

# Analiza studije razumijevanja riječi

Grupa *sudo*

3 May 2017

## Opis eksperimenta

Nad velikim brojem ispitanika proveden je eksperiment razumijevanja engleskog jezika. Ispitanicima su dana dva zadatka te su oba ponavljana više puta. Prvi zadatak (dalje: **A**) bavi se prepoznavanjem ispravne riječi, prilikom čega je ispitanik za zadani niz znakova morao odrediti radi li se o ispravnoj riječi engleskog jezika, a drugi zadatak (dalje: **B**) se bavi pravilnim izgovaranjem zadane riječi. Za svaku riječ i svakog ispitanika mjereno je vrijeme rješavanja svakog zadatka, te niz podataka o ispitaniku.

## Ishodi eksperimenta

Cilj eksperimenta je naučiti kako mjerene veličine ispitanika utječu na vrijeme potrebno za rješavanje pojedinih zadataka. Na temelju tih podataka može se odgovoriti na neka zanimljiva pitanja poput: utječe li dob na brzinu rješavanja zadataka, kako na brzinu rješavanja utječe duljina zadane riječi, je li riječ kraća ukoliko se češće pojavljuje, itd.

## Skup podataka

Za određivanje ishoda eksperimenta potreban nam je skup podataka eksperimenta. Programski jezik R sadrži skup podataka već provedenog eksperimenta te nam dopušta uključivanje tog skupa te analizu podataka. Podaci se nalaze u paketu `languageR`. Nakon instaliranja paketa, podaci se mogu učitati naredbom `require(languageR)` te dohvatiti s naredbom `data(english)`. Kompletно dohvaćanje i uključivanje podataka prikazano je kodom ispod.

```
require(languageR, quietly = TRUE)
data(english)
```

Podaci se sada mogu koristiti naredbom `english`, npr. deskriptivna statistika može se dobiti naredbom `summary(english)`, a pregled prvih par redova podataka može se pregledati naredbom `head(english)`.

## Ishodi eksperimenta

### Utjecaj dobi na brzinu rješavanja

Pitamo se utječe li dobna razlika između starijih i mlađih ispitanika na brzinu rješavanja zadataka? Upoređujući srednje vrijednosti logaritama vremena za rješavanje A i B zadataka mlađih i starijih ispitanika te gledajući dijagrame, možemo zaključiti da su mlađi u prosjeku brže rješavali oba zadatka. *t-testom* potvrđujemo naš zaključak.

```
young = english[english$AgeSubject == "young", ] # mlađi
old = english[english$AgeSubject == "old", ] # stari

# vrijeme potrebno mlađima za rješavanje prvog zadatka
RTlexdec_young = young[, "RTlexdec"]
```

```

# vrijeme potrebno starijima za rješavanje prvog zadatka
RTlexdec_old = old[, "RTlexdec"]

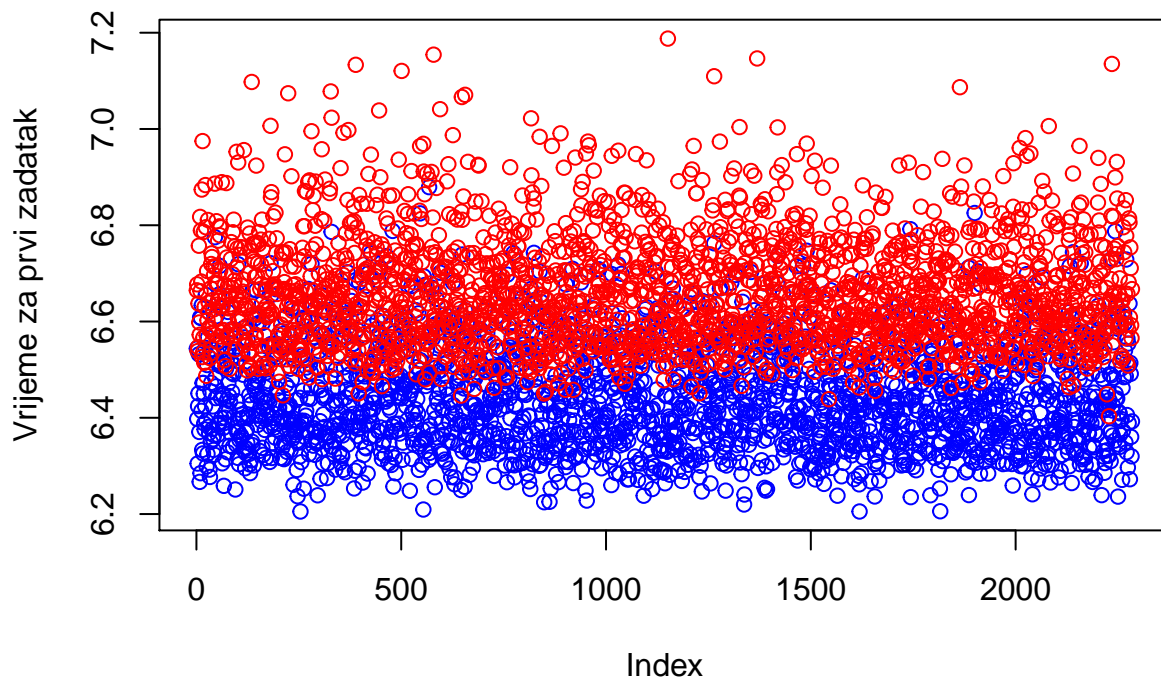
# vrijeme potrebno mlađima za rješavanje drugog zadatka
RTnaming_young = young[, "RTnaming"]

# vrijeme potrebno starijima za rješavanje drugog zadatka
RTnaming_old = old[, "RTnaming"]

plot(RTlexdec_young, col = 'blue',
     ylim = c(min(english$RTlexdec), max(english$RTlexdec)),
     ylab = "Vrijeme za prvi zadatak")

points(RTlexdec_old, col='red')

