PH125.9x Data Science: WordBank project

Rodrigo Dal Ben de Souza

24/02/2022

Contents

1 Introduction

Mastering a language is an daunting task–as any second language learner will quick attest. However infants seem to tackle this problem seamlessly and with great success. They begin their lives with very little linguistic knowledge and by their second anniversary they have already mastered a great deal of their native(s) language(s) (J. Werker and Curtin (2005)). Vocabulary growth is an important source of information about processes and mechanisms of language development. For instance, using validated instruments such as The MacArthur-Bates Communicative Development Inventories (hereafter referred as CDI; Fenson et al. (2007)), researchers have discovered a great deal about comprehension and production of language, grammatical and lexical repertoires, and lexical networks (for a review see Michael C. Frank et al. (2021)). The CDI is a set of inventories that assess vocabulary growth based on parent-report on what their children can understand (receptive vocabulary) and speak (productive vocabulary).

Recently, a joint effort between researchers from across the globe lead to the creation of the WordBank: an evolving repository of CDI data from more than 20 languages and 40,000 children (Michael C. Frank et al. (2017)). The repository is designed to facilitate reuse and reanalyses of CDI data, allowing anyone interested in language development to explore these rich vocabulary growth data (Michael C. Frank et al. (2021)). All data and several analyses are openly available at the WordBank website and as a R package: `wordbankr`.

In the present project, we will use machine learning algorithms to generate insights about relationships between demographic/linguistic variables (our predictors) and vocabulary growth, as measured by **productive vocabulary** on the CDI (our outcome measure). All our analyses are exploratory in nature and we don't have any hypotheses or predictions on what we might find. We will start with curating our dataset, moving to descriptive analyses and visualizations, and finally to three machine learning algorithms to the data: regression trees, random forests, and linear regression.

2 Method

2.1 Libraries

We will use seven packages on this project: 1) `wordbankr` is used to access the data, 2) `tidyverse` is used for data manipulation and visualizations, 3) `here` for a quick way to use relative paths, 4) `caret` is used for creating training and test sets and to assess variable importance in random forests, 5) `rpart` is used to run regression trees, 6) `rpart.plot` for plotting regression trees, and 7) `randomForest` is used to fit random forests. We will also color-blind friendly palette (`cb_pal`) for exploratory plots.

```
# general info from sessionInfo()
# R version 4.1.2
# Platform: aarch64-apple-darwin20 (64-bit)
# Running under: macOS Monterey 12.1

# install packages if necessary -- code based on:
# https://statsandr.com/blog/an-efficient-way-to-install-and-load-r-packages/

# packages w/ version
pkgs <- c("wordbankr", # v0.3.1
          "tidyverse", # v1.3.1
          "here", # v1.0.1
          "caret", # v6.0-90
          "rpart", # v4.1-15
          "rpart.plot", # v3.1.0
          "randomForest" # v4.6-14
          )
```

```r
# if necessary install
installed_pkgs <- pkgs %in% rownames(installed.packages())
if(any(installed_pkgs == F)){install.packages(pkgs[!installed_packages])}

# load packages
invisible(lapply(pkgs, library, character.only = T))

# clean
rm(installed_pkgs, pkgs)

# color-blind friendly palette
color_blind_colors <- c("#E69F00", "#56B4E9", "#009E73", "#F0E442", "#0072B2", "#D55E00", "#CC79A7")
```

2.2 Data

There are several datasets available on WordBank, for more a complete description on available datasets see Michael C. Frank et al. (2021). Here we will use the **administration** dataset, which contains several demographic and linguistic information (e.g., age in months, sex, caregivers' level of educational, language), as well as vocabulary data (comprehension and production).

```r
# load raw data
data_raw <- wordbankr::get_administration_data()
```

The raw administration dataset contains 82055 observations of 15 variables: data_id, age, comprehension, production, language, form, birth_order, ethnicity, sex, zygosity, norming, mom_ed, longitudinal, source_name, license. Given our goal of exploring relationships between demographic and lingusitic variables with vocabulary growth, we will clean variables that are not relevant to our analyses.

```r
# data overview
head(data_raw)

# drop variables
data_clean01 <- data_raw %>%
  select(-data_id, -zygosity, -norming, -longitudinal, -source_name, -license)

# convert character to factor
str(data_clean01)

data_clean01 <- data_clean01 %>%
  mutate(language = as_factor(language),
         form = as_factor(form))

str(data_clean01)
```

Having comparable data across languages is essential for meaningful analyses. It is worth remembering that the CDI is composed of a family of measuring instruments. For instance, the data in our dataset comes from 11 different forms. Now we will check the distribution of data from these forms across the 29 languages.

```r
# calculate frequency of languages for each form
freq_form <- data_clean01 %>%
  group_by(form) %>%
```

```
  summarise(n_language = n_distinct(language),
            n_obs = n()) %>%
  arrange(desc(n_obs))

freq_form
```

```
## # A tibble: 11 x 3
##    form        n_language n_obs
##    <fct>            <int> <int>
##  1 WS                  26 41117
##  2 WG                  21 16868
##  3 TEDS Twos            1 11129
##  4 TEDS Threes          1 10790
##  5 Oxford CDI           1  1210
##  6 TC                   1   652
##  7 IC                   1   230
##  8 FormBOne             1    19
##  9 FormBTwo             1    19
## 10 FormC                1    15
## 11 FormA                1     6
```

Most of our data (57985 observations) comes from the **Words and Sentences** and **Words and Gestures** forms, which were administered across several languages (26, 21, respectively). In contrast, **TEDS Twos** and **TEDS Threes** were also administered thousands of times (21919 observations), but only in one language– as the remaining forms. Thus, we will focus on data from the **Words and Sentences** and **Words and Gestures** forms for our analyses.

```
# filter forms
data_clean02 <- data_clean01 %>%
  filter(form %in% c("WS", "WG")) %>%
  droplevels()
```

Our filtered dataset has 57985 observations from 27 languages. Now let's glance our dataset, especially looking for missing values.

```
# check NAs
summary(data_clean02)
```

```
##       age         comprehension   production                     language
##  Min.   : 7.00   Min.   :  0   Min.   :  0.0   Norwegian            :12225
##  1st Qu.:16.00   1st Qu.: 62   1st Qu.: 18.0   English (American)   : 7955
##  Median :21.00   Median :188   Median :120.0   Danish               : 6112
##  Mean   :21.28   Mean   :252   Mean   :219.9   Portuguese (European): 4326
##  3rd Qu.:27.00   3rd Qu.:423   3rd Qu.:410.0   Turkish              : 3537
##  Max.   :36.00   Max.   :798   Max.   :798.0   Mandarin (Taiwanese) : 2654
##                                                (Other)              :21176
##  form        birth_order       ethnicity         sex
##  WG:16868   First  :12961   Asian   :  113   Female:27866
##  WS:41117   Second : 9430   Black   :  346   Male  :28339
##             Third  : 3875   Other   :  155   NA's  : 1780
##             Fourth :  863   White   : 3137
##             Fifth  :  206   Hispanic:  192
```

```
##             (Other):  130   NA's    :54042
##             NA's   :30520
##           mom_ed
##  College     :13225
##  Secondary   : 6769
##  Graduate    : 6275
##  Some College: 4601
##  Primary     : 2217
##  (Other)     : 1719
##  NA's        :23179
```

We can quickly see several missing values in birth order, ethnicity, caregivers' education, and sex. Let's calculate the proportion of missing values for each of these predictors.

```
# proportion of NA in: birth order, ethnicity, caregivers' education, sex
na_prop <- data_clean02 %>%
  select(birth_order, ethnicity, mom_ed, sex) %>%
  gather(key = predictor) %>%
  group_by(predictor) %>%
  summarise(prop_missing = round(sum(is.na(value))/n(),2))

# create table with proportion of missing data
na_prop %>% knitr::kable()
```

predictor	prop_missing
birth_order	0.53
ethnicity	0.93
mom_ed	0.40
sex	0.03

Birth order, ethnicity, and caregivers' educational level have a substantive number of missing cases. Any manipulation on these variables have the potential to bias our analyses For instance, if we delete ethnicity missing cases we will loose more than 90% of our data, whereas if we replace missing values with most common values we may end with a very biased estimates for birth order and caregivers' educational level. Thus, we will drop these variables from our dataset.

```
# drop birth order, ethnicity, caregivers' education
data_clean03 <- data_clean02 %>%
  select(-birth_order, -ethnicity, -mom_ed)

# glance dataset
summary(data_clean03)
```

```
##       age         comprehension   production                    language
##  Min.   : 7.00   Min.   :  0   Min.   :  0.0   Norwegian            :12225
##  1st Qu.:16.00   1st Qu.: 62   1st Qu.: 18.0   English (American)   : 7955
##  Median :21.00   Median :188   Median :120.0   Danish               : 6112
##  Mean   :21.28   Mean   :252   Mean   :219.9   Portuguese (European): 4326
##  3rd Qu.:27.00   3rd Qu.:423   3rd Qu.:410.0   Turkish              : 3537
##  Max.   :36.00   Max.   :798   Max.   :798.0   Mandarin (Taiwanese) : 2654
##                                                (Other)              :21176
```

```
##  form           sex
##  WG:16868   Female:27866
##  WS:41117   Male  :28339
##             NA's  : 1780
##
##
##
##
```

