Who is good at learning language?

Sanaz Saadatifar

```
language_data <- read_csv("language_data.csv")
```

```
## Rows: 366911 Columns: 123
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (15): gender, native_languages, primary_languages, education, all_coun...
## dbl  (102): age, english_start, english_country_years, dictionary, english_c...
## lgl    (4): psychiatric_disorders, english_home, native_english, primary_eng...
## date   (1): date
## time   (1): time
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Introduction

This report displays the analysis results of a survey related to how well people did in an English-test which is about the proficiency in the English language. participants or a mixture of native and nonnative English speaking people from different countries all over the world. this report focuses only on nonnative people and their test results. Therefore the three main questions mentioned below are the aim of this analysis: Question 1: Does living in an English-speaking country result in more proficiency In English test? Question 2: does the time when participants started learning English (in which decade of their life), make a difference in how well they did in the English test? question 3: if the answer to above questions are yes then does spending more time in an English speaking country would result in a higher score in the English test? In the following sections first some exploratory data analysis will be done like the distribution of the test results, confidence intervals, hypothesis tastings and some other inferential analysis to gain insight about the whole population based on the given samples. and finally based on simple regression and multiple liberation models, a best model will be created to predict the proficiency in the English test.

Exploratory data analysis and ### Inference

The raw data is saved in language data, however only logit, orrect, education, gender, age, native_english, english_start, english_country_years are useful. Therefore a data cleaning process is done here. also new columns are added, such as “English_country_lived” which shows whether or not the participant had ever lived in an English speaking country, “Proportion_YearsLived_EnglishCountry” which shows how long the participant had lived in the English speaking country, and “english_start_categorical” that shows in which decade of the participants life has he or she started learning English.

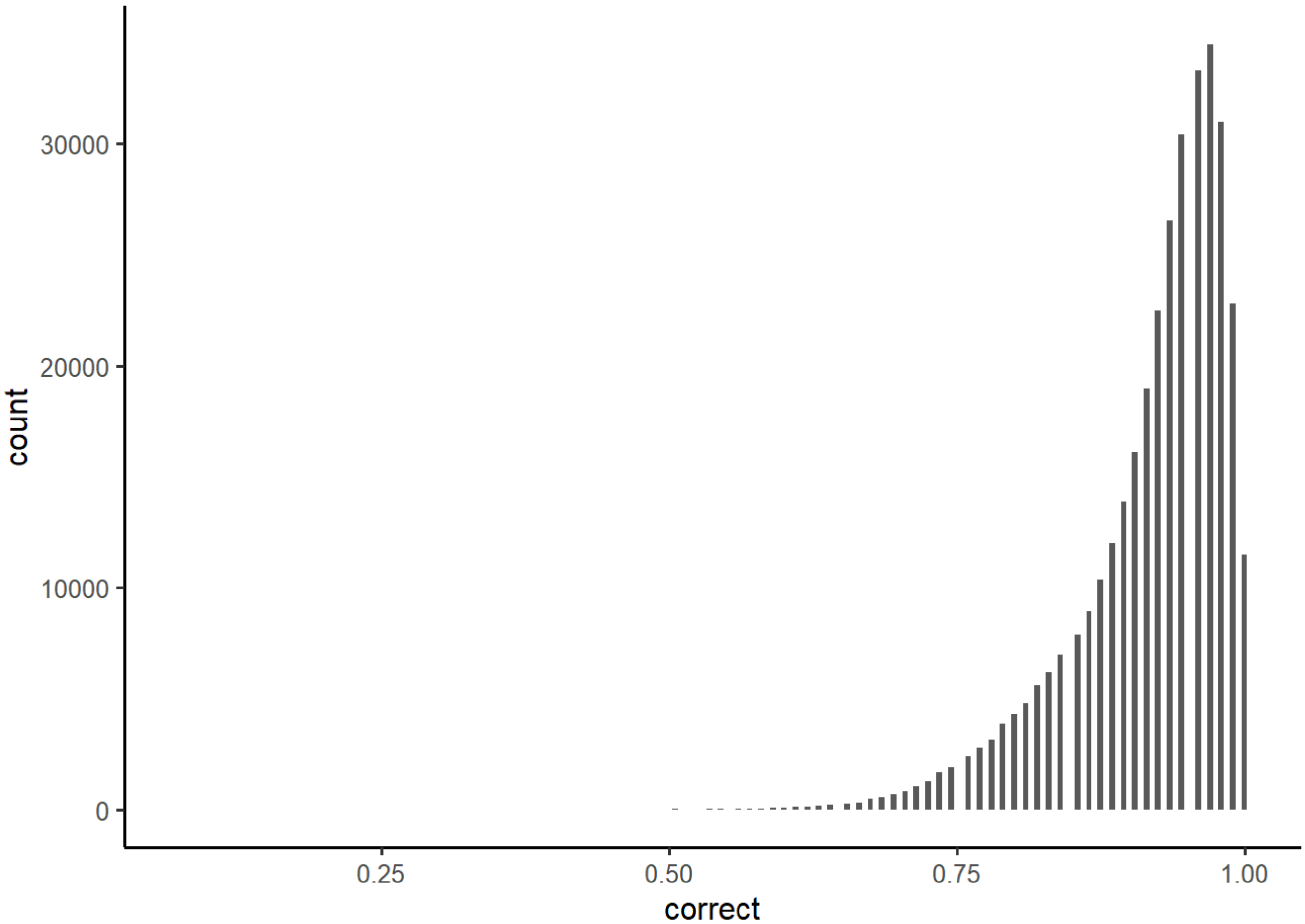
```r
language_data_analysis <- language_data %>%
  select(logit, correct, education, gender, age, native_english, english_start, english_country_
        years)


