Computational Linguistics Project

Abhilasha Kumar

Reading the File

```
rpp = read.csv("rpp_data.csv", header = TRUE, sep = ",")
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
rpp = rpp %>% filter(TextNumber != "" & Abstract != "No Abstract" & Replicate..R. != "")
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
rpp = rpp[,c(138, 139,76,24, 30, 36,37,72)]
colnames(rpp) = c("TextNumber", "Abstract", "Replicated", "Citation Count",
                  "Discipline", "SurprisingResult", "ExcitingResult",
                  "Direction of Replication")
rpp$Replicated = ifelse(rpp$Replicated == "yes", "Yes", "No")
```

Plotting Studies

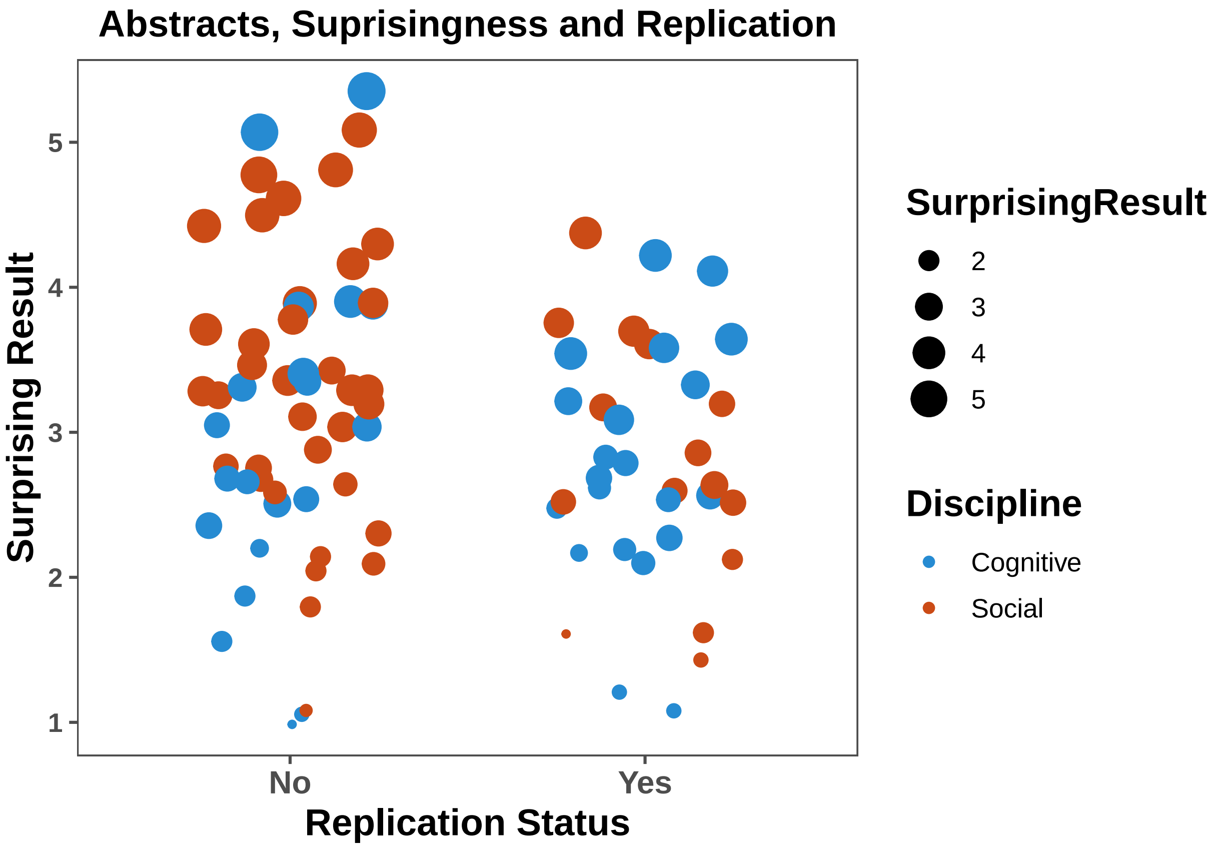
```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
library(ggthemes)
```

```
## Warning: package 'ggthemes' was built under R version 3.4.4
```