We still have to deal with missing values from the **sex** variable, which counts for roughly 3% of the cases. Luckily, previous research have consistently shown that females have larger productive vocabularies than males and that this is true across languages (e.g., Michael C. Frank et al. (2017)). So we will calculate the median productive vocabulary for males and females on each form (WS, WG) and use it to classify missing values.

```
# calculate median vocabulary scores
data_clean03 %>%
  group_by(sex, form) %>%
  summarise(m_prod = median(production))
```

```
## # A tibble: 6 x 3
## # Groups:   sex [3]
##   sex    form  m_prod
##   <fct>  <fct>  <dbl>
## 1 Female WG         8
## 2 Female WS       318
## 3 Male   WG         7
## 4 Male   WS       244
## 5 <NA>   WG         1
## 6 <NA>   WS       165
```

```
# classify missing values
data_clean04 <- data_clean03 %>%
  mutate(sex = case_when((is.na(sex) & form == "WG" & production >= 8) ~ "Female",
                         (is.na(sex) & form == "WG" & production <= 8) ~ "Male",
                         (is.na(sex) & form == "WS" & production >= 318) ~ "Female",
                         (is.na(sex) & form == "WS" & production <= 318) ~ "Male",
                         T ~ as.character(sex)),
         sex = as_factor(sex))

# glance at data and calculate differences
summary(data_clean04)
```

```
##       age         comprehension   production                      language
##  Min.   : 7.00   Min.   :  0   Min.   :  0.0   Norwegian            :12225
##  1st Qu.:16.00   1st Qu.: 62   1st Qu.: 18.0   English (American)   : 7955
##  Median :21.00   Median :188   Median :120.0   Danish               : 6112
##  Mean   :21.28   Mean   :252   Mean   :219.9   Portuguese (European): 4326
##  3rd Qu.:27.00   3rd Qu.:423   3rd Qu.:410.0   Turkish              : 3537
##  Max.   :36.00   Max.   :798   Max.   :798.0   Mandarin (Taiwanese) : 2654
##                                                (Other)              :21176
##  form           sex
##  WG:16868   Female:28381
```

```
##  WS:41117   Male  :29604
##
##
##
##
##
```

Using this classification strategy, 515 NAs were classified as females and 1265 as males. Our final dataset contains 57985 observations and 6 variables, namely: age, comprehension, production, language, form, sex.

Now we will split our data it into training and test sets. The training set will be used to train our models and the test set to evaluate our models' accuracy. Given the relatively small size of our final dataset (57985 observations), we will try to balance the amount of variance in parameter estimation during training with the amount of variance in our performance statistic during test. Thus, we will use a split of 70/30, which is slightly more conservative than other proportions, such as the commonly used Pareto proportion (80/20).

```
# set seed for reproducibility
if_else(getRversion() < 3.5, set.seed(123), set.seed(123, sample.kind = "Rounding"))
```

```
## NULL
```

```
# set outcome
y <- data_clean04$production

# partioning data
# create index: 70% (training) 30% (test)
data_index <- createDataPartition(y, p = 0.7, times = 1, list = F)

# create train and test data
data_train <- data_clean04 %>% slice(data_index)
data_test <- data_clean04 %>% slice(-data_index)

# double check proportions
round(nrow(data_train)/(nrow(data_clean04)), 2)
```

```
## [1] 0.7
```

```
round(nrow(data_test)/(nrow(data_clean04)), 2)
```

```
## [1] 0.3
```

2.3 Descriptives and Visualizations

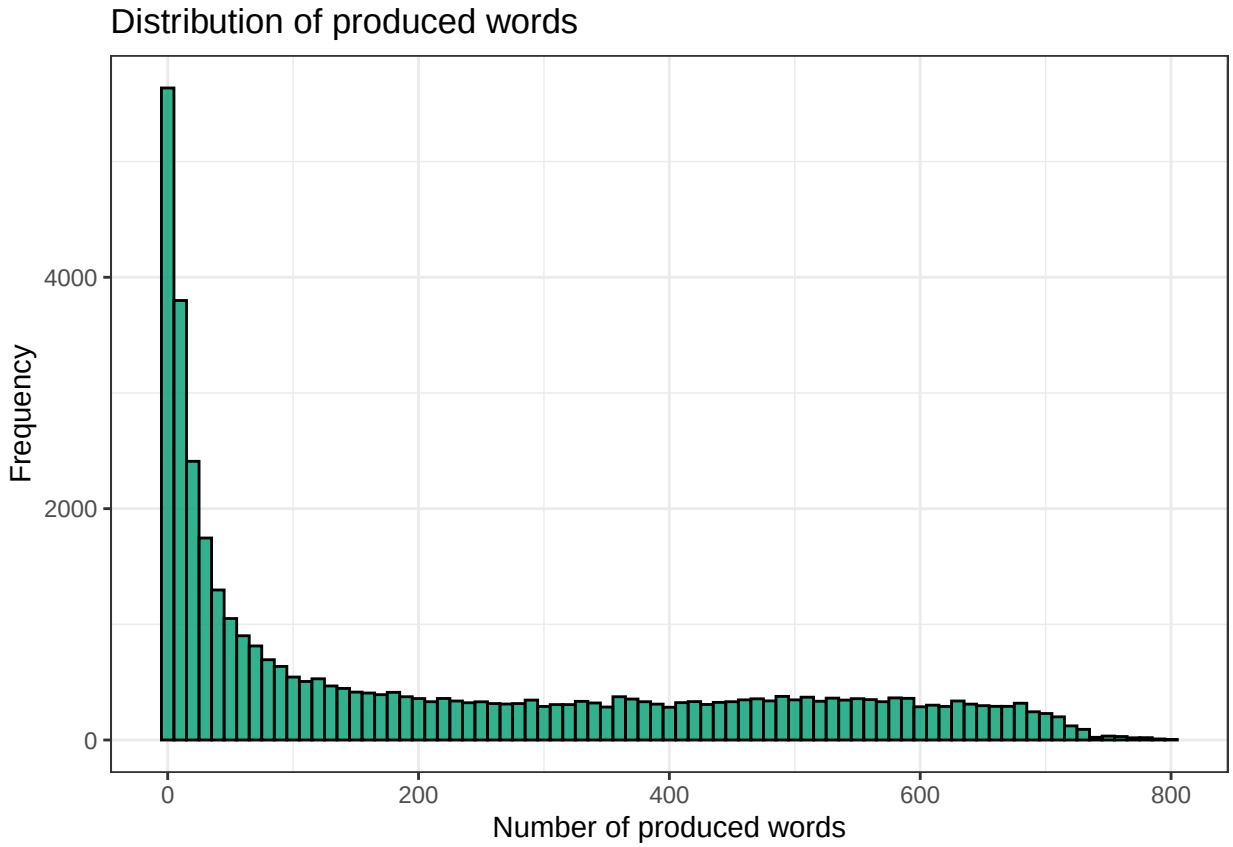
Now we will explore patterns in our data using both descriptive statistics and visualizations. Importantly, we will explore our training set while saving our test set for model evaluation only. We will begin with a brief summary of our data.

```
# summary statistics (training set)
summary(data_train)
```

```
##       age        comprehension     production                     language
##  Min.   : 7.00   Min.   :  0.0   Min.   :  0.0   Norwegian             : 8638
##  1st Qu.:16.00   1st Qu.: 62.0   1st Qu.: 18.0   English (American)    : 5592
##  Median :21.00   Median :188.0   Median :120.0   Danish                : 4231
##  Mean   :21.29   Mean   :252.1   Mean   :219.8   Portuguese (European): 3013
##  3rd Qu.:27.00   3rd Qu.:423.0   3rd Qu.:410.0   Turkish               : 2512
##  Max.   :36.00   Max.   :798.0   Max.   :798.0   Mandarin (Taiwanese) : 1825
##                                                  (Other)              :14780
##  form           sex
##  WG:11850   Female:19854
##  WS:28741   Male  :20737
##
##
##
##
##
```

From this summary, we learn that age vary from 7 to 36 months, with a mean of 21.29 months. We also learn that overall comprehension scores are higher ($Med = 188$) than production scores ($Med = 120$), which is in line with normal vocabulary growth trajectories. Also, data is roughly balanced between male (n = 20737) and females (n = 19854). Now we will plot the distribution of our outcome: productive vocabulary.

```
# distribution of outcome
data_train %>%
  ggplot(aes(x = production)) +
  geom_histogram(binwidth = 10,
                 fill = color_blind_colors[3],
                 color = "black",
                 alpha = 0.8) +
  labs(title = "Distribution of produced words",
       x = "Number of produced words",
       y = "Frequency") +
  theme_bw()
```

