisq)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  analytic_synthetic_marking_chisq
## X-squared = 312.12, df = 1, p-value < 2.2e-16
```

Logistic regression is used to compare frequencies of analytic and inflectional type of marking with regard to their function (indirect object, possessive) and the part og speech (noun, pronoun).

```r
summary(model_function_pos_2)
```

```
##
## Call:
## glm(formula = Case ~ Function + PoS + Function:PoS, family = binomial,
##     data = marking_function_pos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4206  -0.4227  -0.4227  -0.1011   3.2490
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -2.3702     0.1909 -12.415  < 2e-16 ***
## FunctionPOSS              -2.9028     1.0199  -2.846  0.00443 **
## PoSPRONOUN                 2.9258     0.2853  10.254  < 2e-16 ***
## FunctionPOSS:PoSPRONOUN    2.3472     1.1001   2.134  0.03287 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 593.55  on 674  degrees of freedom
## Residual deviance: 387.80  on 671  degrees of freedom
## AIC: 395.8
##
## Number of Fisher Scoring iterations: 7
```

The more frequent values are taken as the baseline: na-OBL, IO, NOUN.

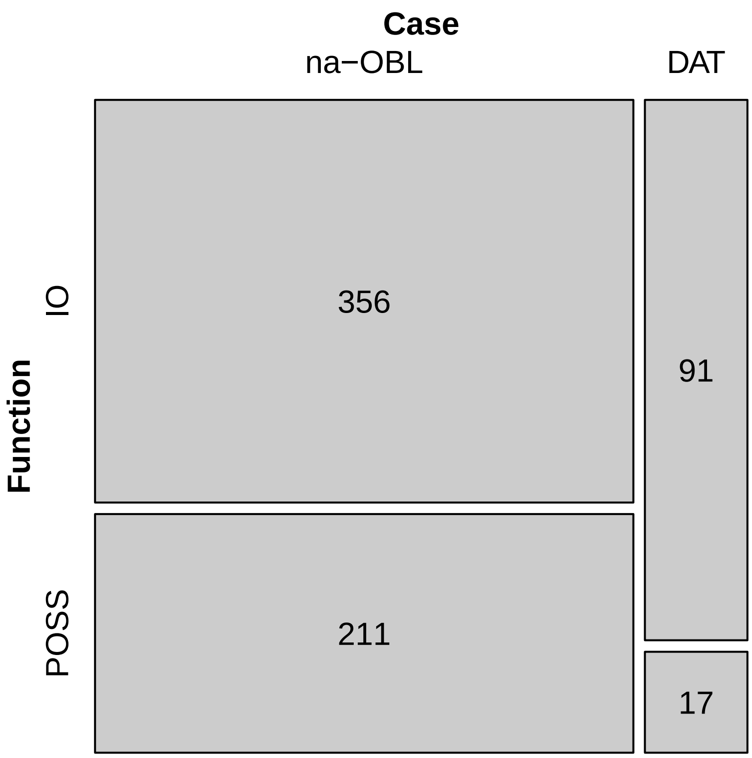
Odds ratio:

```
exp(model_function_pos_2$coefficients)
```

```
##             (Intercept)            FunctionPOSS              PoSPRONOUN
##              0.09345794              0.05487185             18.64857143
## FunctionPOSS:PoSPRONOUN
##             10.45655479
```

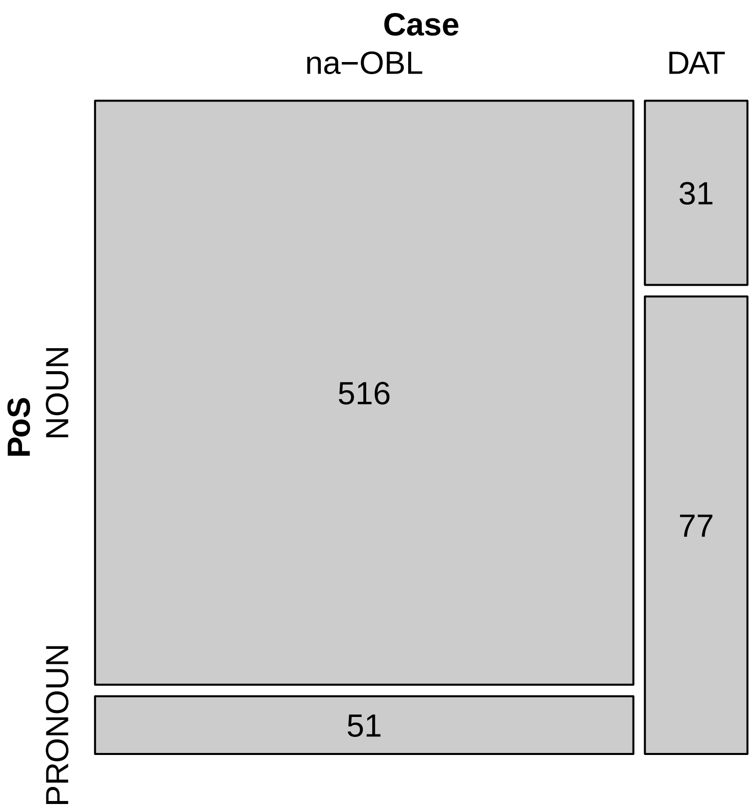
The proportion of each category is visualised in Figure 1, based on the data from the file marking_function_pos.csv. The data was obtained by categorizing each example based on the type of marking and function (see 1_data.xlsx, 1. Case, PoS, Function, columns Case and funciton).

```
# Mosaic 1: Case~function
# run all the lines of code together for the version from the paper
mosaic(case.function, main = "", direction = "v", labeling = labeling_values)
```



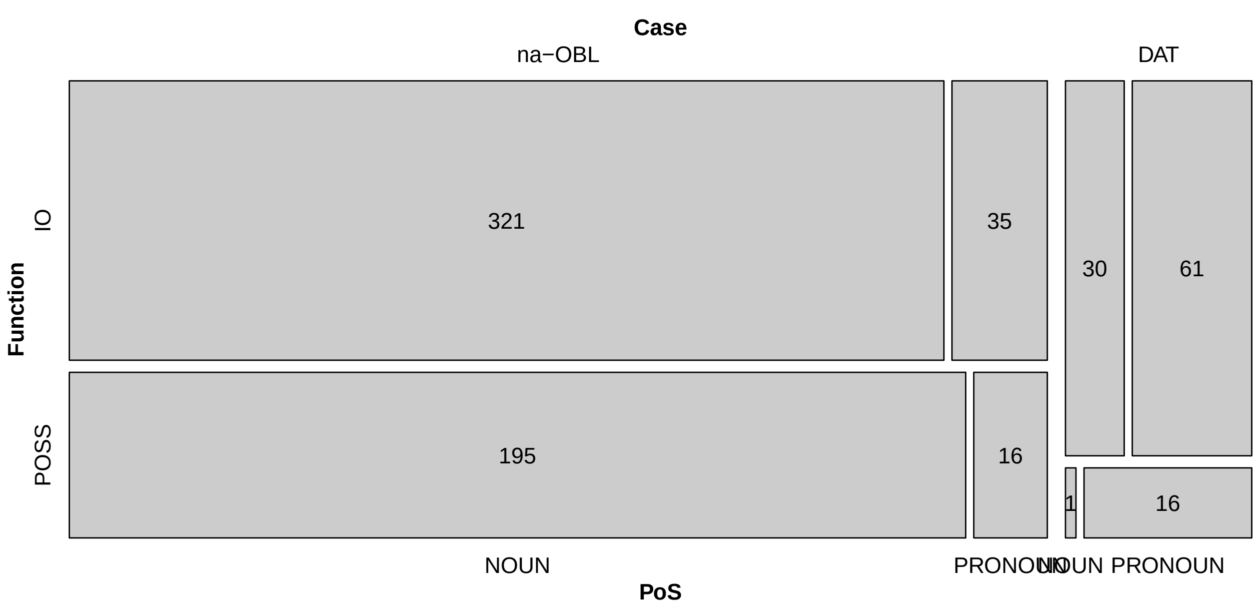

```
# grid.edit("rect:Case=DAT,Function=POSS", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Function=IO", gp = gpar(fill = "gray50"))
```

```
# Mosaic 2: Case~PoS
# run all the lines of code together for the version from the paper
mosaic(case.pos, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Case=DAT,PoS=NOUN", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,PoS=PRONOUN", gp = gpar(fill = "gray50"))
```

```
# Mosaic 3: Case~function~PoS
# run all the lines of code together for the version from the paper
mosaic(case.function.pos, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Case=DAT,Function=POSS,PoS=NOUN", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Function=POSS,PoS=PRONOUN", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Function=IO,PoS=NOUN", gp = gpar(fill = "gray50"))
```

```r
# grid.edit("rect:Case=DAT,Function=IO,PoS=PRONOUN", gp = gpar(fill = "gray50"))
```

Regression analysis of the effect of the nominal categories on the use of analytic or synthetic marking.