```

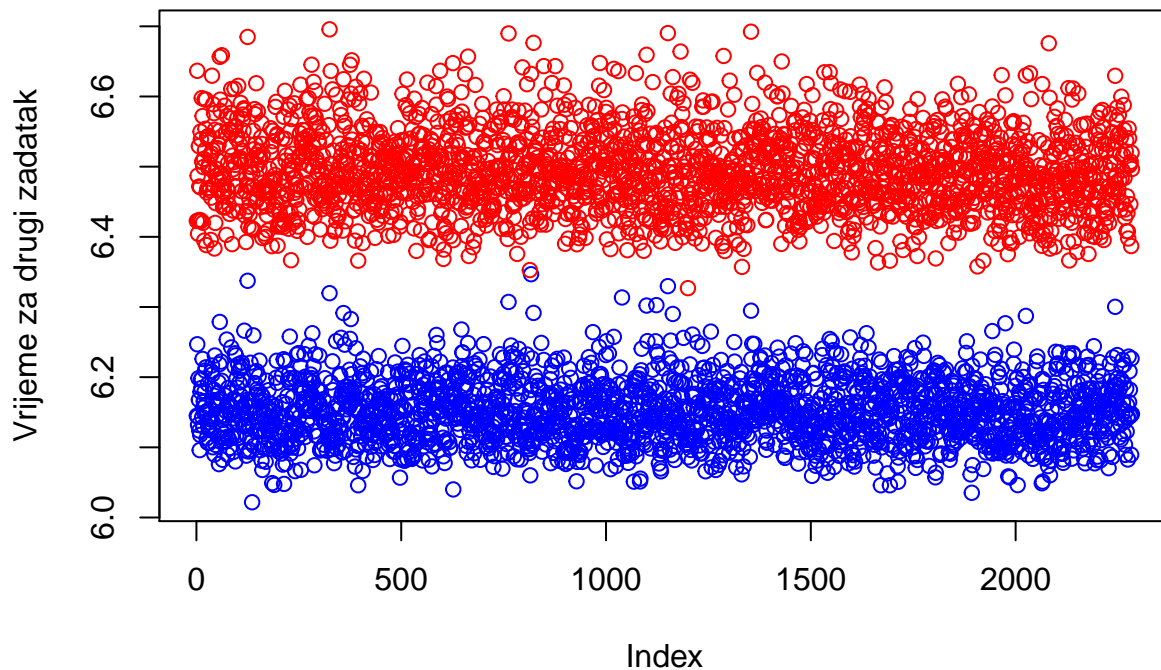


```

plot(RTnaming_young, col = 'blue',
     ylim = c(min(english$RTnaming), max(english$RTnaming)),
     ylab = "Vrijeme za drugi zadatak")

points(RTnaming_old, col='red')