language_data_analysis <- language_data_analysis %>%
  mutate(english_country_lived = if_else(english_country_years > 0 , "Lived", "Did not live"))%
        >%
  mutate(Proportion_YearsLived_EnglishCountry = english_country_years/age) %>%
  filter(gender == "female" | gender == "male") %>%
  filter(native_english == "FALSE")


language_data_analysis <- language_data_analysis %>%
  mutate(english_start_categorical = case_when(english_start < 10 ~ "First Decade",
                                               english_start < 20 ~ "Second Decade",
                                               english_start < 30 ~ "Third Decade",
                                               english_start < 40 ~ "Fourth Decade",
                                               english_start < 50 ~ "Fifth Decade",
                                               english_start < 60 ~ "Sixth Decade",
                                               TRUE ~ "Seventh Decade"))
```

As the first data exploratory analysis, there distribution off “correct” which is The proportion of questions the participant answered correctly is shown below.

```r
language_data_analysis %>%
  summarise(mu = mean(correct),
            med = median(correct),
            sigma = sd(correct),
            iqr = IQR(correct))
```

```
## # A tibble: 1 x 4
##      mu   med  sigma    iqr
##   <dbl> <dbl>  <dbl>  <dbl>
## 1 0.919 0.937 0.0663 0.0842
```