```r
summary(model_nominal_categories_5)
```

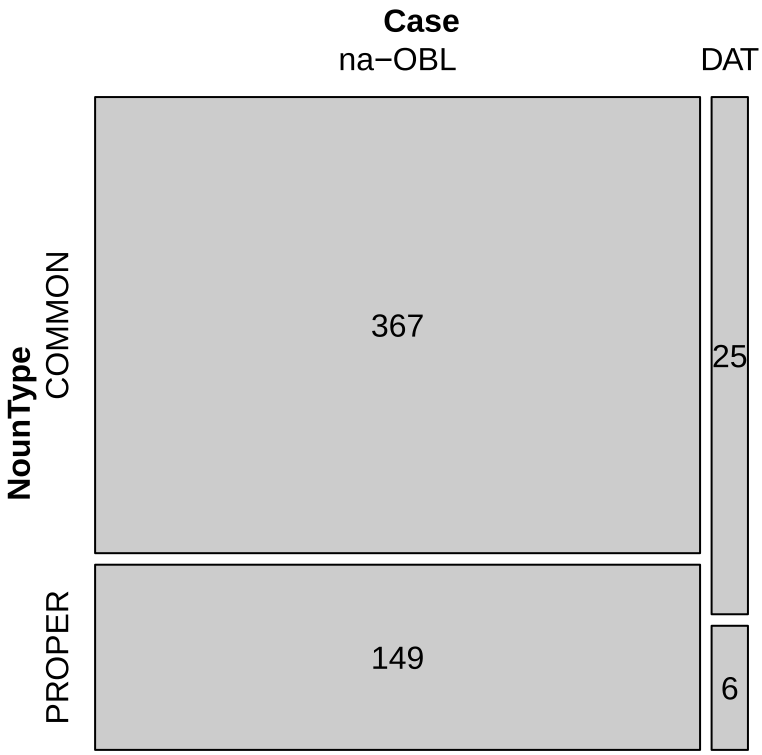
```
##
## Call:
## glm(formula = Case ~ NounType + Gender + Number + ReferenceToPersons +
##     Gender:Number + Gender:ReferenceToPersons + Gender:NounType +
##     Number:NounType, family = binomial, data = marking_nominal_categories)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9005  -0.3150  -0.1870  -0.1870   2.8477
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      -4.0372     0.5824  -6.932 4.16e-12 ***
## NounTypePROPER                    0.6931     0.8272   0.838 0.402081
## GenderMASC                        2.2454     0.6458   3.477 0.000507 ***
## GenderNEUT                      -15.5289  2109.0355  -0.007 0.994125
## NumberPL                          3.1899     0.9030   3.533 0.000412 ***
## ReferenceToPersonsNO            -17.7905  1558.3252  -0.011 0.990891
## GenderMASC:NumberPL              -3.6668     1.1229  -3.265 0.001093 **
## GenderNEUT:NumberPL              -3.1899 12023.3520   0.000 0.999788
## GenderMASC:ReferenceToPersonsNO   18.8892  1558.3257   0.012 0.990329
## GenderNEUT:ReferenceToPersonsNO   17.7905  5982.3580   0.003 0.997627
## NounTypePROPER:GenderMASC        -1.8803     1.0547  -1.783 0.074617 .
## NounTypePROPER:GenderNEUT        15.4171 15664.7062   0.001 0.999215
## NounTypePROPER:NumberPL         -16.1102  6208.8323  -0.003 0.997930
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 238.18  on 546  degrees of freedom
## Residual deviance: 202.54  on 534  degrees of freedom
## AIC: 228.54
##
## Number of Fisher Scoring iterations: 18
```

```r
exp(model_nominal_categories_5$coefficients)
```

```
##                     (Intercept)                  NounTypePROPER
##                    1.764706e-02                    2.000000e+00
##                      GenderMASC                      GenderNEUT
##                    9.444444e+00                    1.802570e-07
##                        NumberPL            ReferenceToPersonsNO
##                    2.428571e+01                    1.877868e-08
##             GenderMASC:NumberPL             GenderNEUT:NumberPL
##                    2.555781e-02                    4.117647e-02
## GenderMASC:ReferenceToPersonsNO GenderNEUT:ReferenceToPersonsNO
##                    1.597556e+08                    5.325187e+07
##       NounTypePROPER:GenderMASC       NounTypePROPER:GenderNEUT
```

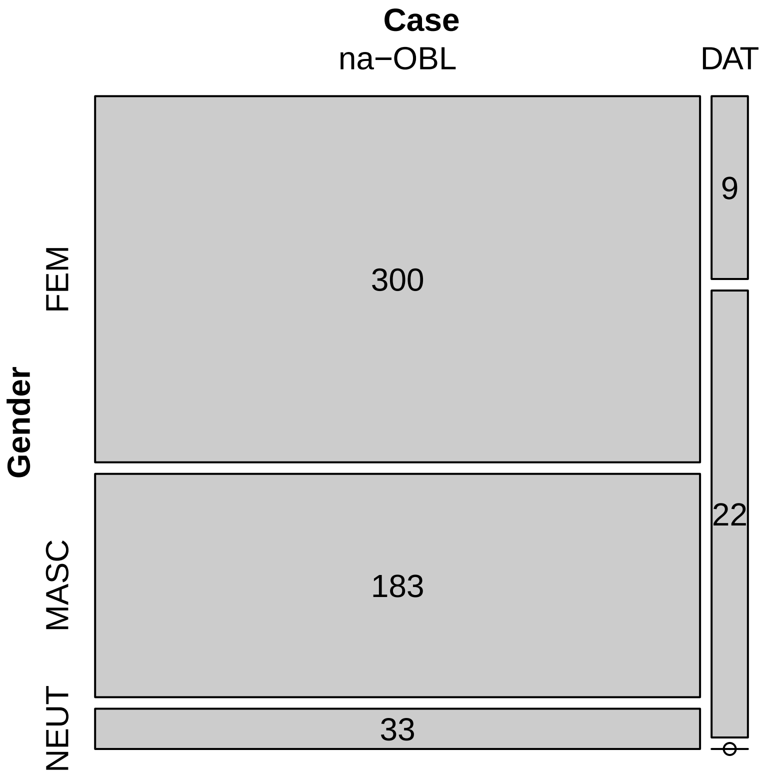
```
##                    1.525424e-01                      4.960773e+06
##         NounTypePROPER:NumberPL
##                    1.007907e-07
```

```r
# mosaic_nomcat1: Case~Noun type
# run all the lines of code together for the version from the paper
mosaic(case.nountype, main = "", direction = "v", labeling = labeling_values)
```



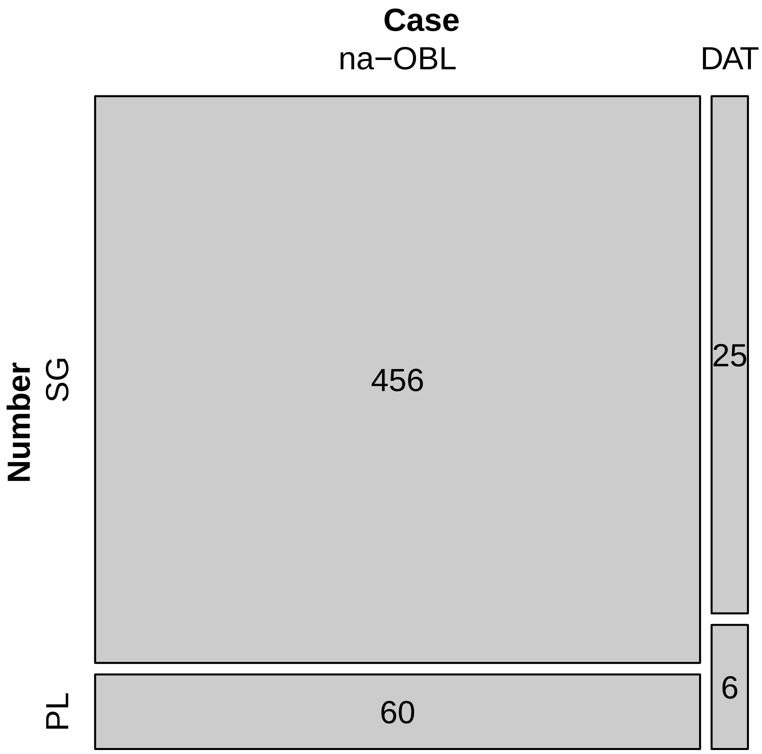

```r
# grid.edit("rect:Case=DAT,NounType=COMMON", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,NounType=PROPER", gp = gpar(fill = "gray50"))
```

```r
# mosaic_nomcat2: Case~Gender
# run all the lines of code together for the version from the paper
mosaic(case.gender, main = "", direction = "v", labeling = labeling_values)
```



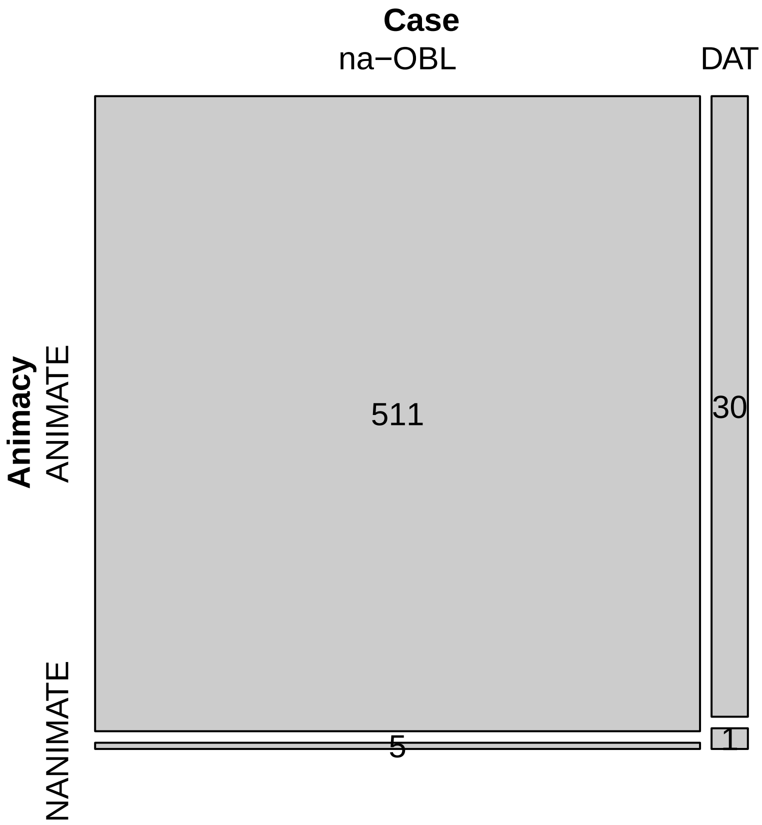

```r
# grid.edit("rect:Case=DAT,Gender=FEM", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Gender=MASC", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Gender=NEUT", gp = gpar(fill = "gray50"))
```