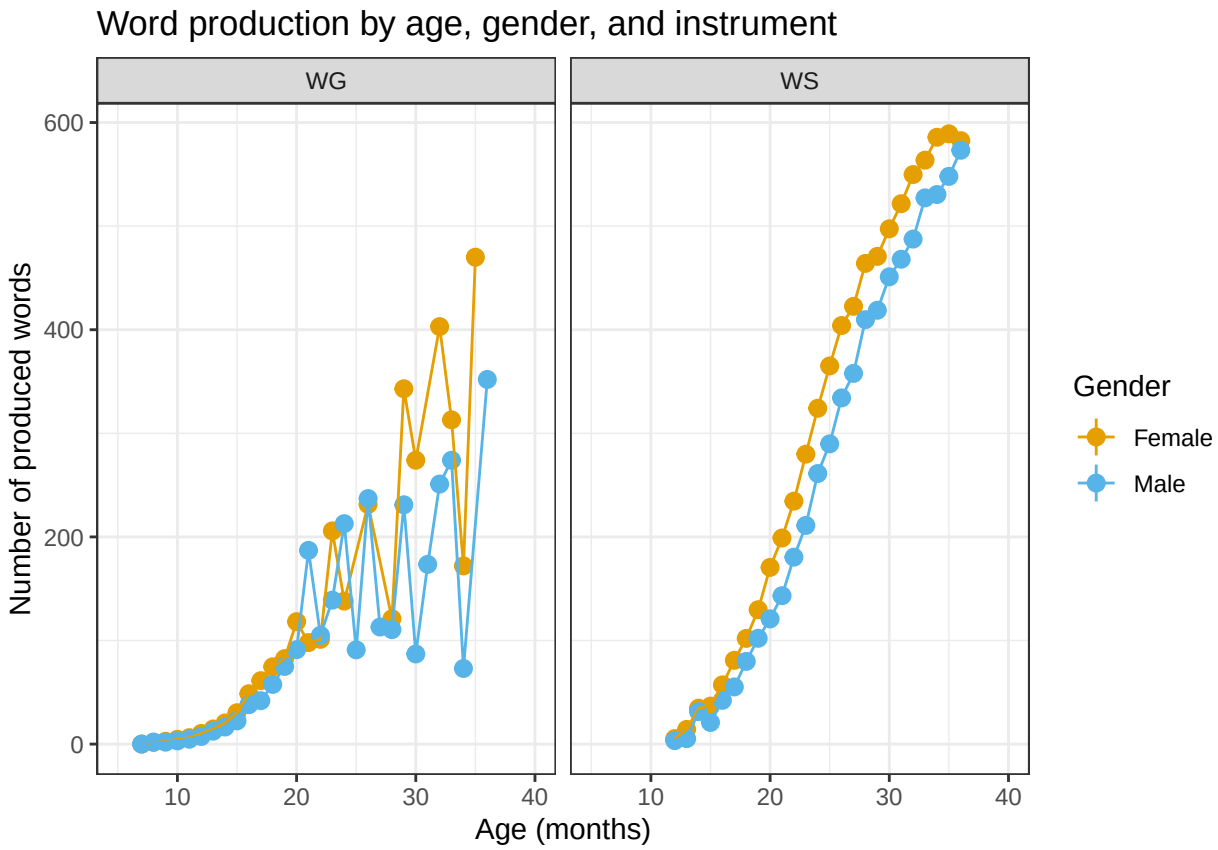


This plot clearly shows that our outcome is positively skewed, drawing a Zipf curve, with most children producing no words and fewer children producing most words. Now we will explore trends in productive vocabulary across age and gender.

```
# productive vocab by age, gender, and instrument (form)
data_train %>%
  ggplot(aes(x = age, y = production, color = sex)) +
  stat_summary(fun = mean) +
  stat_summary(fun = mean, geom = "line") +
  facet_wrap(~ form) +
  scale_color_manual(values= color_blind_colors) +
  xlim(5, 40) +
  labs(title = "Word production by age, gender, and instrument",
       x = "Age (months)",
       y = "Number of produced words",
       color = "Gender") +
  theme_bw()
```

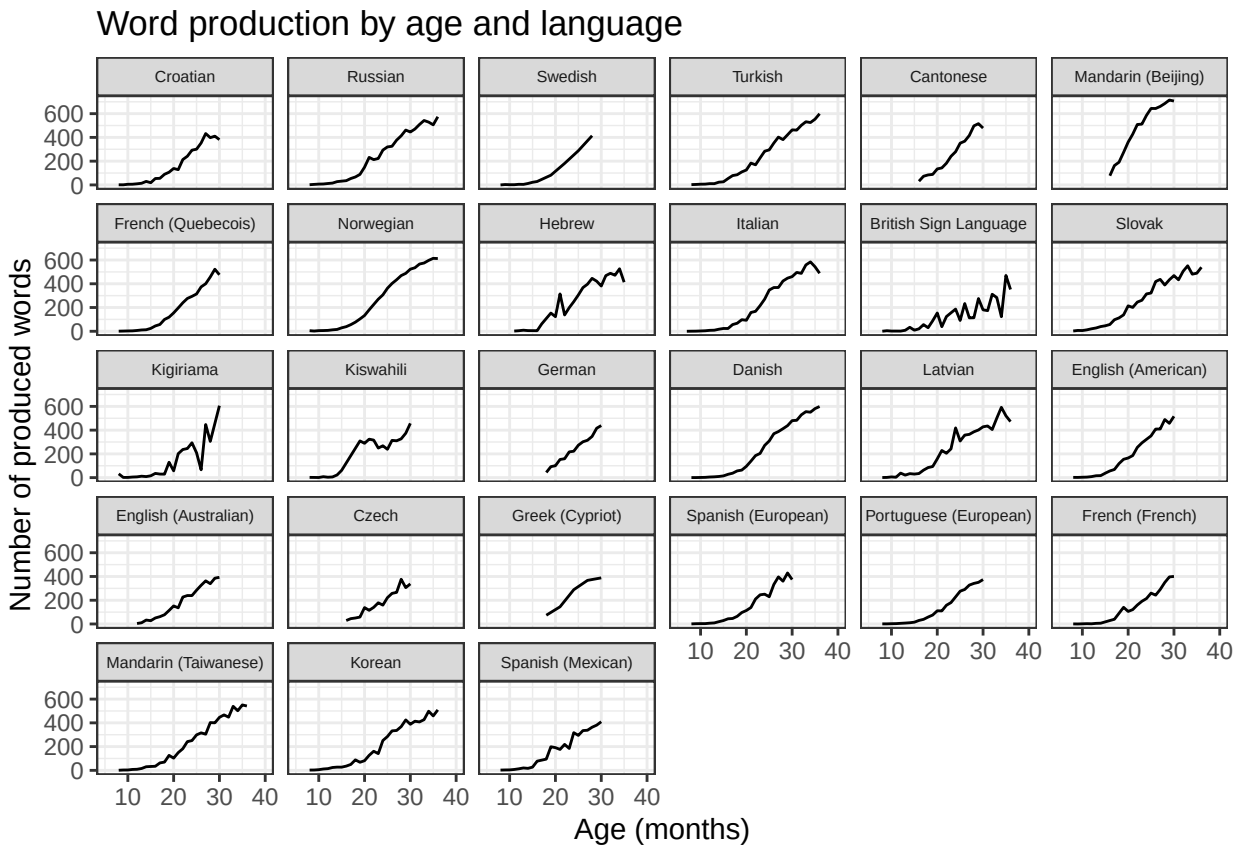
```
## Warning: Removed 55 rows containing missing values (geom_segment).
```

```
## Warning: Removed 50 rows containing missing values (geom_segment).
```



The plots above show that productive vocabulary increases with age for both genders and instruments (forms). In addition, consistent with previous research, overall, females have larger vocabularies than males across both age and instruments (Michael C. Frank et al. (2017)). Now we analyse whether the positive trend between age and productive vocabulary is also observed across languages.