```r
ggplot(language_data_analysis, aes(x = correct)) +
  geom_histogram(binwidth = 0.005)
```



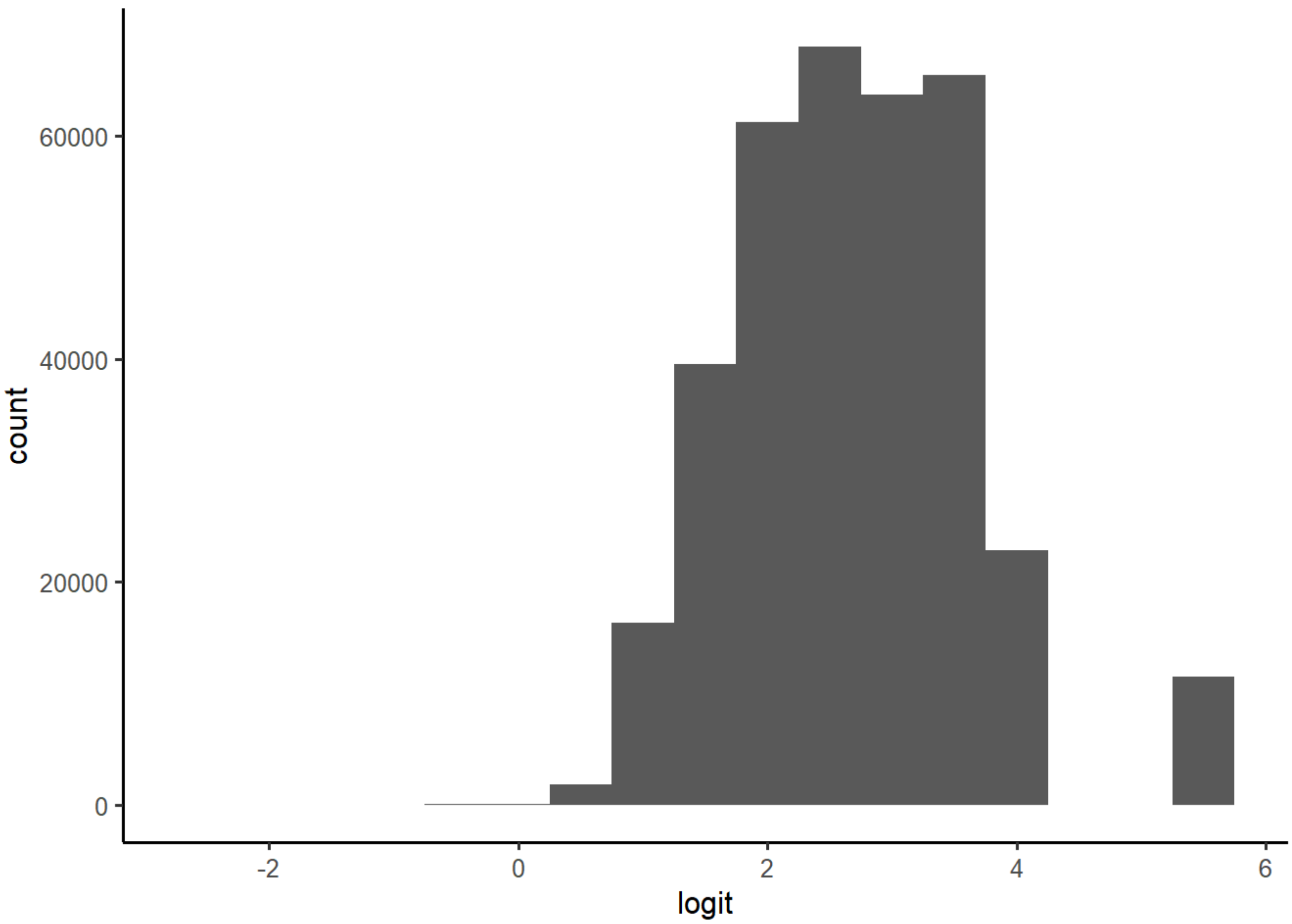

```
#language_data_analysis %>%
#   group_by(english_country_lived, gender) %>%
#   summarise(n = n()) %>%
#   group_by(gender)%>%
#   mutate(prop=n/sum(n))


#ggplot(language_data_analysis, aes(x = english_country_lived)) +
#   geom_bar(position = "dodge", show.legend=T)

#ggplot(language_data_analysis, aes(x = english_country_lived, fill = gender)) +
#   geom_bar(position = "dodge") +
#   theme(legend.position = c(.8,.8))
```

the distribution left-skewed, it would be better to use median, which is 0.9368421. however based on central limit theorem (CLT), the inferential analysis should be done on a normal distribution hence logit of correct we'll be analyzed as a transformation method.

```
language_data_analysis %>%
  summarise(mean = mean(logit),
            sd = sd(logit))
```

```
## # A tibble: 1 x 2
##    mean    sd
##   <dbl> <dbl>
## 1  2.68 0.928
```

```
ggplot(language_data_analysis, aes(x = logit)) +
  geom_histogram(binwidth = 0.5)
```



The distribution of logit, is roughly normal with the mean of 2.678788 and standard deviation of 0.9277099. Since the distribution is not too skewed we will use central limit theorem for further analysis based on mean. in terms of other conditions for central limit theorem (Other than normal sample distribution), Sample size is important and it requires to be large enough (more than 30), Which is the case in this study.

As the first inferential analysis, since the given data is a sample of the whole population, here I wanted to know what is the mean of the population's "logit" With 95% confidence interval.