```r
# mosaic_nomcat3: Case~Number
# run all the lines of code together for the version from the paper
mosaic(case.number, main = "", direction = "v", labeling = labeling_values)
```



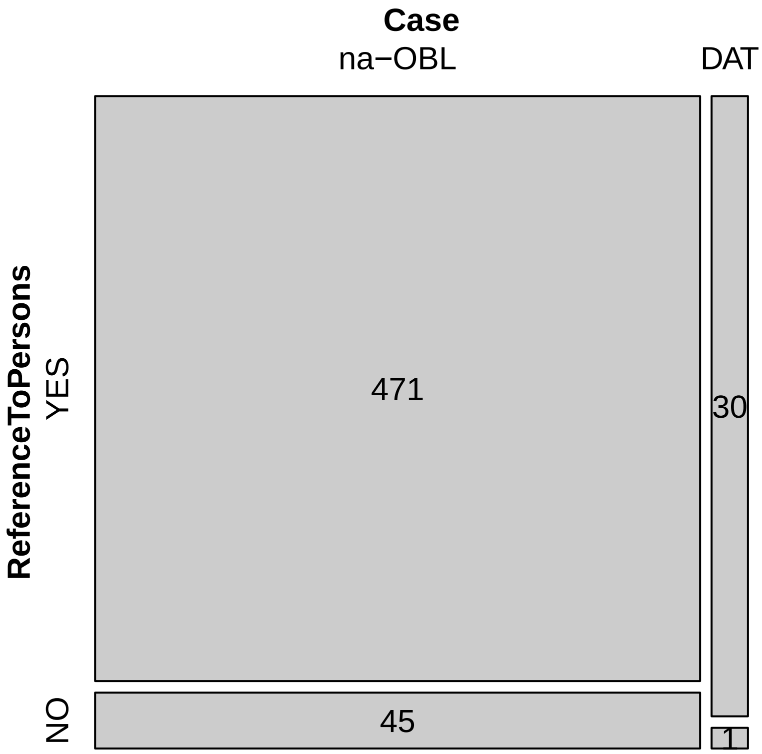

```
# grid.edit("rect:Case=DAT,Number=SG", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Number=PL", gp = gpar(fill = "gray50"))
```

```
# mosaic_nomcat4: Case~Animacy
# run all the lines of code together for the version from the paper
mosaic(case.animacy, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Case=DAT,Animacy=ANIMATE", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,Animacy=INANIMATE", gp = gpar(fill = "gray50"))
```

```
# mosaic_nomcat5: Case~Reference to persons
# run all the lines of code together for the version from the paper
mosaic(case.reftopers, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Case=DAT,ReferenceToPersons=NO", gp = gpar(fill = "gray50"))
# grid.edit("rect:Case=DAT,ReferenceToPersons=YES", gp = gpar(fill = "gray50"))
```

3.1.2 Post-positive demonstratives

In order to identify the distribution of different forms of PPD (nominative/unmarked vs. accusative/oblique, as well based on gender), nouns containing PPD were compared against bare nouns. The comparison regarding gender includes all nouns, while the comparison concerning case takes into account only nouns of the grammatical feminine gender ending in -a and masculine animate nouns ending in a consonant (regardless of the syntactic position). The following variables were used:

- Dependent variable: frequency of the nouns containing PPD and bare nouns (absolute and normalized per 10,000 nouns)
- Independent variables: gender of nouns (masculine ending in consonant, feminine ending in -a, neuter), case of nouns (nominative/unmarked and oblique/accusative singular)

Words with PPD were extracted from the corpus based on their form. The analysis in the present study involved nouns only, as explained in the manuscript. The resulting list of nouns carrying a PPD contains 1,195 tokens (in the corpus there is a total of 1,131 words of all PoS categories carrying a PPD). These words were manually annotated for PoS categories for the purposes of the analysis, because some PoS labels retreived from the corpus had been initially wrong. The examples of words containing PPD are stored in the file 2_examples_nouns_PPD.txt.The file 2_examples_all_nouns_without_PPD.txt contains all bare nouns, 79467 of them, that were derived from the corpus using only PoS tags.

For the analysis of nouns based on gender genders, the data was categorized using PoS tags.

For the analysis of gender and case inflection, the extraction of nouns of different genders was done by using lists of lemmas from each of the categories: - grammatical feminine gender (feminine and masculine nouns ending in -a) - animate masculine nouns ending in consonant The lists were made by first automatically extracting all nouns of each gender from the corpus by using PoS tags and forms, and then manually selecting only correct instances. The feminine group includes the first 1337 correct lemmas, sorted by frequency. Both masculine groups contain all lemmas retreived from the corpus fitting the criteria. The lists of lemmas are available in files 2_PPD_masculine_nouns_in_a.txt, 2_PPD_masculine_animate_nouns_in_consonant.txt, 2_PPD_feminine_nouns_in_a.txt. The number of elements in each list is shown below (not included in the manuscript).

```
lists_of_lemmas_gender
```

```
##                       Category List_size
## 1 Masculine animate in consonant       336
## 2                Feminine in -a      1337
## 3               Masculine in -a       109
```

All nouns were compared for gender, categorized based on gender and the presence of PPD. The total number of bare nouns of all genders is 74,769. The total number of nouns with PPD is 1,195. The data used in the analysis is presented in the file 2_PPD_gender_absfreq.csv.

The count of all nouns based on whether they carry a PPD:

```
table(ppd_gender$PPDSTATUS)
```

```
##
## NOPPD   PPD
## 78565  1225
```

```
chisq.test(table(ppd_gender$PPDSTATUS))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(ppd_gender$PPDSTATUS)
## X-squared = 74965, df = 1, p-value < 2.2e-16
```

Absolute frequencies of each gender in bare nouns and nouns containing a PPD are presented in the file 2_ppd_gender_noun.txt.

Regression analysis between PPD status and Gender

```
model_ppd_gender <- glm(PPDSTATUS ~ GENDER, family=binomial, data=ppd_gender)
summary(model_ppd_gender)
```

```
##
## Call:
## glm(formula = PPDSTATUS ~ GENDER, family = binomial, data = ppd_gender)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.1954  -0.1954  -0.1902  -0.1527   2.9859
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.44598    0.04828 -92.084  < 2e-16 ***
## GENDERF      0.49738    0.06322   7.868 3.62e-15 ***
## GENDERN      0.44200    0.08955   4.936 7.99e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12663  on 79789  degrees of freedom
## Residual deviance: 12596  on 79787  degrees of freedom
```

```
## AIC: 12602
##
## Number of Fisher Scoring iterations: 7
```

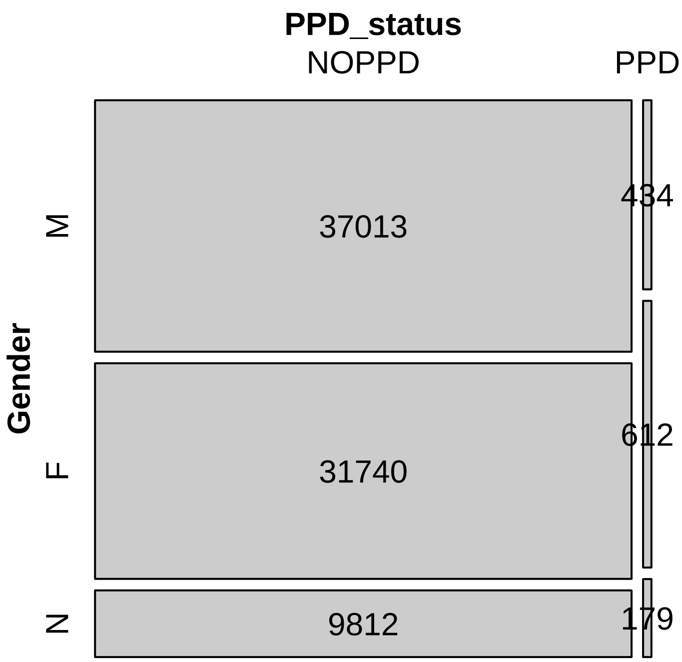
```r
exp(model_ppd_gender$coefficients)
```

```
## (Intercept)     GENDERF     GENDERN
##  0.01172561  1.64440602  1.55582250
```

Regression analysis between PPD status and Gender

Proportions of each gender in bare nouns and nouns containing a PPD is shown in Figure 4.

```r
# run all the lines of code together for the version from the paper
mosaic(ppd.gender, main = "", direction = "v", labeling = labeling_values)
```




```r
# grid.edit("rect:PPD_status=NOPPD,Gender=F", gp = gpar(fill = "gray50"))
# grid.edit("rect:PPD_status=NOPPD,Gender=M", gp = gpar(fill = "gray50"))
# grid.edit("rect:PPD_status=NOPPD,Gender=N", gp = gpar(fill = "gray50"))
```

Regression analysis between PPD status and Gender + Case

```r
model_ppd_gender2_case_1 <- glm(PPDSTATUS ~ GENDER + CASE, family=binomial, data=ppd_gender2_case)
summary(model_ppd_gender2_case_1)
```

```
##
## Call:
## glm(formula = PPDSTATUS ~ GENDER + CASE, family = binomial, data = ppd_gender2_case)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.3265  -0.3265  -0.3199  -0.3199   2.4777
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.94728    0.04547 -64.812   <2e-16 ***
```

```
## GENDERM     -0.07475    0.07913  -0.945    0.345
## CASEOBL      0.04245    0.06016   0.706    0.480
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9551.0  on 23987  degrees of freedom
## Residual deviance: 9549.3  on 23985  degrees of freedom
## AIC: 9555.3
##
## Number of Fisher Scoring iterations: 5
```

```
exp(model_ppd_gender2_case_1$coefficients)
```

```
## (Intercept)     GENDERM     CASEOBL
##  0.05248203  0.92797212  1.04335989
```

Mosaic plots 2:

```
PPD_status = ppd_gender2_case$PPDSTATUS
Gender = ppd_gender2_case$GENDER
Case = ppd_gender2_case$CASE
ppd.gender.case = xtabs(~ PPD_status + Gender + Case)