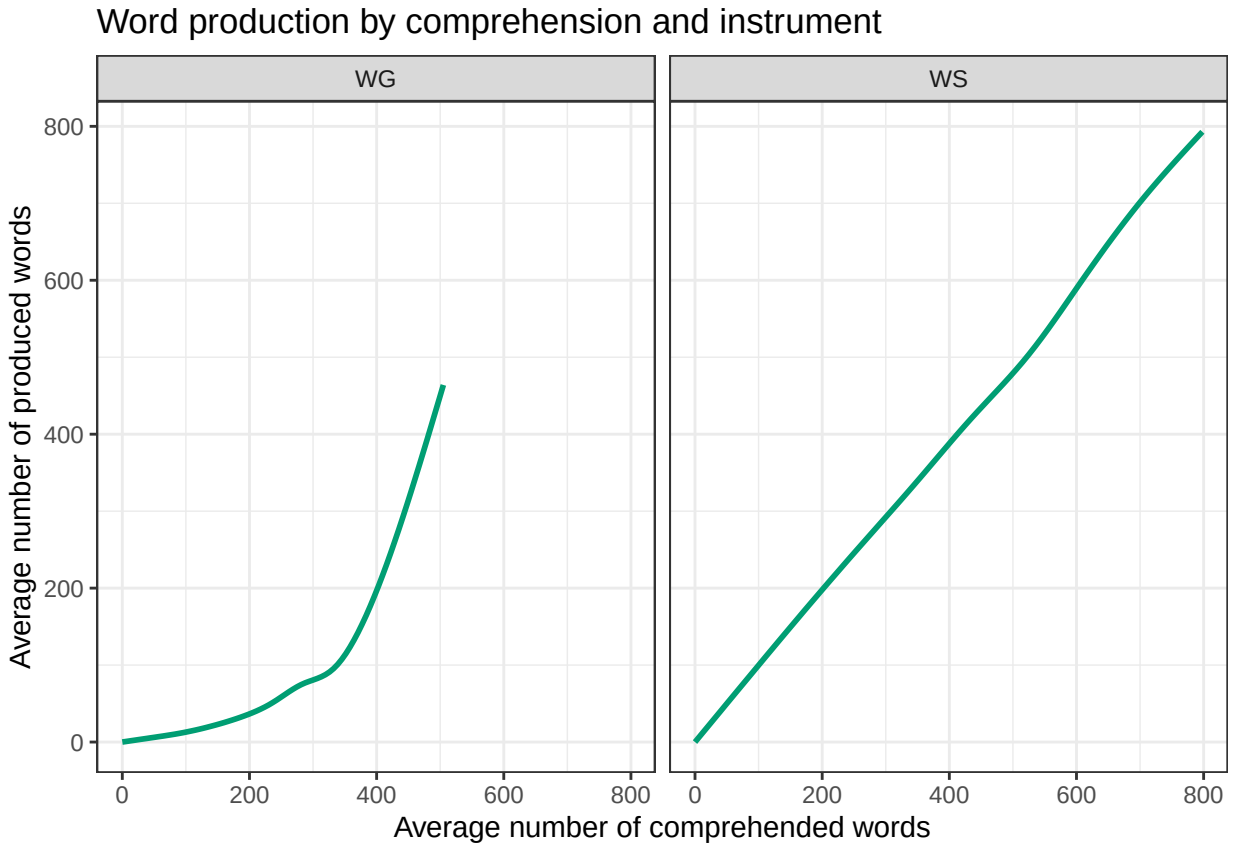
```
# productive vocab by age and language
data_train %>%
  ggplot(aes(x = age, y = production)) +
  stat_summary(fun = mean, geom = "line") +
  facet_wrap(~ language) +
  xlim(5, 40) +
  labs(title = "Word production by age and language",
       x = "Age (months)",
       y = "Number of produced words") +
  theme_bw() +
  theme(strip.text = element_text(size = 6))
```



Indeed, productive vocabulary increases with age across all languages, which suggests that the machine learning analyses we will develop in this project might generalize across languages. Now we will explore trends between receptive vocabulary and productive vocabulary.

```
# productive vocab by comprehension and instrument (form)
data_train %>%
  ggplot(aes(x = comprehension, y = production)) +
  geom_smooth(se = F, color = color_blind_colors[3]) +
  facet_wrap(~ form) +
  labs(title = "Word production by comprehension and instrument",
       x = "Average number of comprehended words",
       y = "Average number of produced words") +
  theme_bw()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



The plots above show that there is a positive trend between receptive and productive vocabulary sizes. The negatively skewed trend for Words and Gestures scores (plot on the left) is in line with the "bootstrap effect" on language development, where initial words are learned slowly and are followed by a "boom" on word production (e.g., Kachergis, Yu, and Shiffrin (2017), J. F. Werker and Gervain (2013)). On the other hand, we see an almost linear trend for scores on the Words and Sentences form, which is not surprising as it measures word learning for older infants that have already mastered a great deal of their language(s).

Building on these exploratory trends, we will now try to model the relationships between our predictors (i.e., age, form, language, sex, comprehension) and our outcome (i.e., productive vocabulary) using three machine learning approaches: regression trees, random forests, and linear regression.

2.4 Models

Regression Trees operate by predicting a continuous outcome variable Y by partitioning the predictors based on their relationships with the outcome. These partioning create a decision tree (that can be visualized as a flowchart) with predictions at the end of the tree (i.e., *nodes*). Mathematically, our model will partition predictors based on its' non-overlapping regions and on the amount of error reduction. We will calculate the RMSE between predicted and observed scores as a measure of model accuracy.

Random Forests are a common method to remedy some of the shortcoming from regression trees. On one side, it potentially improves prediction and reduces variability by averaging many regression trees that are randomly built using boostraping. On the other side, interpreting random forests is more challenging than interpreting regression trees. We will use the `randomForest` package to fit our model and we will assess the predictors' importance using the `caret` package. Again, we will calculate the RMSE between predicted and observed scores as a measure of model accuracy.

Linear Regression models linear relationships between the outcome and predictors and, assuming a fairly linear trend, allow us to predict the outcome score given some predictors scores. This is arguably the simplest

model from the three, but can be quite powerfull. It can also serve as a comparison point for the other two models. We will use the `lm()` function to fit our model and *RMSE* scores to evaluate its accuracy.

3 Results

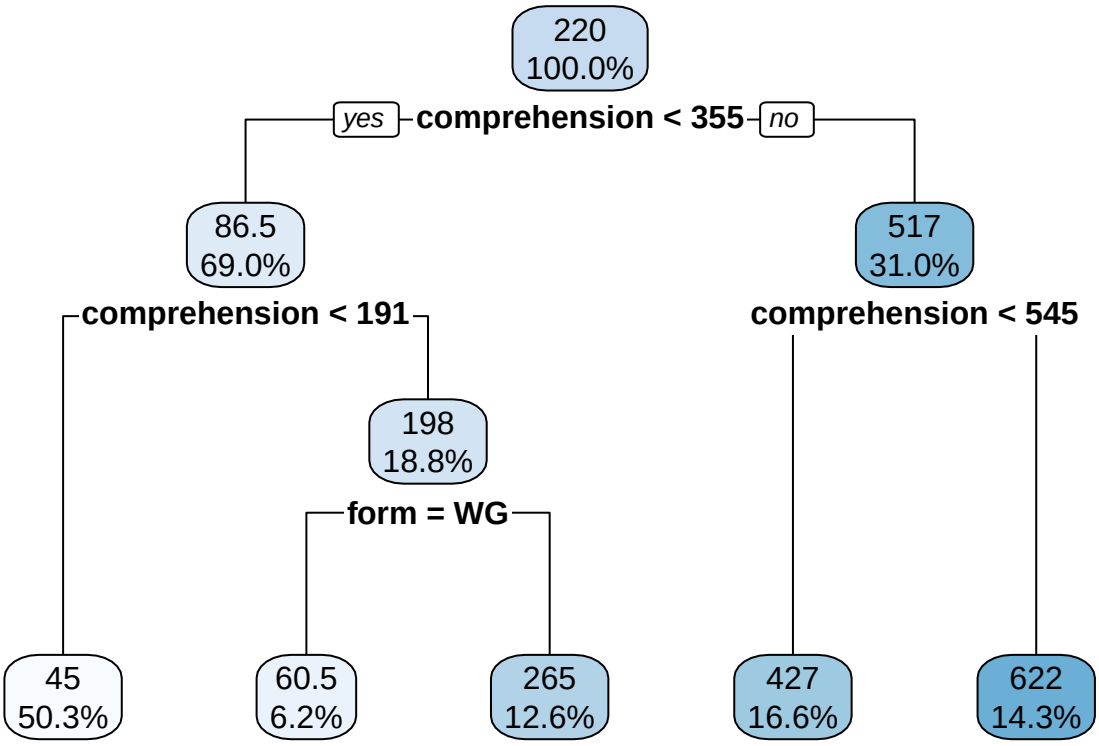
3.1 Regression Tree

Here we fit a regression tree to explore relationships between our predictors (i.e., age, sex, language, comprehension, form) and our outcome (i.e., productive vocabulary).