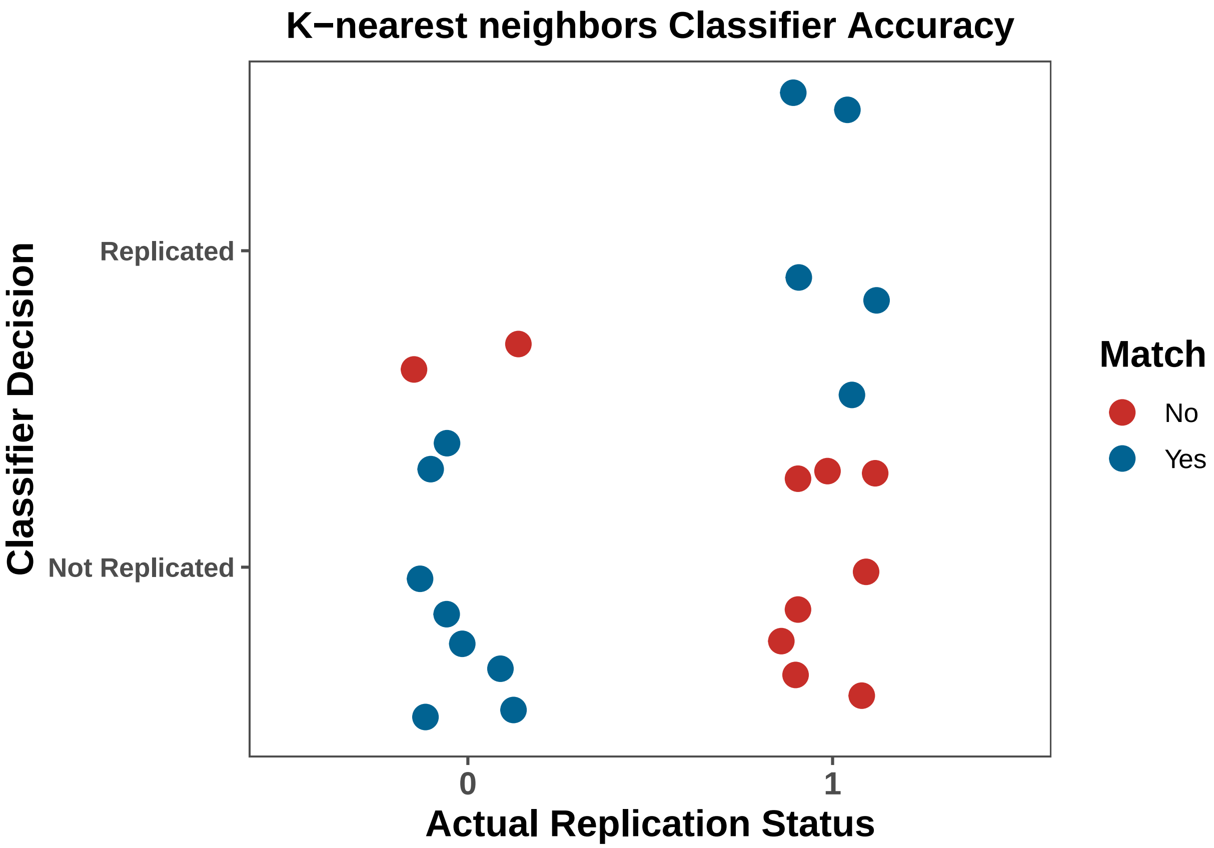
```
rpp_discipline = group_by(rpp, Discipline, Replicated) %>%
  count(Studies = n())
rpp$SurprisingResult = as.numeric(as.character(rpp$SurprisingResult))
rpp %>%
  ggplot(aes(x =Replicated , y = SurprisingResult)) +
  geom_jitter(aes(color = Discipline, size = SurprisingResult), width = 0.25, height = 0.5)+