# run all the lines of code together for the version from the paper
mosaic(ppd.gender.case, main = "", direction = "v", labeling = labeling_values)
```

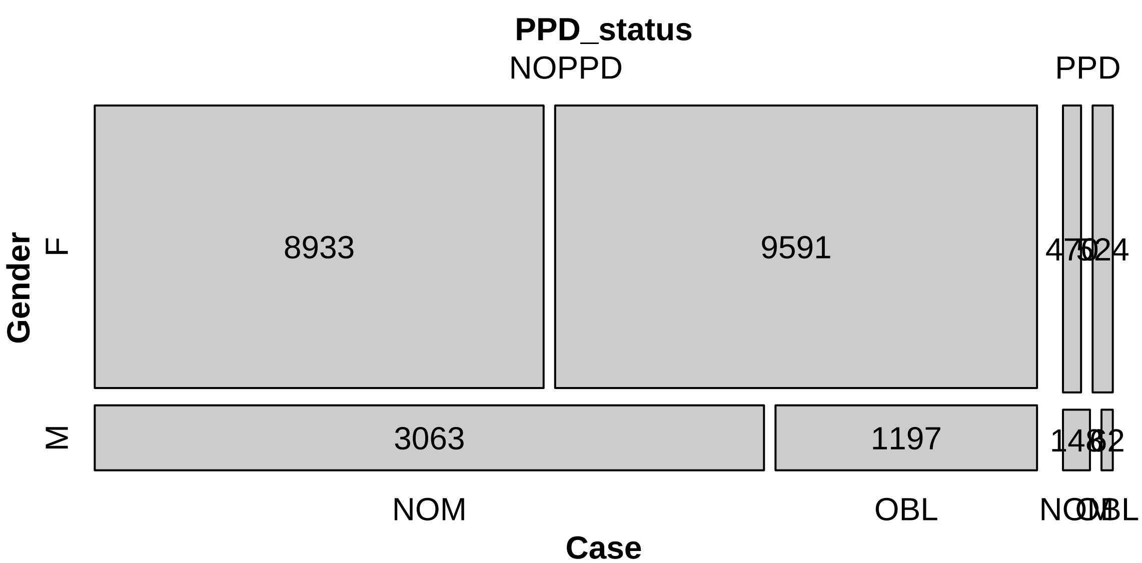


Figure 4: PPD and gender of nouns

```
# grid.edit("rect:PPD_status=NOPPD,Gender=F,Case=NOM", gp = gpar(fill = "gray50"))
# grid.edit("rect:PPD_status=NOPPD,Gender=F,Case=OBL", gp = gpar(fill = "gray50"))
# grid.edit("rect:PPD_status=NOPPD,Gender=M,Case=NOM", gp = gpar(fill = "gray50"))
# grid.edit("rect:PPD_status=NOPPD,Gender=M,Case=OBL", gp = gpar(fill = "gray50"))
```

3.1.3 Particle SI

The analysis is based on the following variables:

- Dependent variable: absolute and normalized frequency of the particle *si* used non-pronominally (per 1,000 verbs)
- Independent variables: properties of the verb (person and number, animacy, reflexivity, lexical type), variation in the syntactic patterns in the contact position between *si* and the verb

The search was done semi-automatically. A python script was used to search for all the occurences of the word 'si' and some unwanted results were excluded (such as the forms of the 2nd person auxiliary, e.g. Ti *si* gledal. 'You were watching.'). The rest was removed manually, by checking each example. Each example was annotated manually for the criteria described in the manuscript. The 1,375 examples of the use of si were extracted from the corpus. Manually annotated data used in the analysis is shown in the file 3_si_examples.xlsx

The frequency of particle *si* categorized based on person and number is shown below (see file 3_si_person.csv).

Chi-square tests:

```r
chisq.test(table(si_variables$PERSON.NUMBER))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(si_variables$PERSON.NUMBER)
## X-squared = 957.65, df = 5, p-value < 2.2e-16
```

```r
chisq.test(table(si_variables$ANIMACY))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(si_variables$ANIMACY)
## X-squared = 647.8, df = 1, p-value < 2.2e-16
```

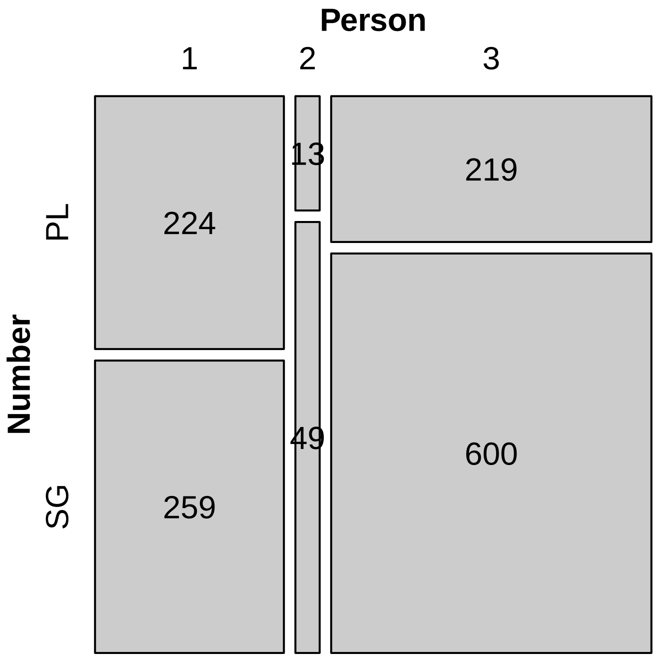
```r
chisq.test(table(si_variables$REFLEXIVITY))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(si_variables$REFLEXIVITY)
## X-squared = 926.23, df = 1, p-value < 2.2e-16
```

```r
chisq.test(table(si_variables$VOICE))
```

```
##
##  Chi-squared test for given probabilities
##
## data:  table(si_variables$VOICE)
## X-squared = 1156.6, df = 1, p-value < 2.2e-16
```

```r
# Mosaic plots:
Person = si_variables$PERSON
Number = si_variables$NUMBER
si.person.number = xtabs(~ Person + Number)

# run all the lines of code together for the version from the paper
mosaic(si.person.number, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Person=1,Number=PL", gp = gpar(fill = "gray50"))
# grid.edit("rect:Person=2,Number=PL", gp = gpar(fill = "gray50"))
# grid.edit("rect:Person=3,Number=PL", gp = gpar(fill = "gray50"))
```

Frequency of the particle *si* compared on the basis of grammatical categories: animacy of the subject, reflexivity of the predicate, voice of the predicate:

The data presenting the analysis of the position of the particle 'si' relative to the verb.

3.1.4 Auxiliary omission in perfect tense

The quantitative analysis of the use of the -AUX forms is based on the following variables:

- Dependant variable: normalized (to the total number of the examples of the use of the perfect tense) frequency of the −AUX and +AUX forms per location.
- Independant variables: gender, several categorical linguistic variables: aspect, transitivity, lexical group.

The automatic search for relevant examples in the Timok corpus made with a user Python scriptrequired all the clauses where perfect participle tense is used. These examples were automatically divided into three groups: clauses with −AUX perfect forms, clauses with +AUX perfect forms and clauses with potential mood (the latter group was subsequently excluded from the analysis). From the total number of 13,233 examples of perfect tense, 8,343 (63.05%) are −AUX forms, 4,890 (36.95%) are +AUX forms.

The file 4_overall_freq.csv shows the frequency of analysed examples of the perfect tense that display +AUX (total_aux) and -AUX (no_aux) pattern per transcript (normalized per 1,000 occurrences of the perfect tense).

```
aux = read.csv("4_aux_gramm.csv")
aux = select(aux,   "Perfect",      "Aspect",  "Transitivity", "Lex_group")
aux = mutate_all(aux, as.factor)

aux_chisq = table(aux$Perfect)
```

Chi-square/goodnes of fit test

```r
chisq.test(aux_chisq)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  aux_chisq
## X-squared = 58.039, df = 1, p-value = 2.57e-14
```

Logistic regression analysis:

```r
aux_fit = glm(Perfect ~ Transitivity + Aspect,  aux, family = "binomial")
summary(aux_fit)
```

```
##
## Call:
## glm(formula = Perfect ~ Transitivity + Aspect, family = "binomial",
##     data = aux)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2433  -1.2433  -0.6017   1.1130   2.1378
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        0.15352    0.04405   3.485 0.000492 ***
## Transitivitytrans -1.77065    0.10631 -16.656  < 2e-16 ***
## Aspectperf        -0.56071    0.11911  -4.708 2.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4337.7  on 3170  degrees of freedom
## Residual deviance: 3986.7  on 3168  degrees of freedom
## AIC: 3992.7
##
## Number of Fisher Scoring iterations: 3
```

```r
aux_fit2 = glm(Perfect ~   Transitivity +  Lex_group, aux, family = "binomial")
summary(aux_fit2)
```

```
##
## Call:
## glm(formula = Perfect ~ Transitivity + Lex_group, family = "binomial",
##     data = aux)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3665  -0.9283  -0.5978   0.9994   1.9027
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         0.43417    0.05152   8.427   <2e-16 ***
## Transitivitytrans  -1.01283    0.12151  -8.336   <2e-16 ***
## Lex_groupnon_modal -1.05289    0.08952 -11.761   <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4337.7  on 3170  degrees of freedom
## Residual deviance: 3865.3  on 3168  degrees of freedom
## AIC: 3871.3
##
## Number of Fisher Scoring iterations: 4
```

The distribution of +AUX/-AUX patterns in the overall sample is shown in Figure 9.

Figure 10: +AUX and -AUX frequencies accross speakers in the overall sample