```
# set seed for reproducibility
if_else(getRversion() < 3.5, set.seed(234), set.seed(234, sample.kind = "Rounding"))
```

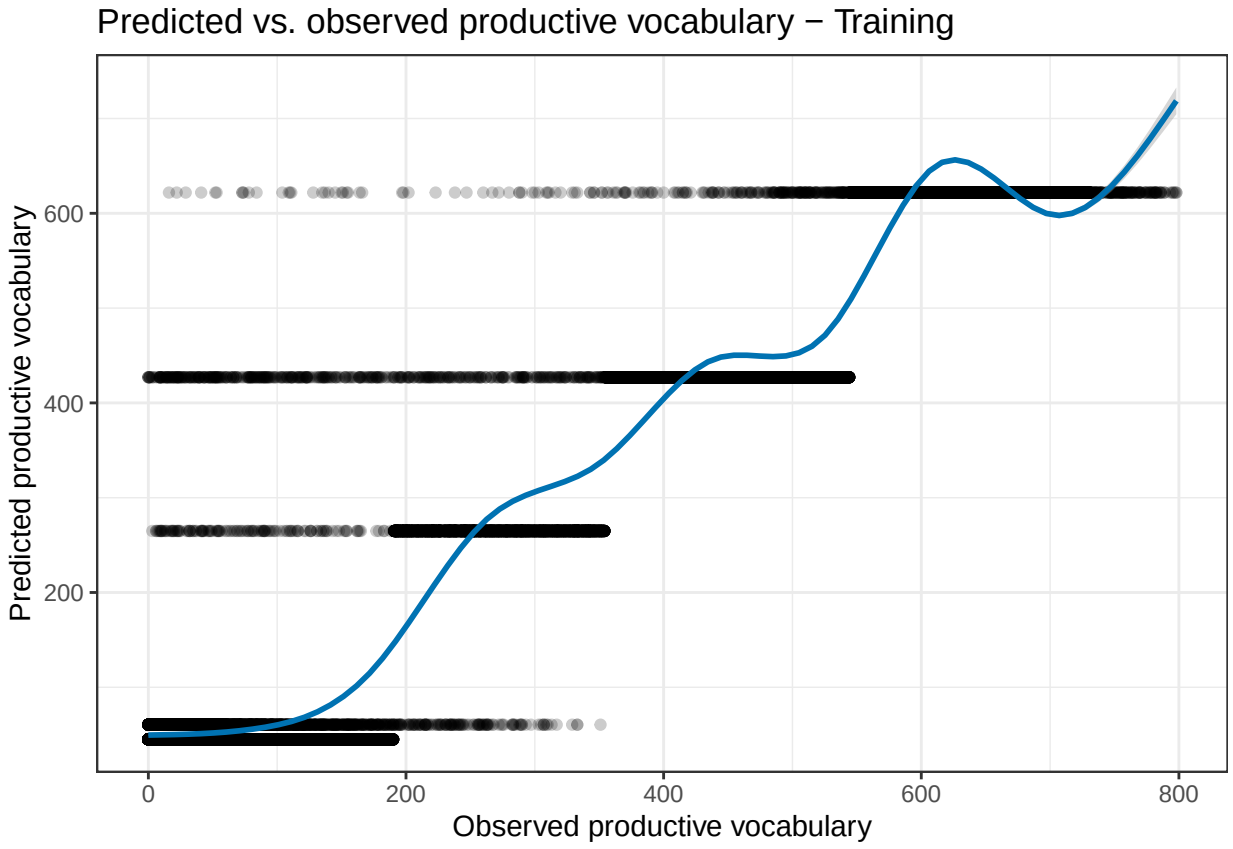
```
## NULL
```

```
# fit regression tree to training set
rpart_fit <- rpart(production ~ ., data = data_train, method = "anova")

# plot regression tree
rpart.plot(rpart_fit, digits = 3, fallen.leaves = T)
```



Our regression tree resulted in five branches/predictions, with comprehension (receptive vocabulary) being the most informative predictor (followed by form). Other predictors (i.e., sex, language, age) were did not reduce error sufficiently to allow the creation of new branches. Now we will measure the RMSE for the training set.

```
# calculate RMSE for training
rpart_pred_train <- data_train %>% mutate(pred_prod = predict(rpart_fit))

## plot predicted vs observed vocab
rpart_pred_train %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Training",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```

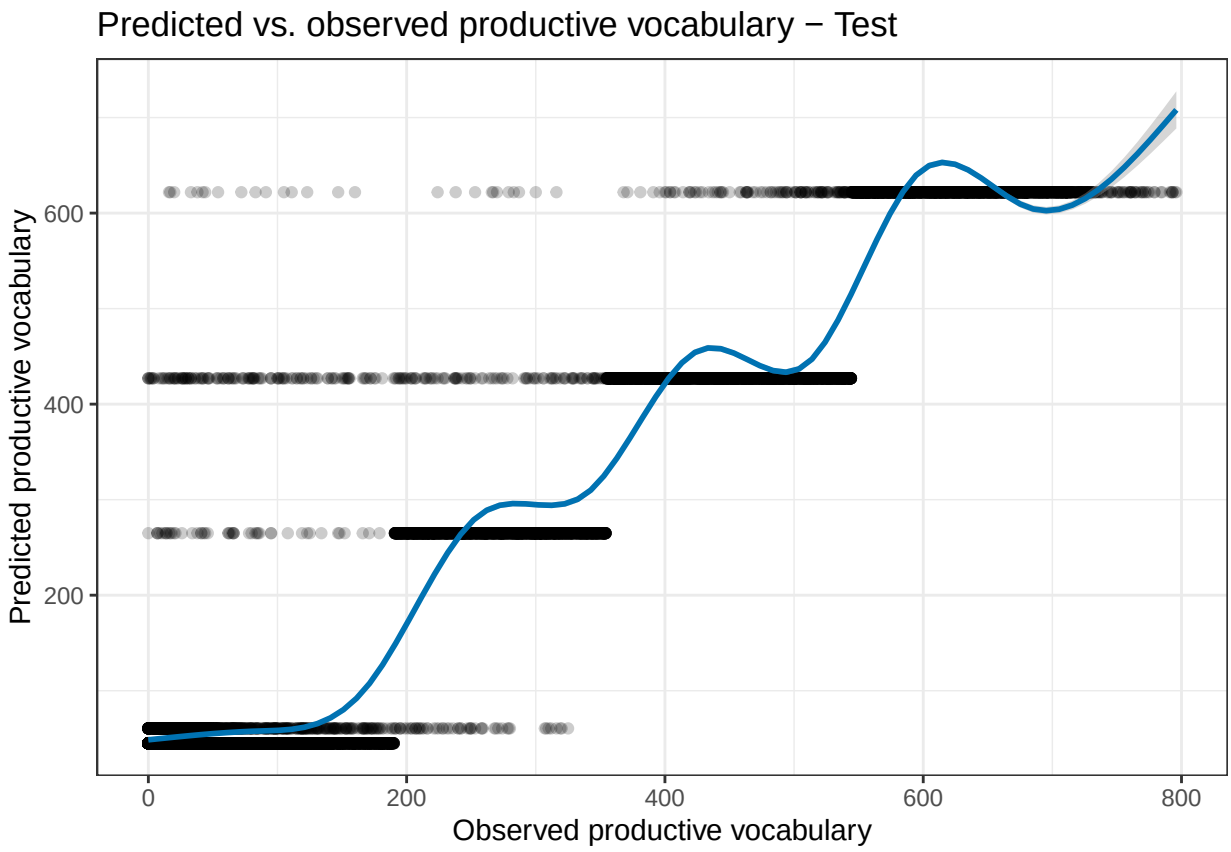



```
# training RMSE
rpart_train_rmse <- caret::RMSE(rpart_pred_train$production, rpart_pred_train$pred_prod)

# create a RMSE table
rmse_scores <- tibble(Model = "Regression Tree - Training",
                      RMSE = rpart_train_rmse)

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.589

The plot above shows that our five predicted branches roughly captures the observed scores in productive vocabulary in a staircase pattern. However, our predictions are off by 66.5889955 words, on average. Now we will apply the same regression tree to our test set and measure its' accuracy.

```
# rpart accuracy on test set
rpart_test <- predict(rpart_fit, data_test)
rpart_pred_test <- data_test %>% mutate(pred_prod = rpart_test)

# RMSE accuracy
## plot predicted vs observed vocab
rpart_pred_test %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Test",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```




```
# test RMSE
rpart_test_rmse <- caret::RMSE(rpart_pred_test$production, rpart_pred_test$pred_prod)
```

```
# add test RMSE to table
rmse_scores <- bind_rows(rmse_scores,
                         tibble(Model = "Regression Tree - Test",
                                RMSE = rpart_test_rmse))

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.58900
Regression Tree - Test	67.22226

Overall, we found similar "staircase" predictions on the test set. There was a slight increase in error, with our predictions being off by 67.222261 words (RMSE) in the test set–an increase of 0.6332655 from the training predictions. Overall regression trees followed the same pattern across training and test sets. Now we will fit random forests to explore whether it provides better predictions.

3.2 Random Forest

We now create a random forest: a set of regression trees and their average predictions. This might improve the accuracy of our predictions. We will also calculate the importance of each variable in this forest.

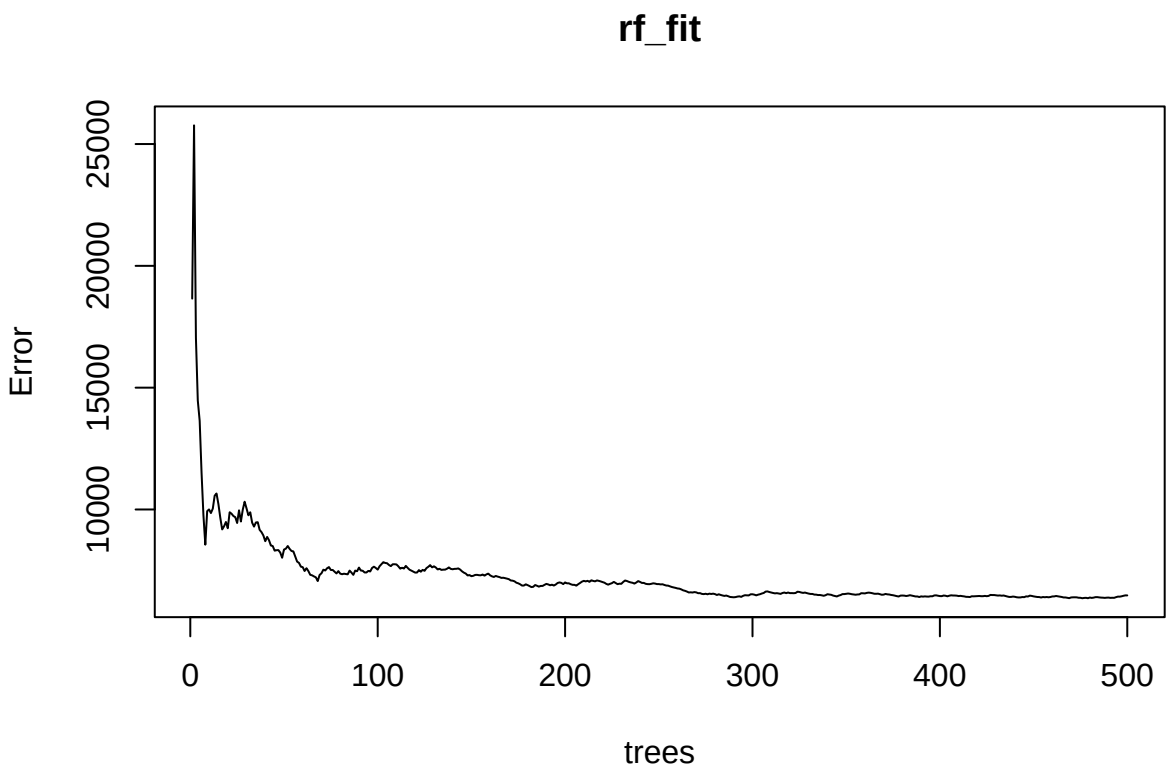
```
# set seed for reproducibility
if_else(getRversion() < 3.5, set.seed(345), set.seed(345, sample.kind = "Rounding"))
```