 # geom_bar(stat = "identity", position = "dodge", width = 0.5)+
    theme_few()+
  scale_color_solarized()+
  xlab("Replication Status") + ylab("Surprising Result") +
    ggtitle("Abstracts, Suprisingness and Replication") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Classifier Decisions

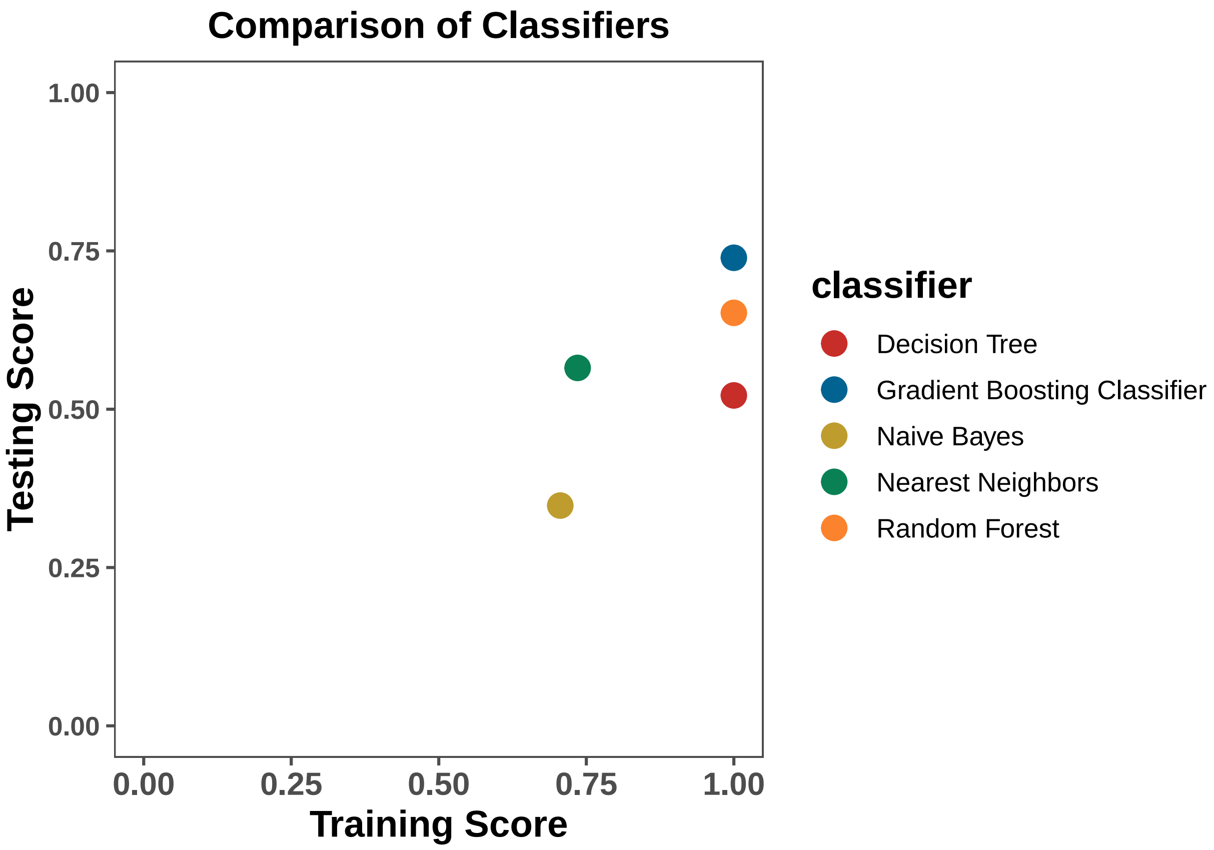
Nearest Neighbors