```



```
# testiranje jednakosti varijance prije t-testa
var.test(RTlexdec_young, RTlexdec_old)
```

```
##
## F test to compare two variances
##
## data: RTlexdec_young and RTlexdec_old
## F = 0.84625, num df = 2283, denom df = 2283, p-value = 6.737e-05
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.7795831 0.9186270
## sample estimates:
## ratio of variances
## 0.8462542
```

```
t.test(RTlexdec_young, RTlexdec_old, alt = "less", var.equal = TRUE)
```

```
##
## Two Sample t-test
##
## data: RTlexdec_young and RTlexdec_old
## t = -67.468, df = 4566, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
## -Inf -0.2163149
## sample estimates:
## mean of x mean of y
```

```
## 6.439237 6.660958
```

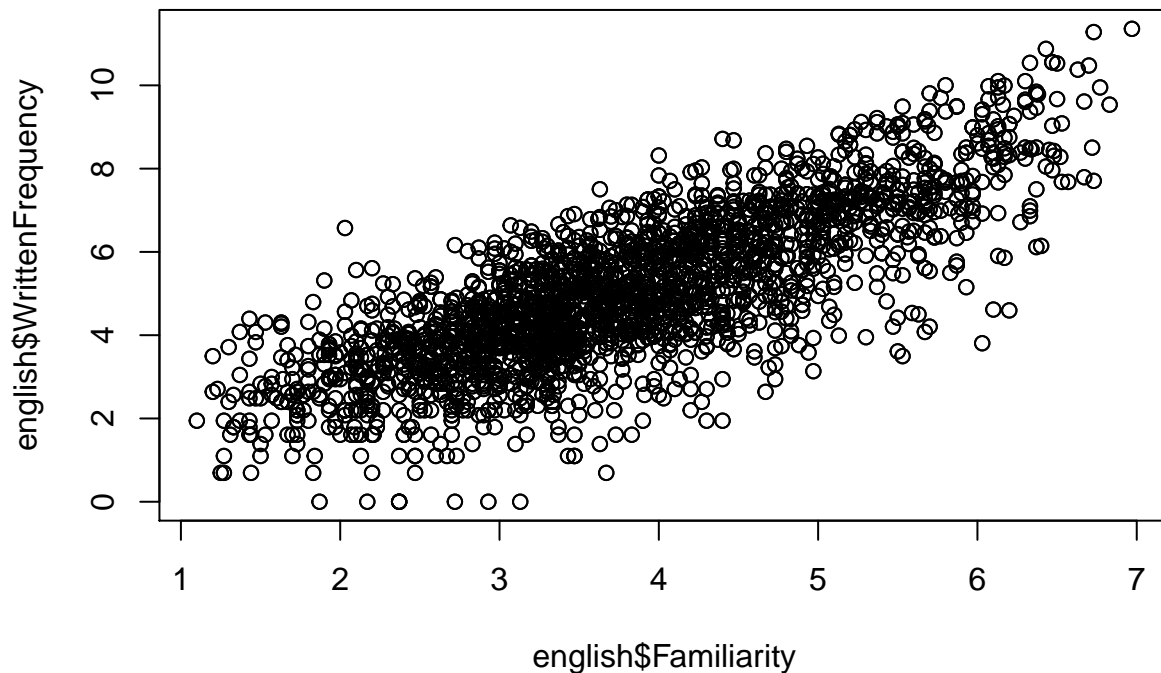
## Prepoznatljivost riječi s obzirom na frekvenciju pojavljivanja

Zanima nas jesu li riječi koje se više pojavljuju prepoznatljivije? Računamo korelaciju između prepoznatljivosti riječi i njenog pojavljivanja u tekstovima. Dobivamo korelaciju  $\sim 0.8$ , što nam potvrđuje da su te dvije stavke povezane, tj. riječi koje se više pojavljuju su prepoznatljivije. To također vidimo i iz dijagrama rasipanja.

```
cor(english$Familiarity, english$WrittenFrequency)
```

```
## [1] 0.7912556
```

```
plot(english$Familiarity, english$WrittenFrequency)
```



## Utjecaj glasa prvog slova na prepoznatljivost riječi

Je li riječ koja počinje na samoglasnik u odnosu na suglasnik ljudima prepoznatljivija? Uzimamo skup riječi koje počinju sa samoglasnikom, te skup riječi koje počinju sa suglasnikom te računamo srednju vrijednost. Kod samoglasnika dobivamo srednju vrijednost 4.0, a kod suglasnika 3.79, što bi nas moglo dovesti do zaključka da riječi koje počinju sa samoglasnikom su prepoznatljivije. No testiranjem putem *t-testa* zaključujemo da ne postoji razlika između prepoznatljivosti riječi koje počinju samoglasnikom u odnosu na one koje počinju suglasnikom.

```
firstVowel = english[english$CV == "V",] # riječi koje počinju sa samoglasnikom  
firstConsonant = english[english$CV == "C",] # riječi koje počinju sa suglasnikom
```