```
## NULL
```

```
# fit random forest on training set
rf_fit <- randomForest::randomForest(production ~ ., data = data_train)

# measure variable importance
varImp(rf_fit) %>% arrange(desc(Overall))
```

```
##                Overall
## comprehension 740356312
## age           479071686
## form          213883068
## language       56813489
## sex             9289708
```

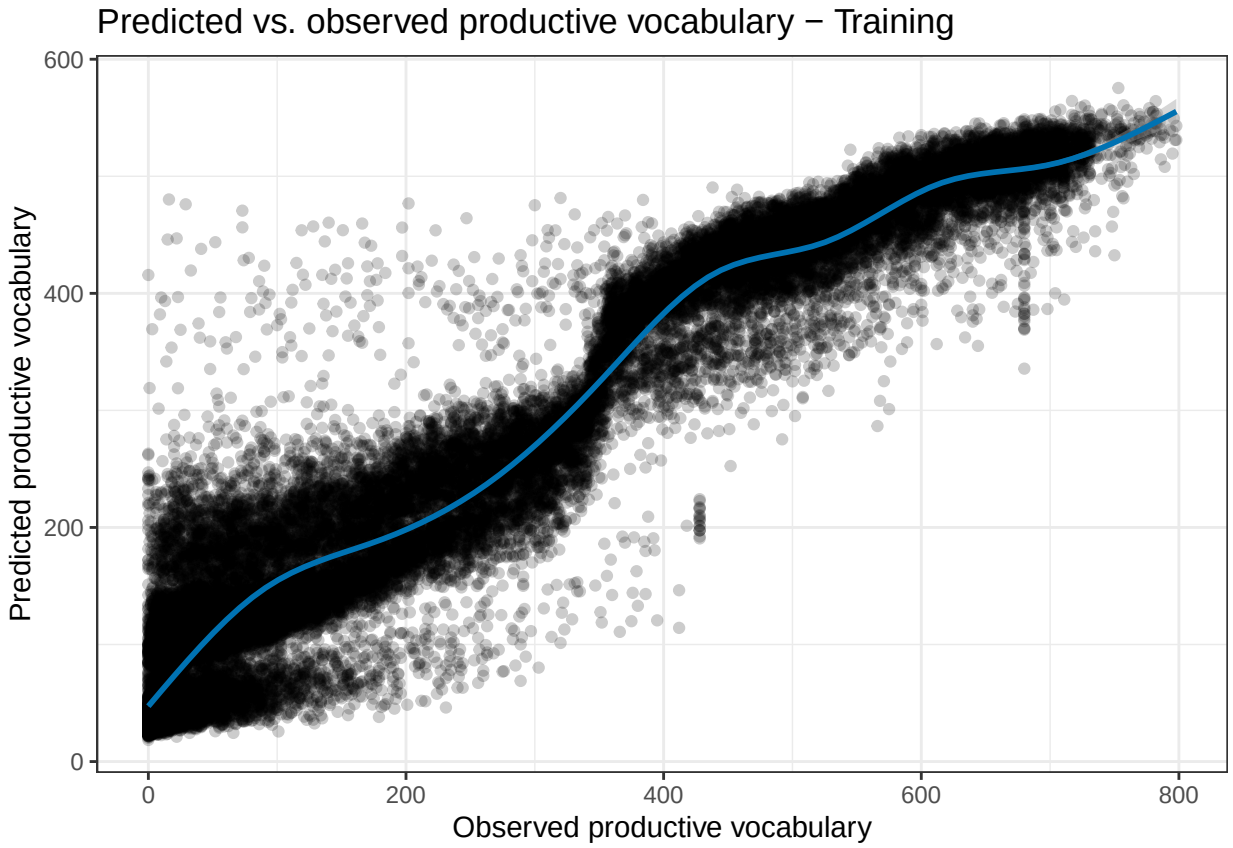
```
# plot random forest
plot(rf_fit)
```



The plot above (error vs. trees) indicate that around 300 trees the error stabilizes. Similarly to regression trees, comprehension was the most important predictor. Now we will measure the accuracy we achieved on the training set.

```
# calculate RMSE for training
rf_pred_train <- data_train %>% mutate(pred_prod = predict(rf_fit))

## plot predicted vs observed vocab
rf_pred_train %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Training",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



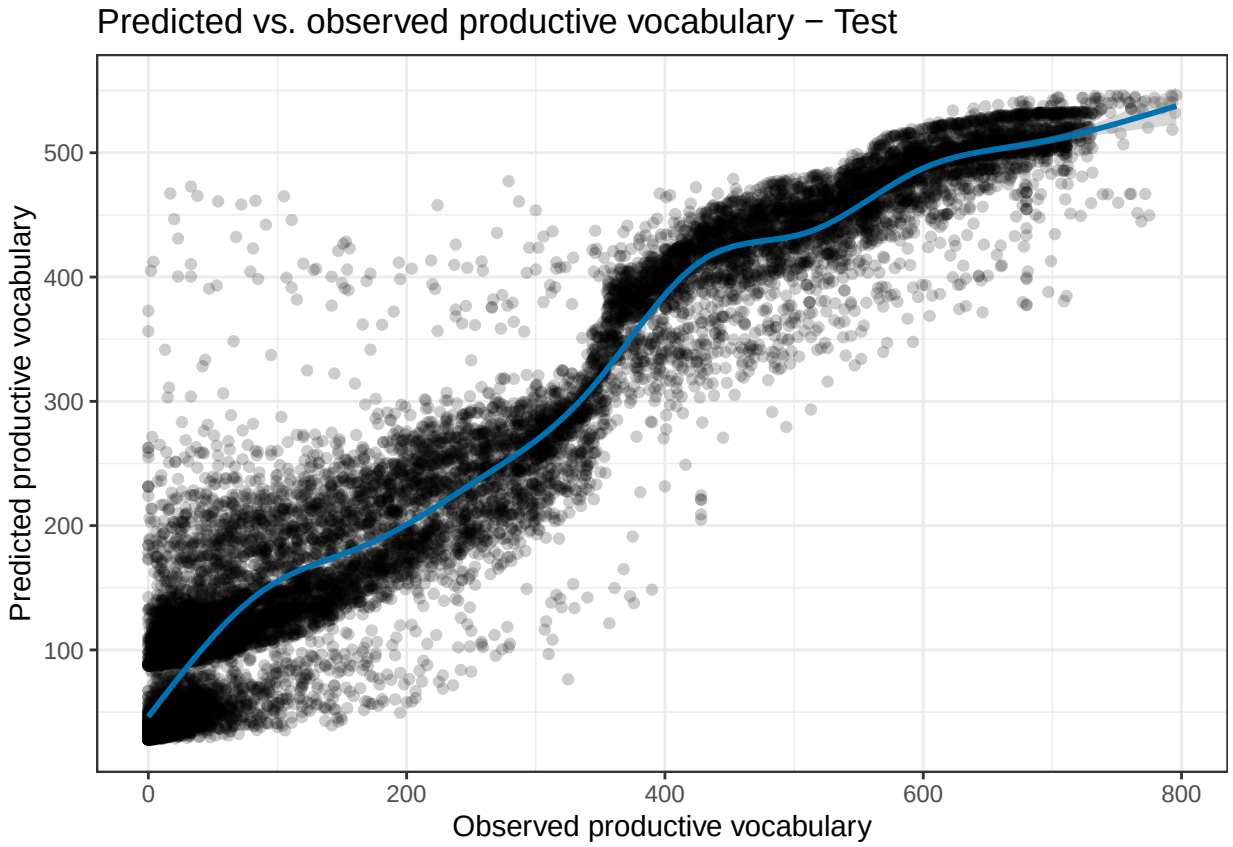

```
# training RMSE
rf_train_rmse <- caret::RMSE(rf_pred_train$production, rf_pred_train$pred_prod)

# create a RMSE table
rmse_scores <- bind_rows(rmse_scores,
                         tibble(Model = "Random Forest - Training",
                                RMSE = rf_train_rmse))

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.58900
Regression Tree - Test	67.22226
Random Forest - Training	80.44524

Despite the more complex approach (in comparison to Regression Trees), the training predictions were off by 80.4452363 words, on average. Now we will apply the same random forest model to our test set.