```
c3 = read.csv("classify_knn.csv", header = TRUE, sep = ",")
c3 %>%
  ggplot(aes(x =factor(Actual), y = Decision)) +
    geom_jitter( width = 0.15, height = 0.5, aes(color = Match), size = 4)+
    theme_few()+
  scale_color_wsj()+
  xlab("Actual Replication Status") + ylab("Classifier Decision") +
    ggtitle("K-nearest neighbors Classifier Accuracy") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
          plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Classifier Performance

```
clf_compare = read.csv("classifier_compare.csv", header = TRUE, sep = ",")

clf_compare = clf_compare %>% arrange(desc(test_score))

clf_compare = clf_compare %>% filter(!classifier %in% c("Logistic Regression",
                                                        "Linear SVM",
                                                        "Neural Net"))


clf_compare %>%
  ggplot(aes(x =train_score, y = test_score)) +
    geom_point(aes(color = classifier), size = 4)+
  theme_few()+
  xlim(0,1)+
  ylim(0,1)+
  scale_color_wsj()+
  xlab("Training Score") + ylab("Testing Score") +
    ggtitle("Comparison of Classifiers") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
   plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



Lexical Diversity and Length

```
ld = read.csv("lexical_diversity.csv", header = TRUE, sep = ",")

ld$R = ifelse(ld$Replicated == "yes", 1,0)
ld$ld_c = scale(ld$Lexical.Diversity, center = TRUE, scale = FALSE)
ld_lm  = glm (data = ld, R ~ Ldnew + Length,
              family = "binomial")
summary(ld_lm)
```

```
##
## Call:
## glm(formula = R ~ Ldnew + Length, family = "binomial", data = ld)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3629  -0.9831  -0.7739   1.2071   1.8834
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.820672   3.104206  -0.587   0.5575
## Ldnew       -1.640713   3.339191  -0.491   0.6232
## Length       0.015536   0.007623   2.038   0.0415 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 122.16  on 90  degrees of freedom
## Residual deviance: 115.76  on 88  degrees of freedom
## AIC: 121.76
## 
## Number of Fisher Scoring iterations: 4
```

```
ld_lm2  = glm (data = ld, R ~  Length,
              family = "binomial")
summary(ld_lm2)
```

```
## 
## Call:
## glm(formula = R ~ Length, family = "binomial", data = ld)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.3662  -0.9786  -0.7906   1.2452   1.8789  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.23329    1.21859  -2.653  0.00797 **
## Length       0.01690    0.00716   2.360  0.01828 * 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 122.16  on 90  degrees of freedom
## Residual deviance: 116.00  on 89  degrees of freedom
## AIC: 120
## 
## Number of Fisher Scoring iterations: 4
```

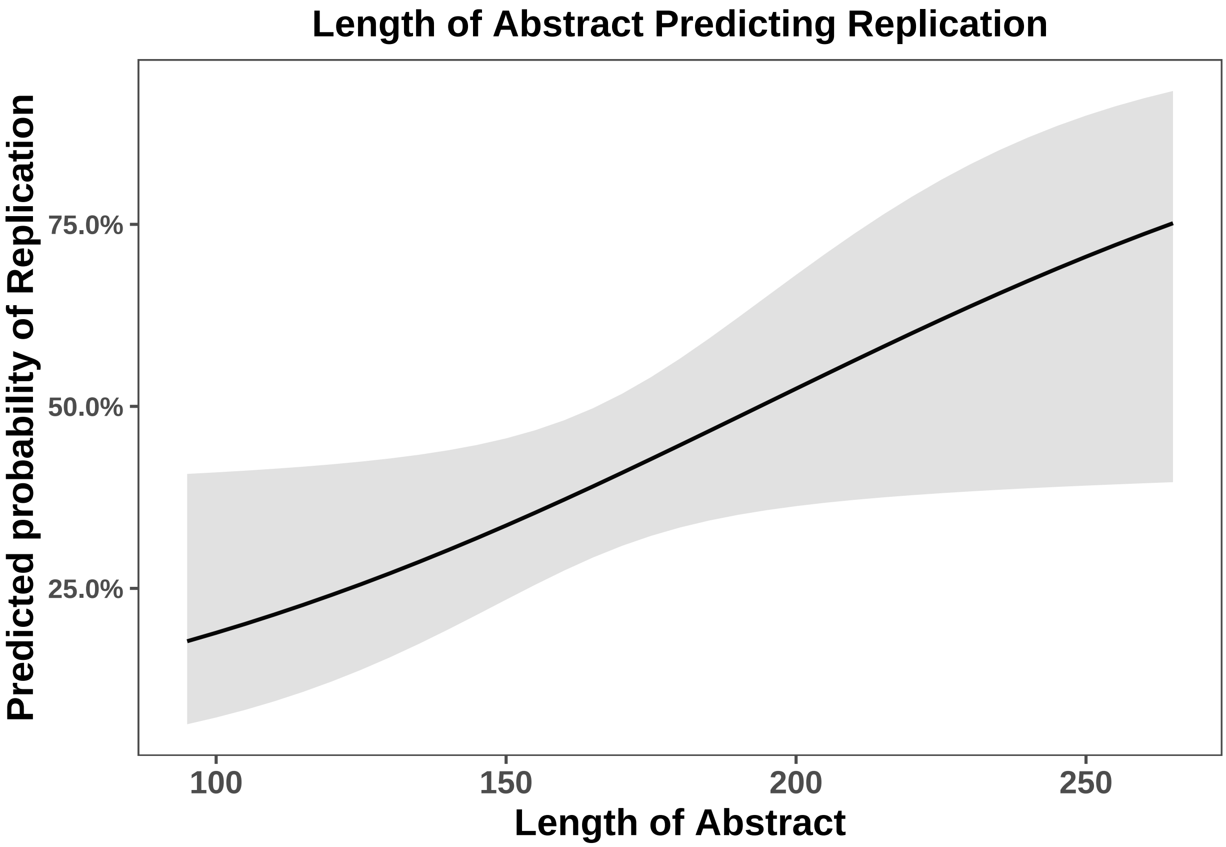
```
anova(ld_lm, ld_lm2)
```