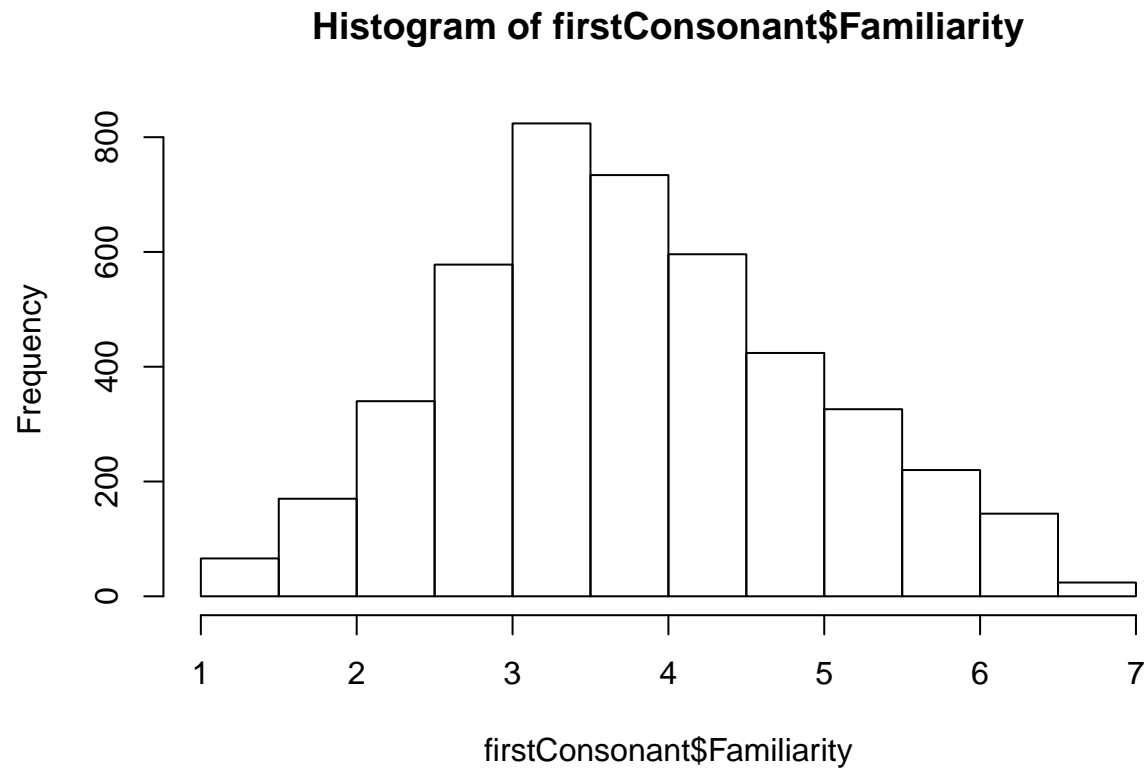
```
mean(firstConsonant$Familiarity)
```

```
## [1] 3.789892
```

```
mean(firstVowel$Familiarity)
```