```
# rf accuracy on test set
rf_test <- predict(rf_fit, data_test)
rf_pred_test <- data_test %>% mutate(pred_prod = rf_test)

# RMSE accuracy
## plot predicted vs observed vocab
```

```
rf_pred_test %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Test",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```




```
# test RMSE
rf_test_rmse <- caret::RMSE(rf_pred_test$production, rf_pred_test$pred_prod)

# add test RMSE to table
rmse_scores <- bind_rows(rmse_scores,
                         tibble(Model = "Random Forest - Test",
                                RMSE = rf_test_rmse))

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.58900
Regression Tree - Test	67.22226
Random Forest - Training	80.44524
Random Forest - Test	80.20374

Despite the more complex approach, predicted scores for the test set were off by 80.2037434 words on average. Random forests were less accurate than regression trees when predicting production vocabulary for training (a deterioration of 13.8562408 words on average) and test sets (a deterioration of 12.9814823 words on average).

3.3 Linear Regression

Here we fit a linear regression model and check the estimated coefficients.

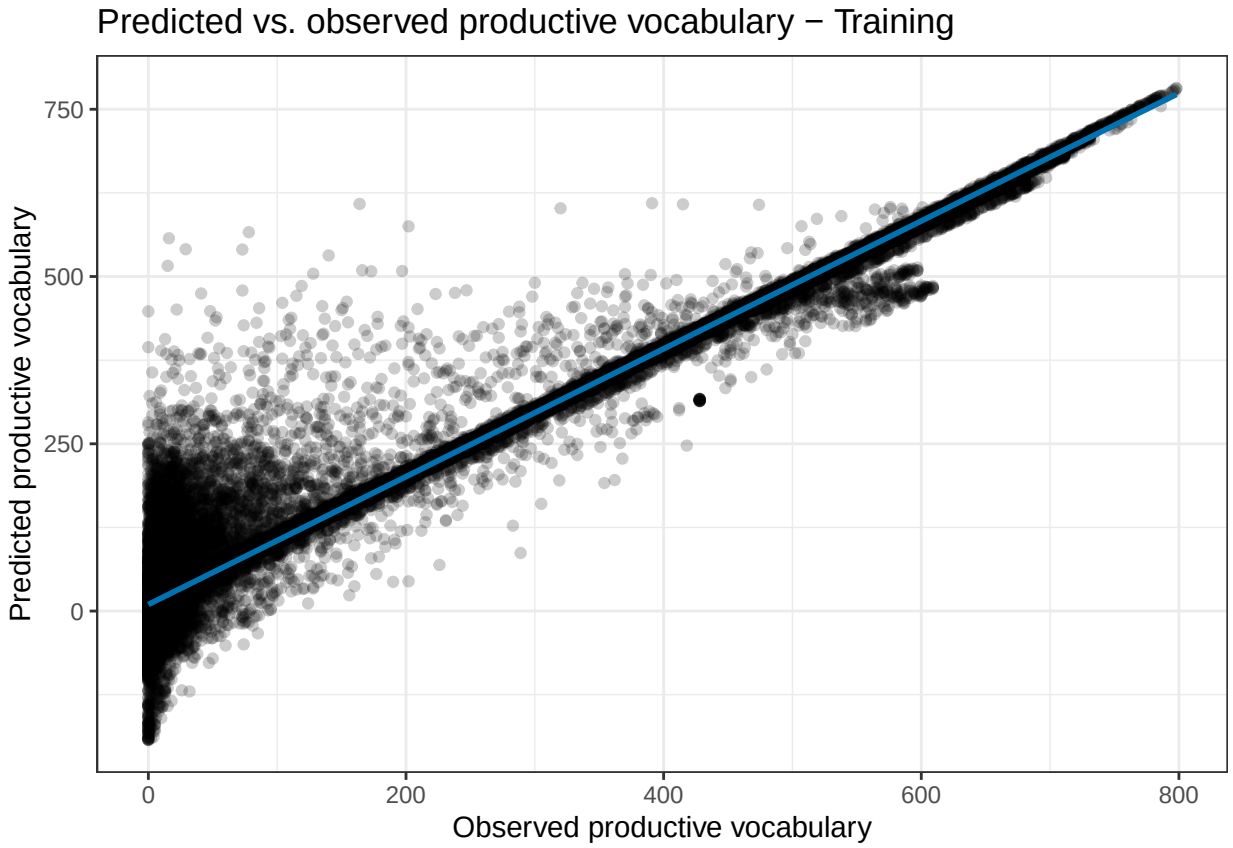
```
# fit lm on training set
lm_fit <- lm(production ~ ., data = data_train)

# model summary
summary(lm_fit)
```

```
##
## Call:
## lm(formula = production ~ ., data = data_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -541.21  -11.24    2.00   18.00  202.22
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -8.811e+01  2.439e+00 -36.129  < 2e-16 ***
## age                            5.354e-01  7.340e-02   7.294 3.07e-13 ***
## comprehension                  9.346e-01  1.816e-03 514.752  < 2e-16 ***
## languageRussian               -2.422e+01  2.650e+00  -9.139  < 2e-16 ***
## languageSwedish               -2.652e+00  2.761e+00  -0.960 0.336846
## languageTurkish               -3.954e+00  2.474e+00  -1.598 0.109989
## languageCantonese              5.042e-01  2.933e+00   0.172 0.863503
## languageMandarin (Beijing)     1.425e+01  2.921e+00   4.880 1.07e-06 ***
## languageFrench (Quebecois)     1.118e+00  2.752e+00   0.406 0.684579
## languageNorwegian              3.885e+00  2.342e+00   1.659 0.097171 .
## languageHebrew                -6.682e+01  3.403e+00 -19.635  < 2e-16 ***
## languageItalian               -9.793e+00  2.760e+00  -3.549 0.000388 ***
## languageBritish Sign Language  4.959e+01  5.081e+00   9.760  < 2e-16 ***
## languageSlovak                -1.071e+02  2.677e+00 -39.995  < 2e-16 ***
## languageKigiriama              9.978e+00  4.596e+00   2.171 0.029934 *
## languageKiswahili              4.466e+00  5.494e+00   0.813 0.416320
## languageGerman                -1.585e+00  2.853e+00  -0.555 0.578613
## languageDanish                 3.919e+00  2.397e+00   1.635 0.102092
## languageLatvian               -4.675e+00  3.189e+00  -1.466 0.142667
## languageEnglish (American)    -5.590e+00  2.368e+00  -2.361 0.018250 *
## languageEnglish (Australian)  -4.300e+00  2.742e+00  -1.569 0.116753
## languageCzech                 -5.560e+00  3.471e+00  -1.602 0.109256
## languageGreek (Cypriot)       -1.963e+00  4.944e+00  -0.397 0.691371
## languageSpanish (European)     2.894e+00  2.918e+00   0.992 0.321331
## languagePortuguese (European)  4.139e+00  2.443e+00   1.695 0.090175 .
## languageFrench (French)        3.521e+00  3.001e+00   1.173 0.240636
## languageMandarin (Taiwanese)   1.582e+00  2.544e+00   0.622 0.534166
## languageKorean                -6.942e+00  2.629e+00  -2.641 0.008279 **
```

```
## languageSpanish (Mexican)     -4.308e+00  2.629e+00  -1.639 0.101246
## formWS                         9.374e+01  8.499e-01 110.301  < 2e-16 ***
## sexMale                       -2.301e+00  4.840e-01  -4.754 2.00e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 48.21 on 40560 degrees of freedom
## Multiple R-squared:  0.9553, Adjusted R-squared:  0.9552
## F-statistic: 2.887e+04 on 30 and 40560 DF,  p-value: < 2.2e-16
```

A brief look into the models' summary reveal that it captured relationships we explored visually (see previous section). For instance, productive vocabulary increases with age, comprehension, and form (WS). On the other hand, it decreases for male children. Now we check the accuracy of predictions for the training set. Furthermore, our model explains 95% of the variance in our data (R^2), suggesting a good fit.