```
## Analysis of Deviance Table
## 
## Model 1: R ~ Ldnew + Length
## Model 2: R ~ Length
##   Resid. Df Resid. Dev Df Deviance
## 1        88     115.76            
## 2        89     116.00 -1 -0.24222
```

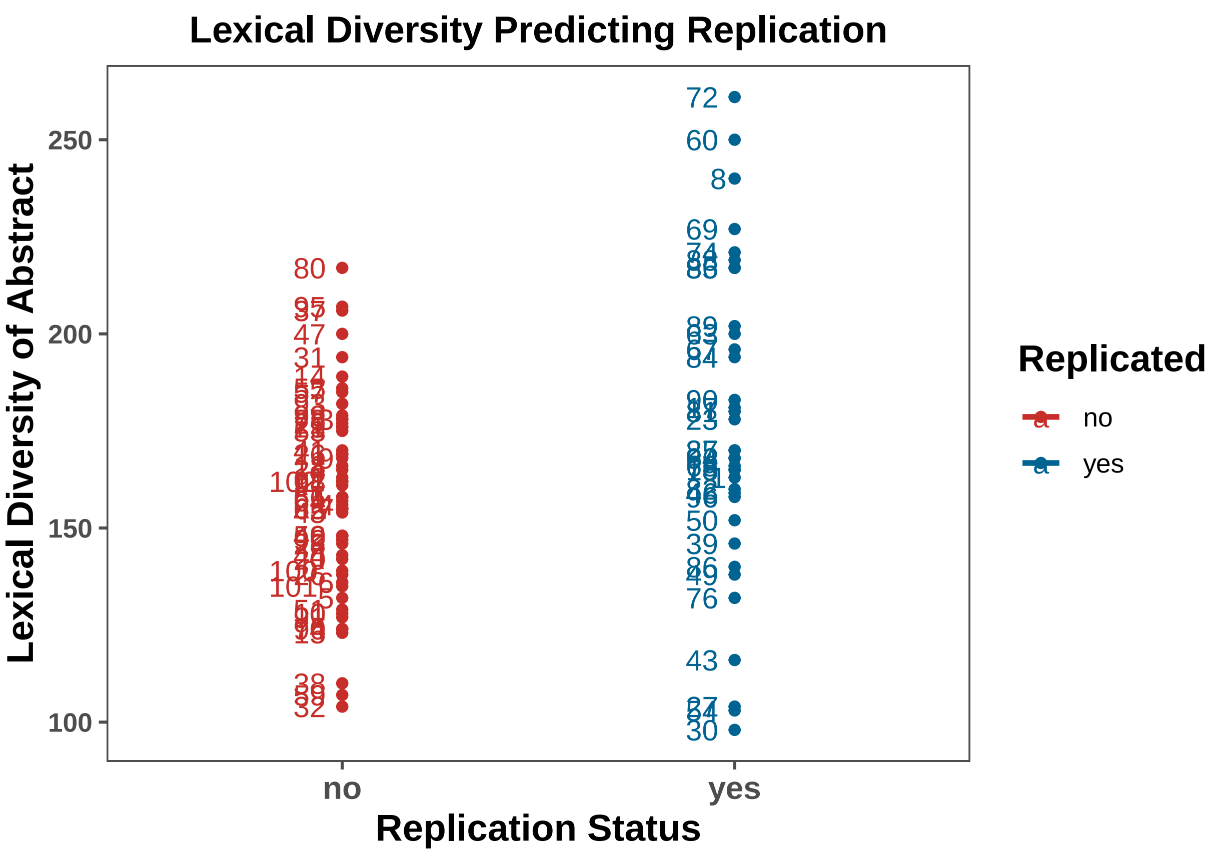
```
## need to plot length figure

x = sjPlot::plot_model(ld_lm, type = "pred", terms = "Length")

x + theme_few()+
  scale_color_wsj()+
  xlab("Length of Abstract") + ylab("Predicted probability of Replication") +
    ggtitle("Length of Abstract Predicting Replication") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
```