```
Figure10
```

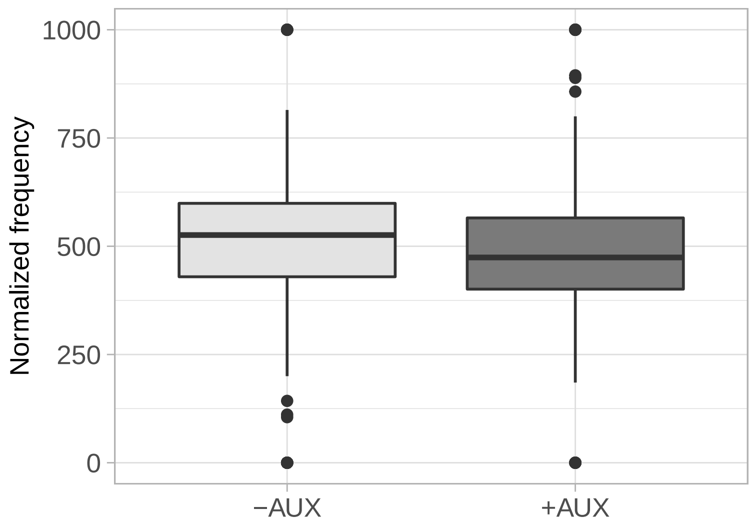
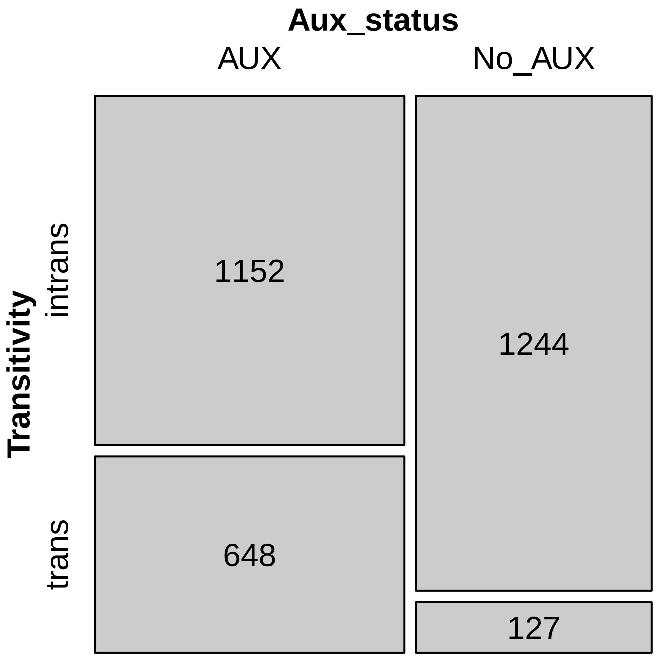


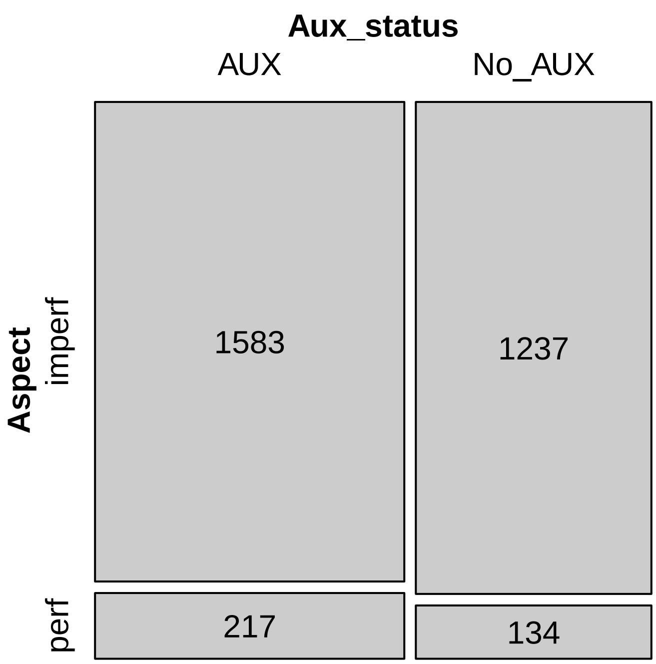
Figure 9: +AUX and -AUX frequencies in the overall sample

Auxiliary omission and verbal categories transitivity, aspect and lexical group:

```
aux_mosaics=read.delim("4_aux.csv")

# Mosaic plots:

transitivity = table(aux_mosaics$Perfect, aux_mosaics$Transitivity)
dimnames(transitivity) = list(Aux_status = c("AUX", "No_AUX"), Transitivity = c("intrans", "trans"))

mosaic(transitivity, main = "", direction = "v", labeling = labeling_values)
```

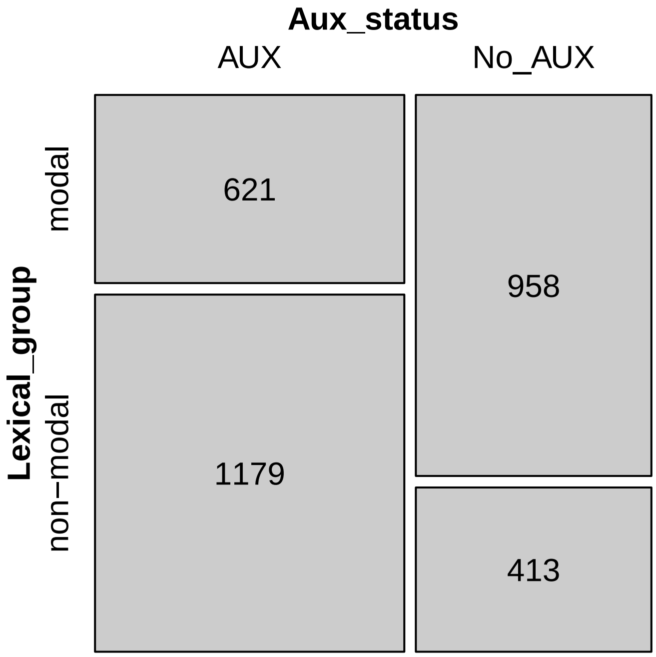



```
# grid.edit("rect:Aux_status=AUX,Transitivity=trans", gp = gpar(fill = "gray50"))
# grid.edit("rect:Aux_status=AUX,Transitivity=intrans", gp = gpar(fill = "gray50"))

aspect = table(aux_mosaics$Perfect, aux_mosaics$Aspect)
dimnames(aspect) = list(Aux_status = c("AUX", "No_AUX"), Aspect = c("imperf", "perf"))

mosaic(aspect, main = "", direction = "v", labeling = labeling_values)
```




```
# grid.edit("rect:Aux_status=AUX,Aspect=perf", gp = gpar(fill = "gray50"))
# grid.edit("rect:Aux_status=AUX,Aspect=imperf", gp = gpar(fill = "gray50"))

lexgroup = table(aux_mosaics$Perfect, aux_mosaics$Lex_group)
```

```r
dimnames(lexgroup) = list(Aux_status = c("AUX", "No_AUX"), Lexical_group = c("modal", "non-modal"))

mosaic(lexgroup, main = "", direction = "v", labeling = labeling_values)
```




```r
# grid.edit("rect:Aux_status=AUX,Lexical_group=modal", gp = gpar(fill = "gray50"))
# grid.edit("rect:Aux_status=AUX,Lexical_group=non-modal", gp = gpar(fill = "gray50"))
```

3.2 Analysis of the socio-geographic factors

Analysis of social and geographic factors involved the dependent variables:

- proportion of the analytic marking of the indirect object and the possessive per total examples analysed per location
- normalized frequency of PPD per 1,000 nouns per location
- normalized frequency of particle *si* per 1,000 verbs
- normalized frequency of AUX omission per 1,000 cases of perfect tense

The independent variables regarding geographic distribution are:

- geographic longitude
- geographic latitude
- altitude
- distance from the city of Knjaževac

The independent variables regarding socio-demographic distribution are:

- age
- gender

Analysis of the geographic factors

We firstly present the comparison of the linguistic frequencies with geographic vairiables (longitude, latitude, altitude, distance from the city). For the analysis of the geographic variables, frequency values have been aggregated for each location. The dependant variables and the geographic variables are continuous. The dependant variable in all 4 analyses does not have normal distribution, so Kendall's correlation test was used. Geographic distribution of frequencies of each feature is presented on maps. (not included in the manuscript)

Marking of indirect object and possessor:

```r
head(marking_geo)
```

```
##          LOCATION FreqNA.Oblq FreqDAT ALL FreqNA.ALL FreqDAT.ALL LATITUDE
## 1         Aldinac          10       0  10  100.00000     0.00000 43.54287
## 2       Balanovac          16      10  26   61.53846    38.46154 43.58993
## 3         Balinac          22       3  25   88.00000    12.00000 43.56462
## 4 Balta Berilovac           2       5   7   28.57143    71.42857 43.39568
## 5         Borovac          10       6  16   62.50000    37.50000 43.73822
## 6           Bu?je          23       4  27   85.18519    14.81481 43.67853
##   LONGITUDE Location Altitude DIST_Bul DIST_city
## 1  22.41992       37      623     4.42     16.44
## 2  22.13367       60      327    25.81      7.04
## 3  22.35576       19      605     7.45     11.58
## 4  22.45872       52      419     9.85     27.00
## 5  22.00940       55      199     7.99     18.68
## 6  22.09256       15      514    20.00     16.05
```

Kendall's rank correlation between analytic case marking frequencies and geographic variables.

```r
cor.test(marking_geo$FreqNA.ALL,marking_geo$LONGITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_geo$FreqNA.ALL and marking_geo$LONGITUDE
## z = 0.88652, p-value = 0.3753
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## 0.08545069
```

```r
cor.test(marking_geo$FreqNA.ALL,marking_geo$LATITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_geo$FreqNA.ALL and marking_geo$LATITUDE
## z = 0.25828, p-value = 0.7962
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## 0.02489508
```

```r
cor.test(marking_geo$FreqNA.ALL,marking_geo$Altitude, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_geo$FreqNA.ALL and marking_geo$Altitude
## z = 0.24433, p-value = 0.807
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
```

```
##        tau
## 0.02357008
```

```r
cor.test(marking_geo$FreqNA.ALL,marking_geo$DIST_city, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_geo$FreqNA.ALL and marking_geo$DIST_city
## z = -0.52355, p-value = 0.6006
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##         tau
## -0.05047776
```

```r
cor.test(marking_nouns_geo$FreqNA.ALL,marking_nouns_geo$LONGITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_nouns_geo$FreqNA.ALL and marking_nouns_geo$LONGITUDE
## z = 1.7764, p-value = 0.07567
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.1822486
```

```r
cor.test(marking_nouns_geo$FreqNA.ALL,marking_nouns_geo$LATITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_nouns_geo$FreqNA.ALL and marking_nouns_geo$LATITUDE
## z = -1.9699, p-value = 0.04885
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## -0.2020974
```

```r
cor.test(marking_nouns_geo$FreqNA.ALL,marking_nouns_geo$Altitude, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_nouns_geo$FreqNA.ALL and marking_nouns_geo$Altitude
## z = -0.0087945, p-value = 0.993
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##           tau
## -0.0009030127
```

```r
cor.test(marking_nouns_geo$FreqNA.ALL,marking_nouns_geo$DIST_city, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_nouns_geo$FreqNA.ALL and marking_nouns_geo$DIST_city
## z = 0.49248, p-value = 0.6224
```

```
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## 0.05053913
```

```r
cor.test(marking_pronouns_geo$FreqNA.ALL,marking_pronouns_geo$LONGITUDE, method = c("kendall"))
```