```
# RMSE accuracy
## add predicted scores to training set
lm_pred_train <- data_train %>% mutate(pred_prod = predict(lm_fit))

## plot predicted vs observed vocab
lm_pred_train %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(method = lm, color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Training",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```



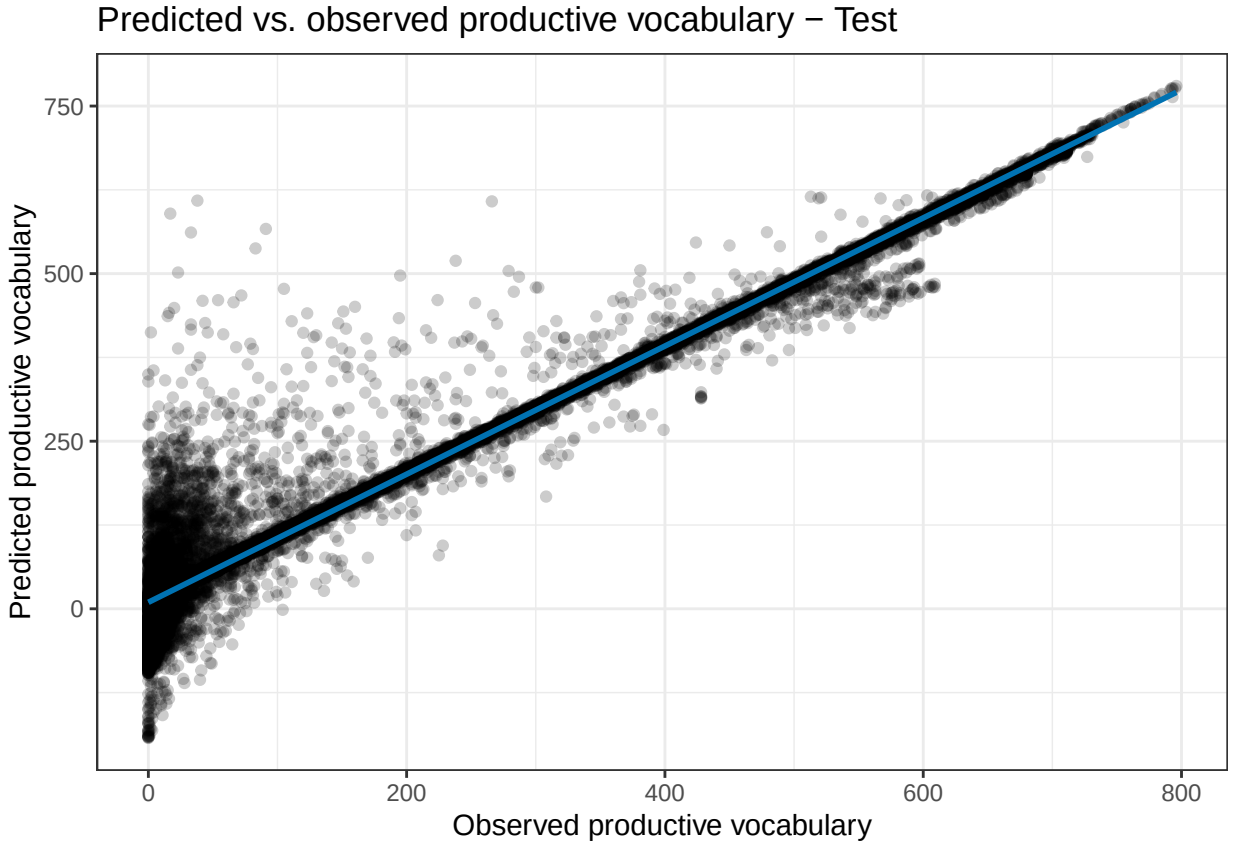

```
# training RMSE
lm_train_rmse <- caret::RMSE(lm_pred_train$production, lm_pred_train$pred_prod)

# add training RMSE to table
rmse_scores <- bind_rows(rmse_scores,
                         tibble(Model = "Linear Regression - Training",
                                RMSE = lm_train_rmse))

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.58900
Regression Tree - Test	67.22226
Random Forest - Training	80.44524
Random Forest - Test	80.20374
Linear Regression - Training	48.18790

Our linear model predictions were positively correlated with observed scores of vocabulary. The largest differences between predicted and observed values occur in the lower scores (close to 0) and accuracy improves as vocabulary increases. Overall, the predicted scores are off by 48.1878962 words (RMSE) in the training set. Now we will fit the same linear regression to our test set and check its accuracy (RMSE).

```
# lm accuracy on test set
lm_test <- predict(lm_fit, data_test)
```

```
lm_pred_test <- data_test %>% mutate(pred_prod = lm_test)

# RMSE accuracy
## plot predicted vs observed vocab
lm_pred_test %>%
  ggplot(aes(x = production, y = pred_prod)) +
  geom_point(alpha = 0.2) +
  geom_smooth(method = lm, color = color_blind_colors[5]) +
  labs(title = "Predicted vs. observed productive vocabulary - Test",
       x = "Observed productive vocabulary",
       y = "Predicted productive vocabulary") +
  theme_bw()
```




```
# test RMSE
lm_test_rmse <- caret::RMSE(lm_pred_test$production, lm_pred_test$pred_prod)

# add test RMSE to table
rmse_scores <- bind_rows(rmse_scores,
                         tibble(Model = "Linear Regression - Test",
                                RMSE = lm_test_rmse))

rmse_scores %>% knitr::kable()
```

Model	RMSE
Regression Tree - Training	66.58900
Regression Tree - Test	67.22226
Random Forest - Training	80.44524
Random Forest - Test	80.20374
Linear Regression - Training	48.18790
Linear Regression - Test	48.61879

Overall, we found very similar predictions on the test set. There was a slight increase in error, with our predictions being off by 48.6187915 words (RMSE) in the test set–an increase of 0.4308953 from training predictions. The positive trend between predicted and observed scores is also present in the test set. This indicates that our linear regression model performed at similar levels in both sets, which indicates its' stability as a predictive machine learning strategy.

Furthermore, our linear regression model was more accurate than regression trees and random forests. For instance, looking at the test sets, it was, on average, 18.6034695 words more precise than regression trees and 31.5849518 words more precise than random forests. We return to this point in the Conclusion.

4 Conclusion

The present project explored potential relationships between age, gender, languages, comprehension, measurement instrument, and productive vocabulary. To do so, we used an open repository of vocabulary growth from the WordBank project and three machine learning strategies (i.e., regression trees, random forests, and linear regression).

The arguably simplest model, linear regression, outperformed regression trees and random forests. On top of that, linear regression comes with the bonus of being easier to interpret and avoids the so called *black box* critic to machine learning. Overall, our linear regression model confirmed the trends we saw in the visualizations. For instance, productive vocabulary increases with age and comprehension across languages. Furthermore, it indicates that females have larger productive vocabularies than males. These trends are in line with previous reports in the language development literature (e.g., Michael C. Frank et al. (2021), Kachergis2017a, Werker2013).

The increase of team-science projects such as the WordBank (see also ManyLabs, ManyBabies, ManyPrimates etc.) have the potential to generate large and carefully collected datasets on complex topics such as language development. In line with recent recommendations (e.g., Jacobucci and Grimm (2020), Yarkoni and Westfall (2017)), scientists can take advantage of analytic approaches based on machine learning algorithms to explore and gain insights on these increasingly rich and complex data.

References

Fenson, Larry et al. 2007. *MacArthur-Bates Communicative Development Inventories*. Paul H. Brookes Publishing Company Baltimore, MD.

Frank, Michael C., Mika Braginsky, Daniel Yurovsky, and Virginia A. Marchman. 2017. "Wordbank: An open repository for developmental vocabulary data." *Journal of Child Language* 44 (3): 677–94. https://doi.org/10.1017/S0305000916000209.

Frank, Michael C, Mika Braginsky, Daniel Yurovsky, and Virginia A Marchman. 2021. *Variability and Consistency in Early Language Learning: The Wordbank Project*. MIT Press.

Jacobucci, Ross, and Kevin J. Grimm. 2020. "Machine Learning and Psychological Research: The Unexplored Effect of Measurement." *Perspectives on Psychological Science* 15 (3): 809–16. https://doi.org/10.1177/1745691620902467.

Kachergis, George, Chen Yu, and Richard M Shiffrin. 2017. "A Bootstrapping Model of Frequency and Context Effects in Word Learning." *Cognitive Science* 41 (3): 590–622. https://doi.org/10.1111/cogs.12353.

Werker, Janet F., and Judit Gervain. 2013. *Speech Perception in Infancy*. Edited by Philip David Zelazo. Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199958450.013.0031.

Werker, Janet, and Suzanne Curtin. 2005. "PRIMIR: A Developmental Framework of Infant Speech Processing." *Language Learning and Development* 1 (2): 197–234. https://doi.org/10.1207/s15473341lld0102_4.

Yarkoni, Tal, and Jacob Westfall. 2017. "Choosing Prediction Over Explanation in Psychology: Lessons From Machine Learning." *Perspectives on Psychological Science* 12 (6): 1100–1122. https://doi.org/10.1177/1745691617693393.