```r
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
      plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```

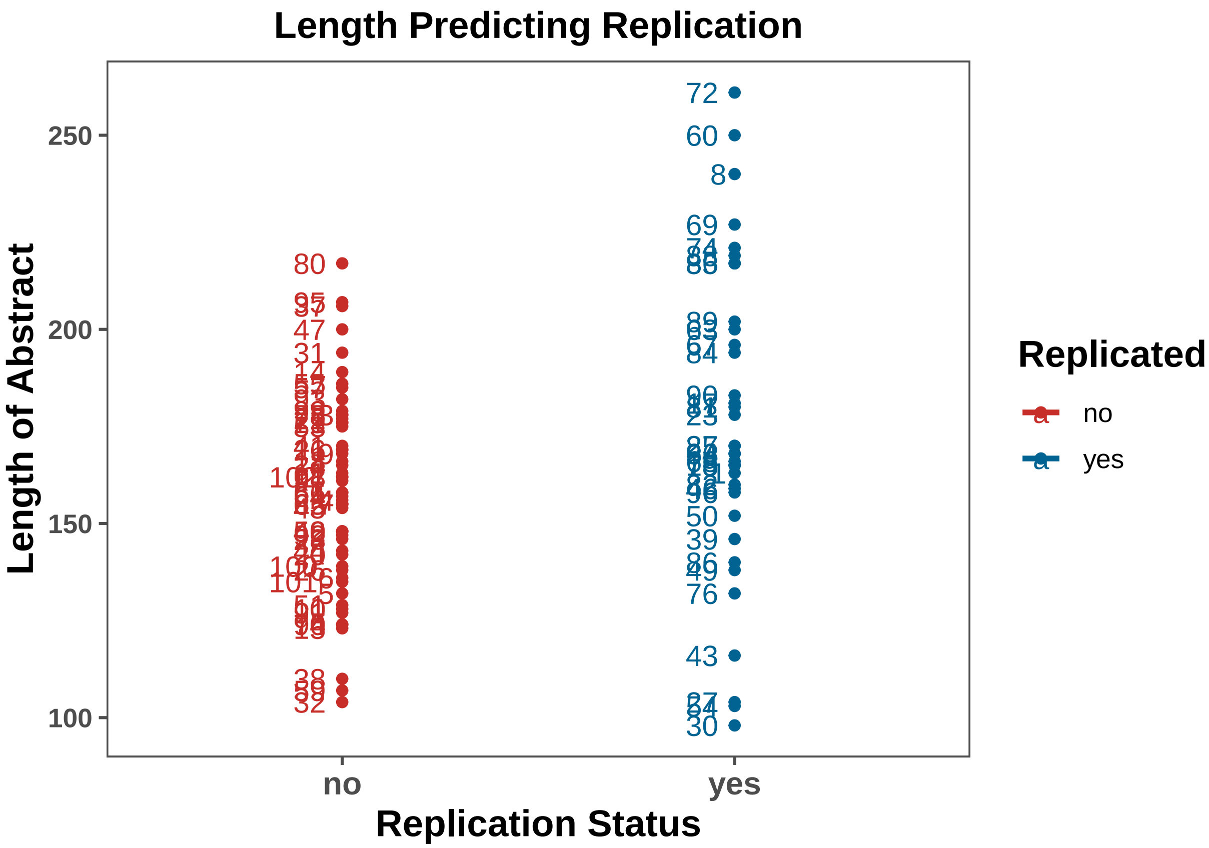
```
## Scale for 'colour' is already present. Adding another scale for
## 'colour', which will replace the existing scale.
```



```r
ld$Lexical.Diversity = round(ld$Lexical.Diversity, digits = 2)
ld %>%
  ggplot(aes(x =Replicated, y = Length, color = Replicated)) +
    geom_point()+
    geom_text(aes(label=TextNumber), hjust = 1.5, vjust = .5)+
  geom_smooth(method = "glm", se = FALSE)+
  theme_few()+
  scale_color_wsj()+
  xlab("Replication Status") + ylab("Lexical Diversity of Abstract") +
    ggtitle("Lexical Diversity Predicting Replication") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
      plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