```
#inferences 1.1
#Foundations for statistical inference - Confidence intervals



z_star_95 <- qnorm(0.975)
z_star_95
```

```
## [1] 1.959964
```

```
language_data_analysis %>%
  summarise(x_bar = mean(logit),
            sd = sd(logit),
            n = n(),
            se = sd(logit) / sqrt(n),
            me = z_star_95 * se,
            lower = x_bar - me,
            upper = x_bar + me)
```

```
## # A tibble: 1 x 7
##   x_bar    sd      n      se      me lower upper
##   <dbl> <dbl>  <int>   <dbl>   <dbl> <dbl> <dbl>
## 1  2.68 0.928 350440 0.00157 0.00307  2.68  2.68
```

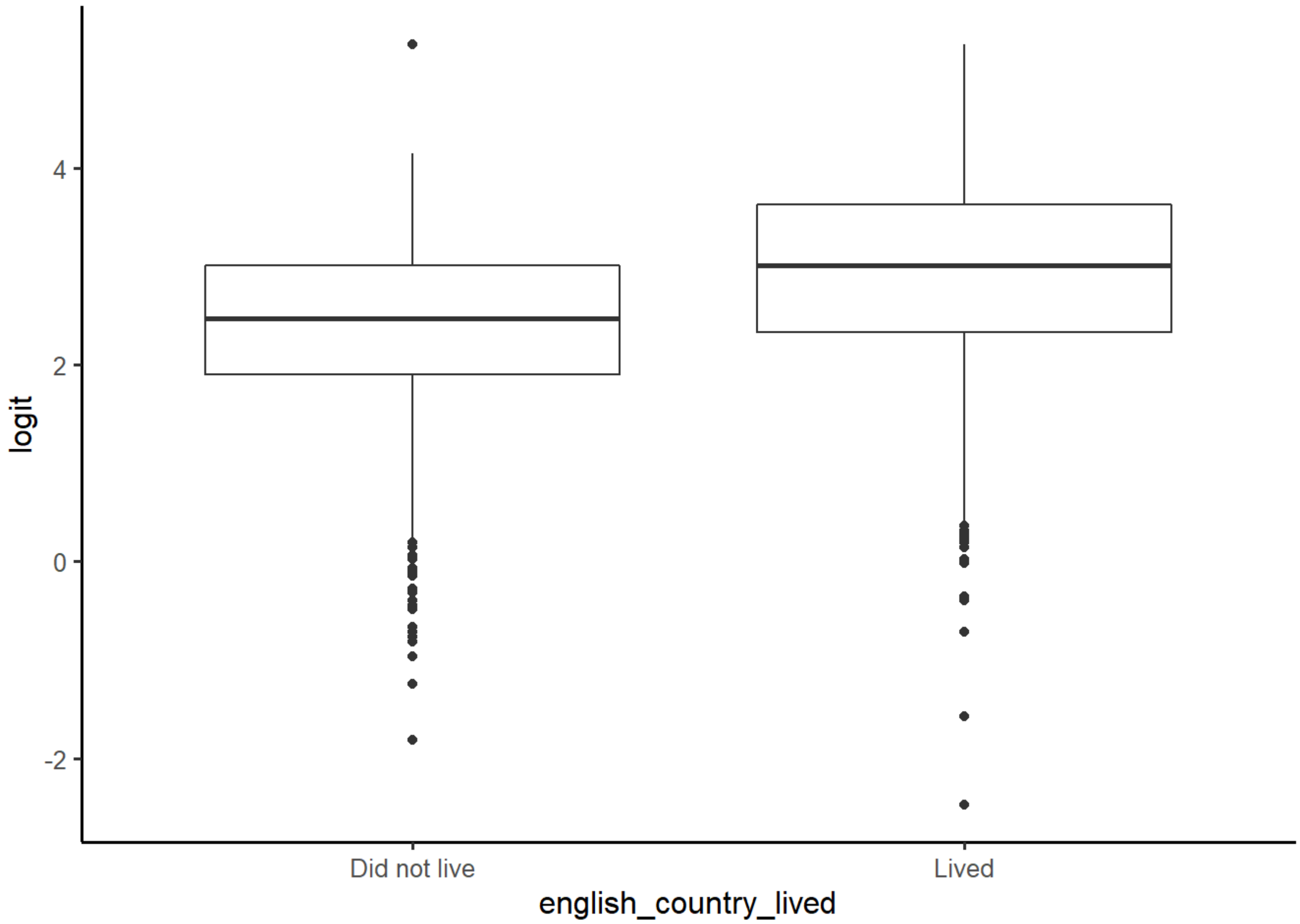
The mean of the populations logit is between (2.675716, 2.681859) with 95% CI.

To answer Question 1: Does living in an English-speaking country result in more proficiency In English test?, first and EDA is done regarding the differences between the mean od logits for two different group of people who lived in an English speaking country and people who didn't.