```
## Warning in cor.test.default(marking_pronouns_geo$FreqNA.ALL,
## marking_pronouns_geo$LONGITUDE, : Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_pronouns_geo$FreqNA.ALL and marking_pronouns_geo$LONGITUDE
## z = 1.4015, p-value = 0.1611
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.1757065
```

```r
cor.test(marking_pronouns_geo$FreqNA.ALL,marking_pronouns_geo$LATITUDE, method = c("kendall"))
```

```
## Warning in cor.test.default(marking_pronouns_geo$FreqNA.ALL,
## marking_pronouns_geo$LATITUDE, : Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_pronouns_geo$FreqNA.ALL and marking_pronouns_geo$LATITUDE
## z = 0.10992, p-value = 0.9125
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.0137809
```

```r
cor.test(marking_pronouns_geo$FreqNA.ALL,marking_pronouns_geo$Altitude, method = c("kendall"))
```

```
## Warning in cor.test.default(marking_pronouns_geo$FreqNA.ALL,
## marking_pronouns_geo$Altitude, : Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_pronouns_geo$FreqNA.ALL and marking_pronouns_geo$Altitude
## z = 0.082442, p-value = 0.9343
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## 0.01033568
```

```r
cor.test(marking_pronouns_geo$FreqNA.ALL,marking_pronouns_geo$DIST_city, method = c("kendall"))
```

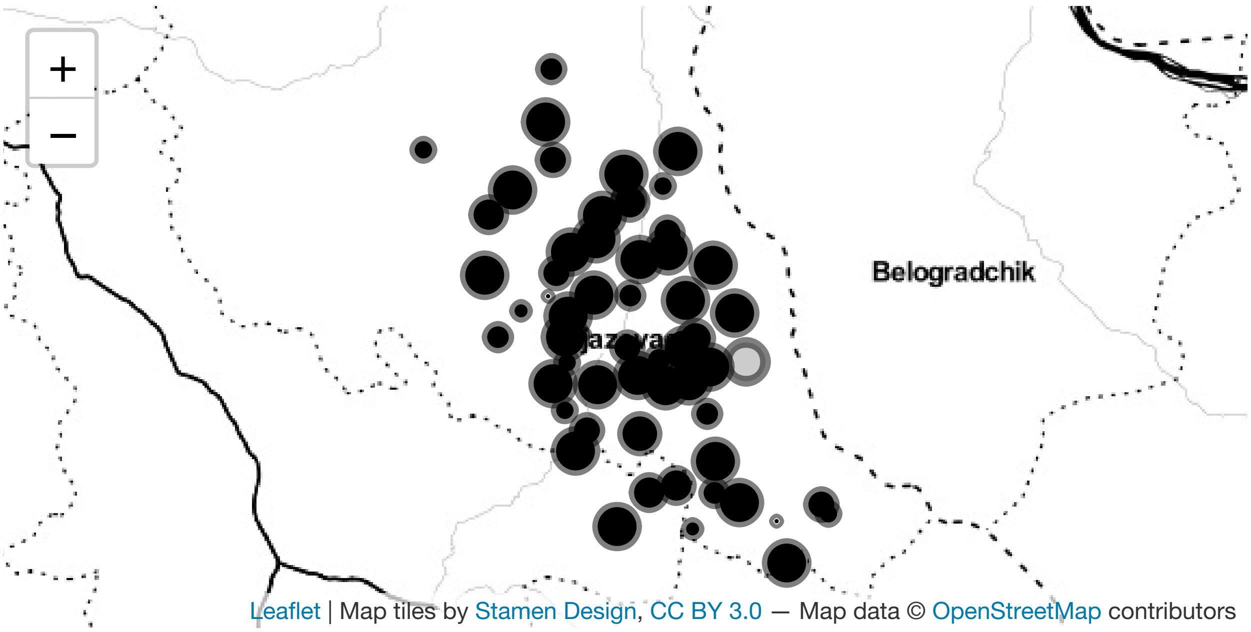
```
## Warning in cor.test.default(marking_pronouns_geo$FreqNA.ALL,
## marking_pronouns_geo$DIST_city, : Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  marking_pronouns_geo$FreqNA.ALL and marking_pronouns_geo$DIST_city
```

```
## z = -0.74203, p-value = 0.4581
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## -0.093091
```

The map presenting the areal distribution of the analytic case marking in IO and POSS:

```r
marking_map
```



Post-positive demonstratives:

```r
head(ppd_geo)
```

```
##          location art_abs_freq nouns tokens  art_norm transcript LATITUDE
## 1         Aldinac           10   831   4507 12.033694 TOR_C_0051 43.54287
## 2       Balanovac           12  1795  12604  6.685237 TOR_C_0043 43.58993
## 3         Balinac           70  2249  13104 31.124944 TOR_C_0003 43.56462
## 4 Balta Berilovac           20   569   3478 35.149385 TOR_C_0075 43.39568
## 5         Borovac            2  1089   7872  1.836547 TOR_C_0080 43.73822
## 6           Bučje           38  2645  14593 14.366730 TOR_C_0024 43.67853
##   LONGITUDE Altitude DIST_Bul DIST_city
## 1  22.41992      623     4.42     16.44
## 2  22.13367      327    25.81      7.04
## 3  22.35576      605     7.45     11.58
## 4  22.45872      419     9.85     27.00
## 5  22.00940      199     7.99     18.68
## 6  22.09256      514    20.00     16.05
```

Kendall's rank correlation between post-positive demonstatives frequencies and geographic variables.

```r
cor.test(ppd_geo$art_norm, ppd_geo$LONGITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
```

```
## data:  ppd_geo$art_norm and ppd_geo$LONGITUDE
## z = 4.8815, p-value = 1.053e-06
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.4400721
```

```r
cor.test(ppd_geo$art_norm, ppd_geo$LATITUDE, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  ppd_geo$art_norm and ppd_geo$LATITUDE
## z = -1.7495, p-value = 0.08021
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## -0.1577171
```

```r
cor.test(ppd_geo$art_norm, ppd_geo$Altitude, method = c("kendall"))
```

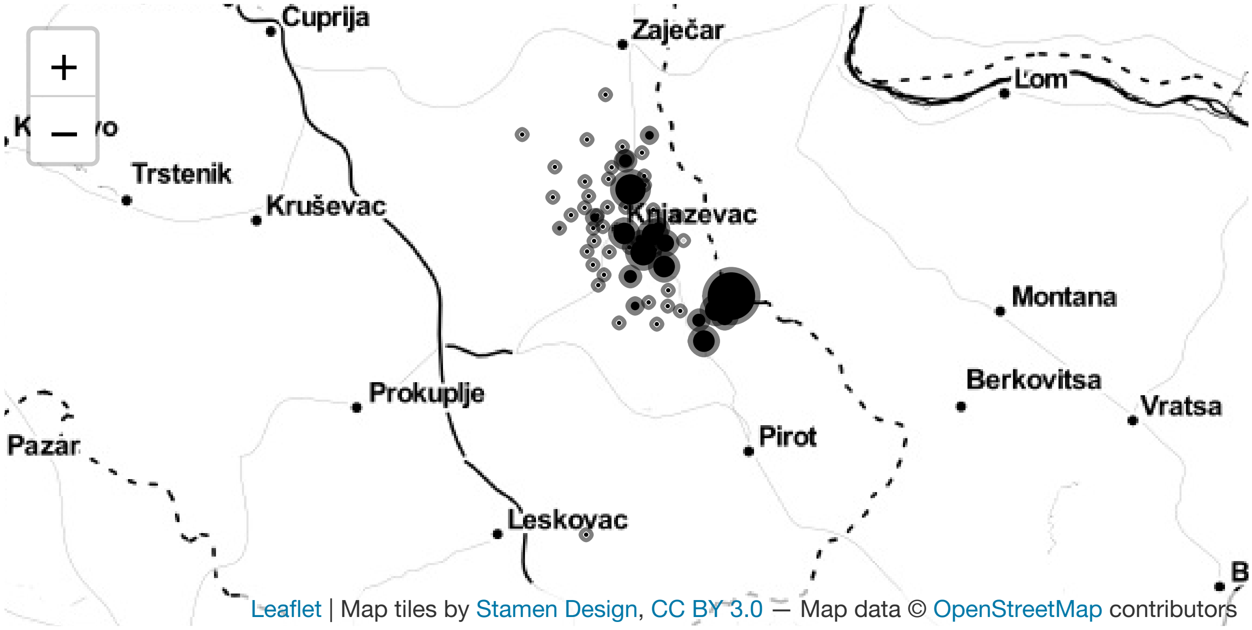
```
##
##  Kendall's rank correlation tau
##
## data:  ppd_geo$art_norm and ppd_geo$Altitude
## z = 1.6119, p-value = 0.107
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.1453549
```

```r
cor.test(ppd_geo$art_norm, ppd_geo$DIST_city, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  ppd_geo$art_norm and ppd_geo$DIST_city
## z = 1.9002, p-value = 0.0574
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.1713533
```

The map presenting the areal distribution of the post-positive demonstatives:

```r
ppd_map
```



Particle SI:

```r
head(si_geo)
```

```
##          Location NumberSI NumberVerbs NormFreqSI LATITUDE LONGITUDE Altitude
## 1         Aldinac       13        1152  11.284722 43.54287  22.41992      623
## 2       Balanovac        9        1725   5.217391 43.58993  22.13367      327
## 3         Balinac       76        3387  22.438736 43.56462  22.35576      605
## 4 Balta Berilovac        3         926   3.239741 43.39568  22.45872      419
## 5         Borovac        8        2047   3.908158 43.73822  22.00940      199
## 6           Bučje       43        3454  12.449334 43.67853  22.09256      514
##   DIST_Bul DIST_city
## 1     4.42     16.44
## 2    25.81      7.04
## 3     7.45     11.58
## 4     9.85     27.00
## 5     7.99     18.68
## 6    20.00     16.05
```

Kendall's rank correlation between particle 'si' frequencies and geographic variables.