```
ld %>%
  ggplot(aes(x =Replicated, y = Length, color = Replicated)) +
    geom_point()+
    geom_text(aes(label=TextNumber), hjust = 1.5, vjust = .5)+
  geom_smooth(method = "glm", se = FALSE)+
  theme_few()+
  scale_color_wsj()+
  xlab("Replication Status") + ylab("Length of Abstract") +
    ggtitle("Length Predicting Replication") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
          legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.2)),
      plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```



POS Tagging

```r
pos_data = read.csv("pos_python.csv", header = TRUE, sep = ",")
library(dplyr)
## This data is in wide format: need to convert to long format

pos_long = tidyr::gather(pos_data, PartOfSpeech, Count,
                         Adjective, Noun, Verb, Other, factor_key=TRUE)
pos_long = pos_long %>% arrange(TextNumber)

pos_long$Percent = pos_long$Count/pos_long$Length
pos_long$Percent = round(pos_long$Percent, digits = 2)

pos_long$R = ifelse(pos_long$Replicated == "yes", 1,0)

contrasts(pos_long$PartOfSpeech) = contr.treatment(4, base = 1)
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 3.4.4
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 3.4.4
```

```r
cl3 <- glmerControl(optimizer="optimx",
                    optCtrl=list(method="nlminb",maxiter=10000))
```

```r
library(optimx)
pos_lm  = glmer (data = pos_long, R ~ PartOfSpeech*Percent  +
                   (1|TextNumber),
              family = "binomial", control = cl3)
summary(pos_lm)
```