```
language_data_analysis %>%
  group_by(english_country_lived) %>%
  summarise(xbar = mean(logit),
            s = sd(logit),
            n = n())
```

```
## # A tibble: 2 x 4
##   english_country_lived  xbar     s      n
##   <chr>                 <dbl> <dbl>  <int>
## 1 Did not live           2.51 0.883 226852
## 2 Lived                  2.98 0.930 123588
```

```
ggplot(language_data_analysis, aes(x = english_country_lived, y = logit)) +
  geom_boxplot()
```



Based on above boxplot, it is shown that in this sample people who lived in an English speaking country has a higher logit mean compared to people who didn't. however to get inferences about the population, the hypothesis test should be done either with sampling and sampling simulation or with a T test.

H0: there is not a difference between the mean of logit between people who lived and those who did not live in an English speaking country. HA: there is a difference between the mean of logit between people who lived and those who did not live in an English speaking country.

```
t.test(logit~english_country_lived, data = language_data_analysis)
```

```
##
##  Welch Two Sample t-test
##
## data:  logit by english_country_lived
## t = -145.52, df = 242925, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group Did not live and group Lived i
s not equal to 0
## 95 percent confidence interval:
##  -0.4765164 -0.4638510
## sample estimates:
## mean in group Did not live        mean in group Lived
##                   2.512970                   2.983154
```

```
#People_lived <- language_data_analysis %>%
#  filter(english_country_lived == "Lived")

#People_not_lived <- language_data_analysis %>%
#  filter(english_country_lived == "Did not live")


#Lived_mean <- People_lived %>%
#  summarise(mu = mean(correct)) %>%
#  pull(mu)

#not_Lived_mean <- People_not_lived %>%
#  summarise(mu = mean(correct)) %>%
#  pull(mu)


#empirical_diff <- Lived_mean - not_Lived_mean
#empirical_diff







#diff_main <- language_data_analysis %>%
#  group_by(english_country_lived) %>%
#  summarise(mean = mean(correct)) %>%
#  ungroup() %>%
#  summarise(diff = first(mean) - last(mean))