```r
cor.test(si_geo$NormFreqSI, si_geo$LONGITUDE, method = c("kendall"))
```

```
## 
##  Kendall's rank correlation tau
## 
## data:  si_geo$NormFreqSI and si_geo$LONGITUDE
## z = 0.58201, p-value = 0.5606
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau 
## 0.05201636
```

```r
cor.test(si_geo$NormFreqSI, si_geo$LATITUDE, method = c("kendall"))
```

```
## 
```

```
##  Kendall's rank correlation tau
##
## data:  si_geo$NormFreqSI and si_geo$LATITUDE
## z = -0.25504, p-value = 0.7987
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##         tau
## -0.02279369
```

```
cor.test(si_geo$NormFreqSI, si_geo$Altitude, method = c("kendall"))
```

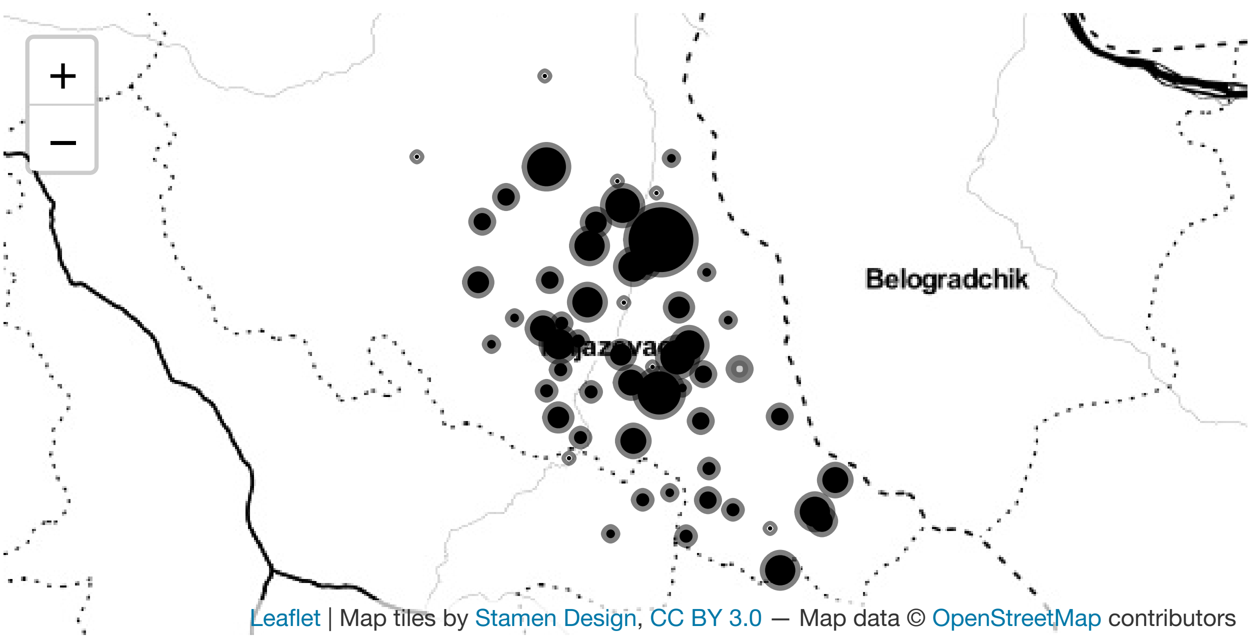
```
##
##  Kendall's rank correlation tau
##
## data:  si_geo$NormFreqSI and si_geo$Altitude
## z = 0.71283, p-value = 0.4759
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.0637427
```

```
cor.test(si_geo$NormFreqSI, si_geo$DIST_city, method = c("kendall"))
```

```
##
##  Kendall's rank correlation tau
##
## data:  si_geo$NormFreqSI and si_geo$DIST_city
## z = -0.56241, p-value = 0.5738
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##        tau
## -0.0502777
```

The map presenting the areal distribution of the particle 'si':

```
si_map
```



Auxiliary omission in the perfect tense:

```r
head(aux_geo)
```

```
##   X        LOCATION perf_count perf_aux perf_no_aux perf_no_aux_norm latitude
## 1 1         Aldinac        212      108         104         490.5660 43.54287
## 2 2       Balanovac        726      445         281         387.0523 43.58993
## 3 3         Balinac        245       88         157         640.8163 43.56462
## 4 4         Balinac        245       88         157         640.8163 43.56462
## 5 5 Balta Berilovac         95       47          48         505.2632 43.39568
## 6 6         Borovac        243       94         149         613.1687 43.73822
##   longitude Altitude DIST_city
## 1  22.41992      623     16.44
## 2  22.13367      372      7.90
## 3  22.35576      605     11.58
## 4  22.35576      605     11.58
## 5  22.45872      419     27.00
## 6  22.00940      199     18.68
```

Kendall's rank correlation between Auxiliary omission in the perfect tense frequencies and geographic variables.

```r
cor.test(aux_geo$perf_no_aux, aux_geo$longitude, method = c("pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  aux_geo$perf_no_aux and aux_geo$longitude
## t = -1.0327, df = 66, p-value = 0.3055
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3538832  0.1158094
## sample estimates:
##        cor
## -0.1260975
```

```r
cor.test(aux_geo$perf_no_aux, aux_geo$latitude, method = c("pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  aux_geo$perf_no_aux and aux_geo$latitude
## t = 0.068149, df = 66, p-value = 0.9459
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2304979  0.2463207
## sample estimates:
##        cor
## 0.00838822
```

```r
cor.test(aux_geo$perf_no_aux, aux_geo$Altitude, method = c("pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  aux_geo$perf_no_aux and aux_geo$Altitude
## t = -0.18431, df = 66, p-value = 0.8543
## alternative hypothesis: true correlation is not equal to 0
```

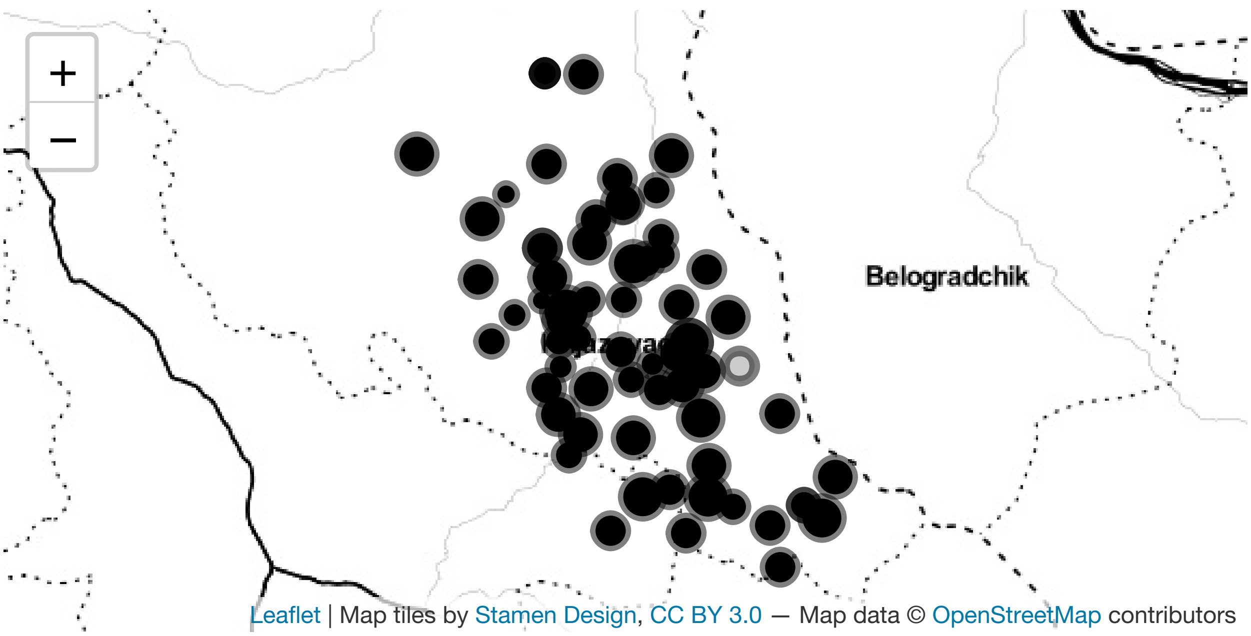
```
## 95 percent confidence interval:
##  -0.2597017  0.2169171
## sample estimates:
##         cor
## -0.02268098
```

```r
cor.test(aux_geo$perf_no_aux, aux_geo$DIST_city, method = c("pearson"))
```

```
##
##  Pearson's product-moment correlation
##
## data:  aux_geo$perf_no_aux and aux_geo$DIST_city
## t = -0.24936, df = 66, p-value = 0.8039
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2671511  0.2092758
## sample estimates:
##         cor
## -0.03068015
```

The map presenting the areal distribution of the auxiliary omission in the perfect tense:

```r
aux_map
```



Analysis of the socio-demographic factors

What follows is the correlation of the linguistic frequencies with socio-demographic variables (age, gender). For the analysis of the geographic variables, frequency values have been aggregated for each location. The dependant variables is coninuous, while the geographic variables are binary. The dependant variables in all analyses except PPD do not have normal distribution, so Wilcoxon Rank Sum test was used, while for PPD, we used Pearson's rank correlation.

Marking of indirect object and possessor:

(see file 1_marking_socio_all.csv)

Analytic marking and age:

```r
wilcox.test(marking_socio$Freq.NA.Obl..ALL[marking_socio$AGE=="older"], marking_socio$Freq.NA.Obl..ALL[m
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  marking_socio$Freq.NA.Obl..ALL[marking_socio$AGE == "older"] and marking_socio$Freq.NA.Obl..AL
## W = 304.5, p-value = 0.01454
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(marking_socio_nouns$Freq.NA.Obl..ALL[marking_socio_nouns$AGE=="older"], marking_socio_nouns$
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  marking_socio_nouns$Freq.NA.Obl..ALL[marking_socio_nouns$AGE ==  and marking_socio_nouns$Freq
## W = 134, p-value = 0.01514
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$AGE=="older"], marking_socio_
```