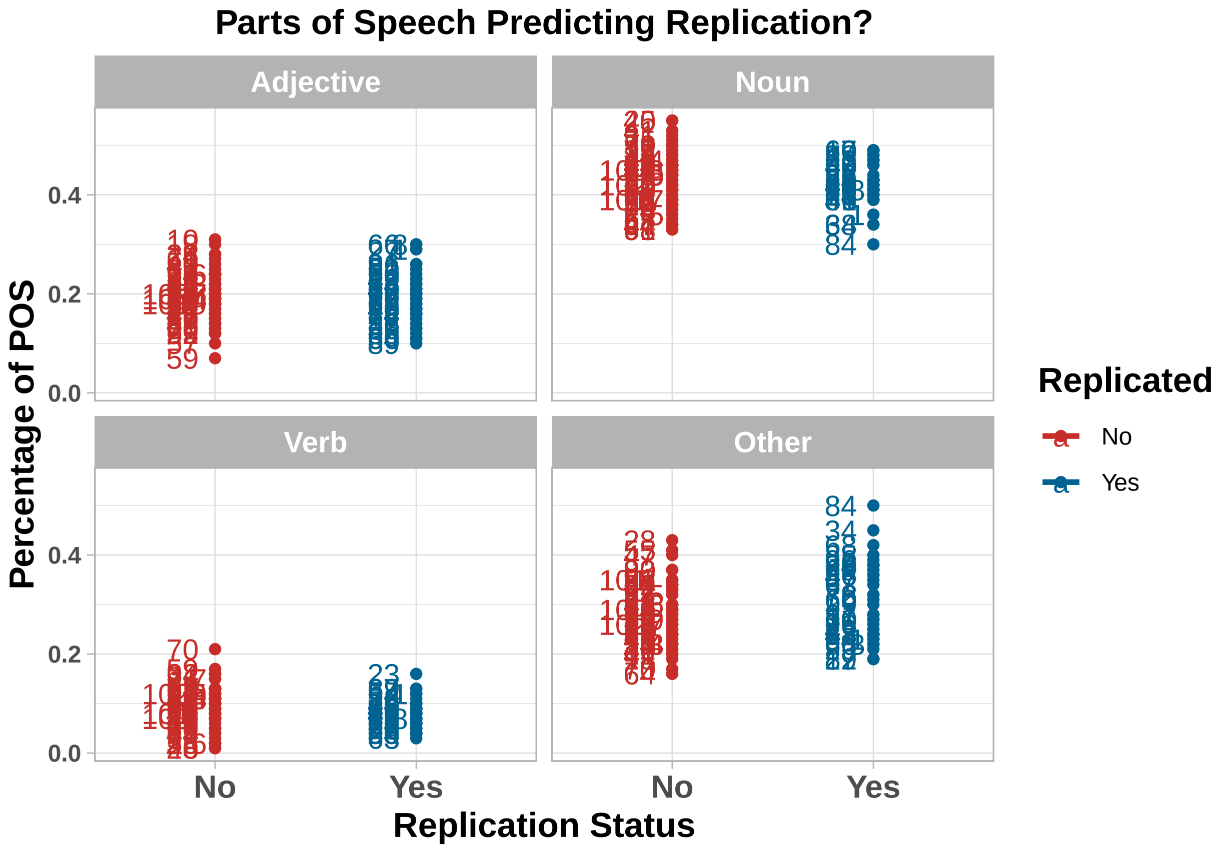
```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: R ~ PartOfSpeech * Percent + (1 | TextNumber)
##    Data: pos_long
## Control: cl3
##
##      AIC      BIC   logLik deviance df.resid
##    132.7    167.8    -57.3    114.7      355
##
## Scaled residuals:
##       Min        1Q    Median        3Q       Max
## -0.002221 -0.001736 -0.001580  0.036242  0.042923
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  TextNumber (Intercept) 3457     58.8
## Number of obs: 364, groups:  TextNumber, 91
##
## Fixed effects:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -12.5772     6.1014  -2.061   0.0393 *
## PartOfSpeech2           0.9861    14.0685   0.070   0.9441
## PartOfSpeech3           0.4694     6.8714   0.068   0.9455
## PartOfSpeech4          -1.1406     9.6022  -0.119   0.9054
## Percent                -0.4872    29.6337  -0.016   0.9869
## PartOfSpeech2:Percent  -2.0591    42.2061  -0.049   0.9611
## PartOfSpeech3:Percent  -6.2600    49.8876  -0.125   0.9001
## PartOfSpeech4:Percent   4.0429    39.6274   0.102   0.9187
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) PrtOS2 PrtOS3 PrtOS4 Percnt POS2:P POS3:P
## PartOfSpch2 -0.409
## PartOfSpch3 -0.826  0.373
## PartOfSpch4 -0.695  0.246  0.580
## Percent     -0.945  0.412  0.828  0.697
## PrtOfSpc2:P  0.664 -0.929 -0.593 -0.447 -0.705
## PrtOfSpc3:P  0.544 -0.260 -0.872 -0.356 -0.579  0.425
## PrtOfSpc4:P  0.783 -0.296 -0.663 -0.963 -0.826  0.547  0.440
```

```r
car::Anova(pos_lm)
```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
##
## Response: R
##                         Chisq Df Pr(>Chisq)
## PartOfSpeech         0.0001  3     1.0000
## Percent              0.0000  1     0.9964
## PartOfSpeech:Percent 0.0530  3     0.9968
```

```
pos_long$Replicated = ifelse(pos_long$Replicated == "yes", "Yes", "No")
```

```
library(ggplot2)
library(ggthemes)
pos_long %>%
  ggplot(aes(x =Replicated, y = Percent, color = Replicated)) +
    geom_point()+
   geom_text(aes(label=TextNumber), hjust = 1.5, vjust = .5)+
  geom_smooth(method = "glm", se = FALSE)+
    theme_light()+
  scale_color_wsj()+
  facet_wrap(~PartOfSpeech)+
  xlab("Replication Status") + ylab("Percentage of POS") +
    ggtitle("Parts of Speech Predicting Replication?") +
  theme(axis.text = element_text( face = "bold", size = rel(0.8)),
          axis.title = element_text(face = "bold", size = rel(1.2)),
           legend.title = element_text(face = "bold", size = rel(1.2)),
        axis.text.x = element_text(face = "bold", size = rel(1.4)),
          strip.text.x = element_text(face = "bold", size = rel(1.2)),
    plot.title = element_text(face = "bold", size = rel(1.2), hjust = .5))
```


Parts of Speech Predicting Replication?
Adjective
Noun
Verb
Other
Percentage of POS
No
Yes
No
Yes
Replication Status
Replicated
No
Yes