#diff_main
```

based on the T-test result, the differences of the meand with 95 percent confidence interval falls within (-0.4765164 -0.4638510), since This range does not include zero then we have enough evidence to reject the null hypothesis in favor of alternative hypothesis.

To answer the Question 2: does the time when participants started learning English (in which decade of their life), make a difference in how well they did in the English test?, Since here are multiple categories such as different decades, the ANOVA or F-test is used.

H0:logit means do not vary across decades HA:logit means vary across decades

I want to look at the english proficiency test results (logit) in each decade. I want a histogram for each one. The standard ggplot way to do this is with facets, but that has a ton of white space and is hard to process. I’ll use a geom_density_ridges instead.

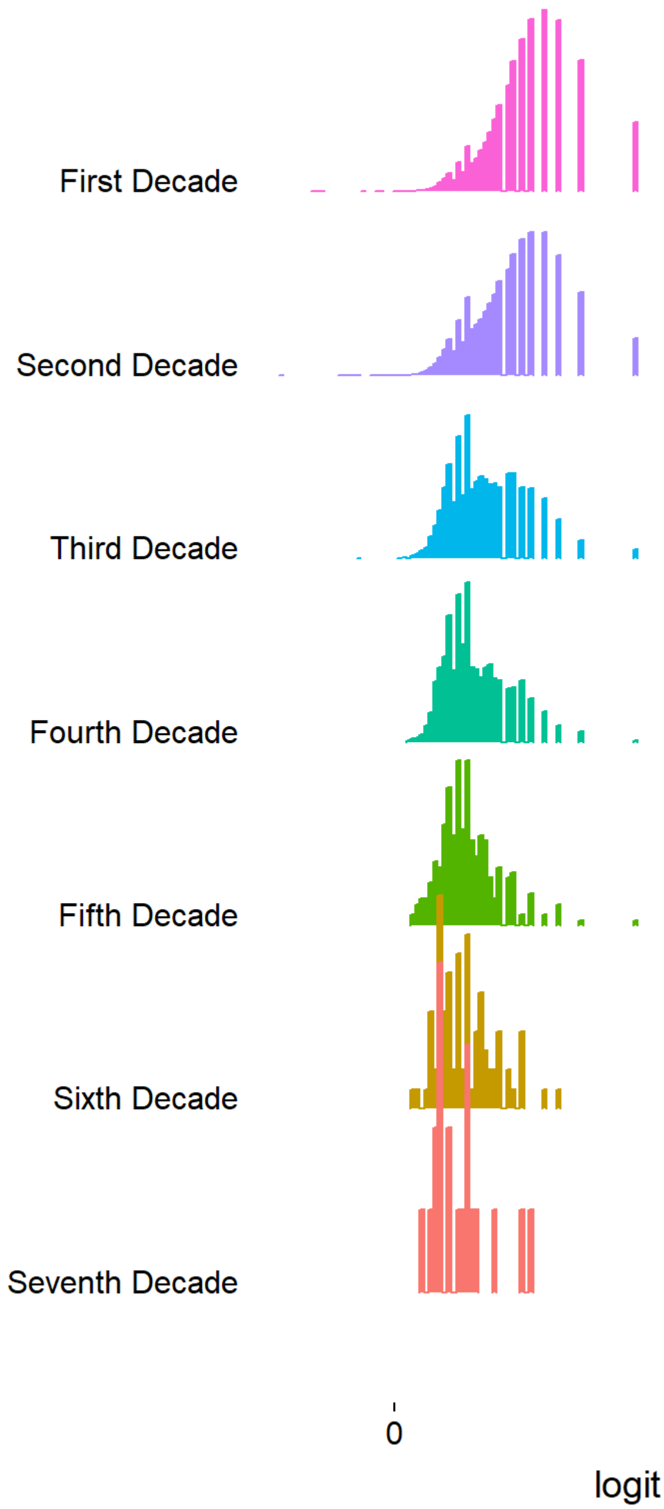
```
order <- language_data_analysis %>%
  group_by(english_start_categorical) %>%
  summarise(logit = mean(logit)) %>%
  arrange(logit)

order
```

```
## # A tibble: 7 x 2
##   english_start_categorical logit
##   <chr>                     <dbl>
## 1 Seventh Decade             1.43
## 2 Sixth Decade               1.57
## 3 Fifth Decade               1.63
## 4 Fourth Decade              1.75
## 5 Third Decade               1.97
## 6 Second Decade              2.53
## 7 First Decade               2.83
```

```
# I want to arrange the decade categories on my plot by their mean logit. To do this, I need to
          make a factor. A factor is like a string but it can have non-alphabetical orders.
ordered_data <- language_data_analysis %>%
  mutate(english_start_categorical = factor(english_start_categorical, levels = pull(order, engl
         ish_start_categorical)))