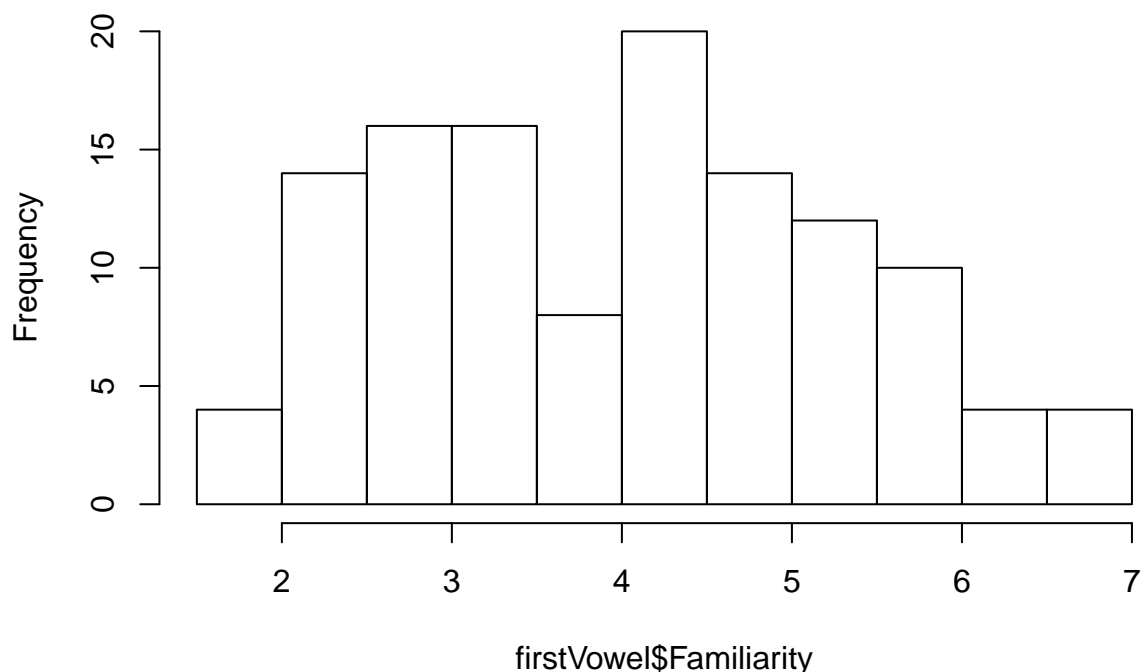
```
## [1] 4.002951
```

```
hist(firstConsonant$Familiarity)
```



```
hist(firstVowel$Familiarity)
```

## Histogram of firstVowel\$Familiarity



```
# prije testiranja t-testom trebamo zaključiti jesu li varijance jednake u oba slučaja
var.test(firstConsonant$Familiarity, firstVowel$Familiarity)
```

```
##
## F test to compare two variances
##
## data: firstConsonant$Familiarity and firstVowel$Familiarity
## F = 0.78386, num df = 4445, denom df = 121, p-value = 0.04732
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.5971133 0.9971773
## sample estimates:
## ratio of variances
## 0.7838635

t.test(firstVowel$Familiarity, firstConsonant$Familiarity, alt = "greater", var.equal = TRUE)

##
## Two Sample t-test
##
## data: firstVowel$Familiarity and firstConsonant$Familiarity
## t = 2.0207, df = 4566, p-value = 0.02168
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
## 0.03959434 Inf
## sample estimates:
## mean of x mean of y
```

```
## 4.002951 3.789892
```

Utjecaj broja samoglasnika na brzinu rješavanja zadataka

int procjena za broj slova

Veza između frekvencije pojavljivanja i duljine riječi

jel rijeci koje se pisu slicno imaju isti broj slova

ako se rijec pise slicno ko neka druga, jesu li obe najcesce u sluzbi glagola/imenice

ovisnost vremena i broja slova

ovisnost vremena i broja pojavljivanja rijeci

pojavljuju li se vise kratke ili duge rijeci

Prepoznatljivost glagola u odnosu na imenice

Ukoliko nađemo prepoznatljivosti glagola te prepoznatljivosti imenica, s obzirom da imamo veliku količinu podataka, možemo provesti *z-test* nad prepoznatljivostima te uz alternativnu hipotezu da je prepoznatljivost glagola veća od prepoznatljivosti imenica ne zaključujemo da su glagoli prepoznatljiviji, pa provodimo drugi *z-test* da su imenice prepoznatljivije od glagola. Na temelju drugog testa zaključujemo da su imenice prepoznatljivije od glagola.

```
require(BSDA, quietly = TRUE)
```

```
##
```

```
## Attaching package: 'BSDA'
```

```
## The following object is masked from 'package:datasets':
```

```
##
```

```
## Orange
```

```
verb_familiarity = english[english$WordCategory == "V", ]$Familiarity  
noun_familiarity = english[english$WordCategory == "N", ]$Familiarity
```

```
verb_sd = sd(english[english$WordCategory == "V", ]$Familiarity)
```

```
noun_sd = sd(english[english$WordCategory == "N", ]$Familiarity)
```

```
z.test(verb_familiarity, y = noun_familiarity, alternative = "greater", sigma.x = verb_sd, sigma.y = noun_sd)
```

```
##
```

```
## Two-sample z-Test
```

```
##
```

```
## data: verb_familiarity and noun_familiarity
```

```
## z = 8.5172, p-value < 2.2e-16
```

```
## alternative hypothesis: true difference in means is greater than 0
```

```
## 95 percent confidence interval:
```

```
## 0.24113 NA
```

```
## sample estimates:
```

```
## mean of x mean of y
```

```
## 3.985565 3.686722
z.test(noun_familiarity, y = verb_familiarity, alternative = "greater", sigma.x = noun_sd, sigma.y = ve

##
## Two-sample z-Test
##
## data: noun_familiarity and verb_familiarity
## z = -8.5172, p-value = 1
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
## -0.3565562 NA
## sample estimates:
## mean of x mean of y
## 3.686722 3.985565

cor(english$WrittenFrequency, english$LengthInLetters)

## [1] -0.06663196

#cor(english$WrittenSpokenFrequencyRatio, english$RTlexdec)

moreSpoken = english[english$WrittenSpokenFrequencyRatio > 0, ]
#je li riječi koje se više govore, dakle writespokefrequncyratio < 0 imaju i veći rezultat na RTnamingu

#numericYoung = young[sapply(english, is.numeric)]
#numericOld = old[sapply(english, is.numeric)]

#cor(numericOld)
#cor(numericYoung)

#diff = cor(numericOld) - cor(numericYoung)
#diff
```