```
## Warning in
## wilcox.test.default(marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$AGE
## == : cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$AGE ==  and marking_socio_prono
## W = 125, p-value = 0.01873
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(marking_socio$Freq.NA.Obl..ALL[marking_socio$GENDER=="female"], marking_socio$Freq.NA.Obl..A
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  marking_socio$Freq.NA.Obl..ALL[marking_socio$GENDER == "female"] and marking_socio$Freq.NA.Obl
## W = 875.5, p-value = 0.001487
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(marking_socio_nouns$Freq.NA.Obl..ALL[marking_socio_nouns$GENDER=="female"], marking_socio_no
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  marking_socio_nouns$Freq.NA.Obl..ALL[marking_socio_nouns$GENDER ==  and marking_socio_nouns$Fr
## W = 364, p-value = 0.9918
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$GENDER=="female"], marking_so
```

```
## Warning in
## wilcox.test.default(marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$GENDER
## == : cannot compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  marking_socio_pronouns$Freq.NA.Obl..ALL[marking_socio_pronouns$GENDER ==  and marking_socio_p
## W = 356, p-value = 0.001919
## alternative hypothesis: true location shift is not equal to 0
```

```
marking_socio1<-marking_socio[!(marking_socio$AGE=="missing"),]

marking_age_plot = ggplot(data=marking_socio1)+
  geom_boxplot(aes(x=AGE, y=Freq.NA.Obl..ALL, fill = AGE))+
  labs(title = "Analytic case marking", x = NULL, y = "Relative freguency")+
  theme_light() +
  scale_fill_manual(values = c("grey89", 'grey48'))+
  theme(axis.text = element_text(size = 15),
        axis.title = element_text(size = 20),
        legend.title = element_text(size = 20),
        legend.position = "none", title = element_text(size = 20))


marking_socio2<-marking_socio[!(marking_socio$GENDER=="missing"),]

marking_gender_plot = ggplot(data=marking_socio2)+
  geom_boxplot(aes(x=GENDER, y=Freq.NA.Obl..ALL, fill = GENDER))+
  labs(title = "Analytic case marking", x = NULL, y = "Relative freguency")+
  theme_light() +
  scale_fill_manual(values = c("grey89", 'grey48'))+
  theme(axis.text = element_text(size = 15),
        axis.title = element_text(size = 20),
        legend.title = element_text(size = 20),
        legend.position = "none", title = element_text(size = 20))
```

Post-positive demonstratives:

(see files 2_PPD_age.csv and 2_PPD_gender.csv)

Post-positive demonstratives and age:

```
head(ppd_age)
```

```
##        speaker art_abs_freq nouns tokens art_norm_freq old
## 1 TIM_SPK_0104            9   584   3739     15.410959 OLD
## 2 TIM_SPK_0041            4   563   3273      7.104796 OLD
## 3 TIM_SPK_0096           10   831   4507     12.033694 OLD
## 4 TIM_SPK_0121            3   683   3568      4.392387 OLD
## 5 TIM_SPK_0012            0  1243   6825      0.000000 OLD
## 6 TIM_SPK_0113            0   253   1290      0.000000 OLD
```

Wilcoxon Rank Sum test used to compare the distribution accross OLD and YOUNG speakers.

```
wilcox.test(art_norm_freq ~ old, data = ppd_age)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  art_norm_freq by old
## W = 369, p-value = 0.001617
## alternative hypothesis: true location shift is not equal to 0
```

Post-positive demonstratives and gender:

```r
head(ppd_gender)
```

```
##   X      speaker art_abs_freq nouns tokens art_norm_freq gender female
## 1 1 TIM_SPK_0104            9   584   3739     15.410959   MALE      0
## 2 2 TIM_SPK_0041            4   563   3273      7.104796 FEMALE      1
## 3 3 TIM_SPK_0096           10   831   4507     12.033694 FEMALE      1
## 4 4 TIM_SPK_0121            3   683   3568      4.392387 FEMALE      1
## 5 5 TIM_SPK_0012            0  1243   6825      0.000000 FEMALE      1
## 6 6 TIM_SPK_0113            0   253   1290      0.000000 FEMALE      1
```

Wilcoxon Rank Sum test used to compare the distribution accross MALE and FEMALE speakers.

```r
wilcox.test(art_norm_freq ~ female, data = ppd_gender)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  art_norm_freq by female
## W = 260, p-value = 0.006433
## alternative hypothesis: true location shift is not equal to 0
```

Particle SI:

(see file 3_si_socio.csv)

Particle 'si' and age:

```r
wilcox.test(si_socio$Freq.of.SI[si_socio$AGE=="younger"], si_socio$Freq.of.SI[si_socio$AGE=="older"], al
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  si_socio$Freq.of.SI[si_socio$AGE == "younger"] and si_socio$Freq.of.SI[si_socio$AGE == "older
## W = 102, p-value = 0.07101
## alternative hypothesis: true location shift is not equal to 0
```

```r
wilcox.test(si_socio$Freq.of.SI[si_socio$GENDER=="female"], si_socio$Freq.of.SI[si_socio$GENDER=="male"]
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  si_socio$Freq.of.SI[si_socio$GENDER == "female"] and si_socio$Freq.of.SI[si_socio$GENDER == "r
## W = 841.5, p-value = 0.003746
## alternative hypothesis: true location shift is not equal to 0
```

Auxiliary omission in the perfect tense:

(see files 4_aux_age.csv and 4_aux_gender.csv)

Auxiliary omission in the perfect tense and age:

```r
head(aux_age)
```

```
##     transcript perf_count perf_aux perf_no_aux perf_no_aux_norm   old OLD_1
## 1 TIM_SPK_0052          1        1           0           0.0000   OLD     1
## 2 TIM_SPK_0030          2        1           1         500.0000   OLD     1
## 3 TIM_SPK_0042          2        1           1         500.0000   OLD     1
## 4 TIM_SPK_0120          3        1           2         666.6667   OLD     1
## 5 TIM_SPK_0167          3        3           0           0.0000 YOUNG     0
```

```
## 6 TIM_SPK_0107          5        4            1         200.0000   OLD     1
```

Wilcoxon Rank Sum test used to compare the distribution accross OLD and YOUNF speakers.

```r
wilcox.test(perf_no_aux_norm ~ OLD_1, data = aux_age)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  perf_no_aux_norm by OLD_1
## W = 305.5, p-value = 0.2502
## alternative hypothesis: true location shift is not equal to 0
```

Auxiliary omission in the perfect tense and gender:

```r
head(aux_gender)
```

```
##     transcript perf_count perf_aux perf_no_aux perf_no_aux_norm gender female
## 1 TIM_SPK_0001        127       71          56         440.9449 FEMALE      1
## 2 TIM_SPK_0003        243       86         157         646.0905 FEMALE      1
## 3 TIM_SPK_0005        118       53          65         550.8475 FEMALE      1
## 4 TIM_SPK_0007        193      106          87         450.7772 FEMALE      1
## 5 TIM_SPK_0008         32       15          17         531.2500 FEMALE      1
## 6 TIM_SPK_0009        236      114         122         516.9492 FEMALE      1
```

Wilcoxon Rank Sum test used to compare the distribution accross MALE and FEMALE speakers.

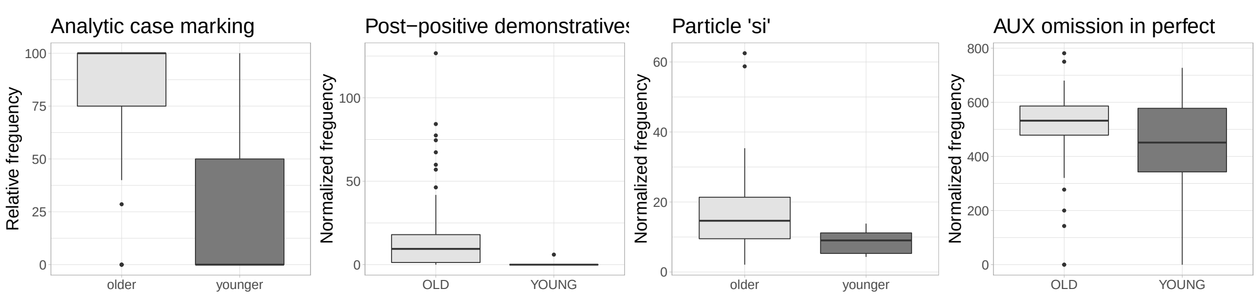
```r
wilcox.test(perf_no_aux_norm ~ female, data = aux_gender)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  perf_no_aux_norm by female
## W = 403.5, p-value = 0.007375
## alternative hypothesis: true location shift is not equal to 0
```

The ranges of values of the lingustic frequencies categorized according to age are shown in Figure 11.

Figure 11: Age

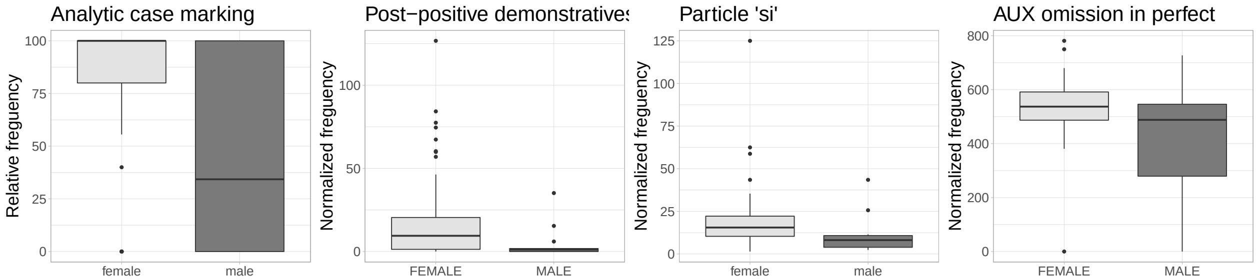
```r
Figure11 = grid.arrange(marking_age_plot, ppd_age_plot, si_age_plot, aux_age_plot, nrow = 1)
```



The ranges of values of the lingustic frequencies categorized according to gender are shown in Figure 12.

Figure 12: Gender

```r
Figure12 = grid.arrange(marking_gender_plot, ppd_gender_plot, si_gender_plot, aux_gender_plot, nrow = 1
```


Analytic case marking
Relative frequency
100
75
50
25
0
female
male
Post−positive demonstratives
Normalized frequency
100
50
0
FEMALE
MALE
Particle 'si'
Normalized frequency
125
100
75
50
25
0
female
male
AUX omission in perfect
Normalized frequency
800
600
400
200
0
FEMALE
MALE