## Logistička regresija

Naučite model logističke regresije da predviđa varijablu WordCategory na temelju prediktorskih varijabli RTlexdec i RTnamingu.

```
model1 = glm(WordCategory ~ RTlexdec + RTnaming, data = english, family = binomial())
summary(model1)
```

```
##
## Call:
## glm(formula = WordCategory ~ RTlexdec + RTnaming, family = binomial(),
## data = english)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0552  -0.9643  -0.9219   1.3964   1.6145
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.5912     1.3046   1.986  0.04701 *
## RTlexdec     -0.8996     0.3046  -2.953  0.00314 **
```



```
## RTnaming      0.4339      0.2661      1.630  0.10301
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 5991.7  on 4567  degrees of freedom
## Residual deviance: 5982.0  on 4565  degrees of freedom
## AIC: 5988
##
## Number of Fisher Scoring iterations: 4
```

Pomoću anove ćemo testirati postoji li razlika između prije naučenih modela na razini značajnosti 95% tako što ćemo testirati nultu hipotezu da nema razlike. U prvom slučaju nam p-vrijednost ispadne veća od 0.05, pa ne možemo odbaciti nultu hipotezu. U drugom slučaju nam p-vrijednost ispadne manja od 0.05, pa nultu hipotezu odbacujemo.

```
model2 = glm(WordCategory ~ RTlexdec, data = english, family = binomial())
summary(model2)
```

```
##
## Call:
## glm(formula = WordCategory ~ RTlexdec, family = binomial(), data = english)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0218  -0.9596  -0.9260   1.3977   1.5591
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.8646     1.2912   2.219  0.02651 *
## RTlexdec      -0.5225     0.1972  -2.650  0.00805 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 5991.7  on 4567  degrees of freedom
## Residual deviance: 5984.7  on 4566  degrees of freedom
## AIC: 5988.7
##
## Number of Fisher Scoring iterations: 4
```

```
anova(model1, model2, test= "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: WordCategory ~ RTlexdec + RTnaming
## Model 2: WordCategory ~ RTlexdec
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         4565      5982.0
## 2         4566      5984.7 -1    -2.6643   0.1026
```

```
model3 = glm(WordCategory ~ RTnaming, data = english, family = binomial())
summary(model3)
```

```
##
```

```
## Call:
## glm(formula = WordCategory ~ RTnaming, family = binomial(), data = english)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -0.9709  -0.9604  -0.9405   1.4094   1.4473
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.4868     1.0899   0.447   0.655
## RTnaming     -0.1651     0.1724  -0.958   0.338
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 5991.7  on 4567  degrees of freedom
## Residual deviance: 5990.8  on 4566  degrees of freedom
## AIC: 5994.8
##
## Number of Fisher Scoring iterations: 4
```

```
anova(model1, model3, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: WordCategory ~ RTlexdec + RTnaming
## Model 2: WordCategory ~ RTnaming
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      4565      5982.0
## 2      4566      5990.8 -1   -8.8029 0.003007 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Predviđanja i šanse (malo radi fore):

```
test1 = english[5,]
p1 = predict(model1, test1, type = "response")
p2 = predict(model2, test1, type = "response")
p3 = predict(model3, test1, type = "response")

test2 = english[18,]
p4 = predict(model1, test2, type = "response")
p5 = predict(model2, test2, type = "response")
p6 = predict(model3, test2, type = "response")

odds1 = p1/(1-p1)
odds2 = p4/(1-p4)
```