ordered_data
```

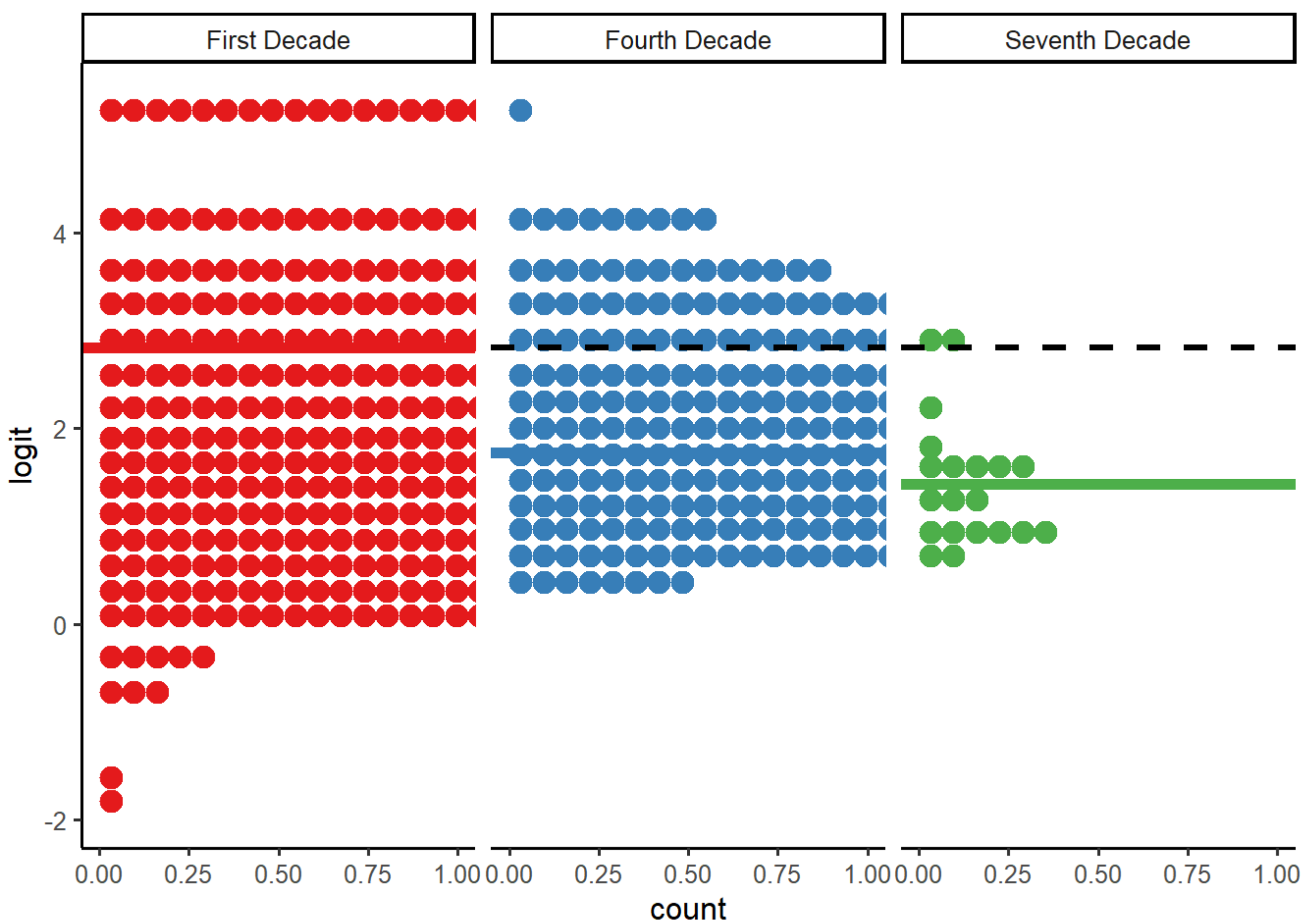
```
## # A tibble: 350,440 x 11
##    logit correct education            gender   age native_english english_start
##    <dbl>   <dbl> <chr>                <chr>  <dbl> <lgl>                  <dbl>
##  1  2.80   0.947 undergraduate degree male      53 FALSE                      0
##  2  3.62   0.979 some graduate        female    25 FALSE                      8
##  3  3.01   0.958 undergraduate degree female    20 FALSE                      6
##  4  1.57   0.832 graduate degree      male      37 FALSE                     18
##  5  3.62   0.979 graduate degree      female    29 FALSE                     12
##  6  2.80   0.947 graduate degree      female    29 FALSE                      6
##  7  2.47   0.926 graduate degree      male      31 FALSE                      3
##  8  2.80   0.947 graduate degree      male      29 FALSE                      5
##  9  2.80   0.947 high school degree   female    33 FALSE                      7
## 10  4.14   0.989 some undergraduate   male      39 FALSE                     11
## # ... with 350,430 more rows, and 4 more variables:
## #   english_country_years <dbl>, english_country_lived <chr>,
## #   Proportion_YearsLived_EnglishCountry <dbl>, english_start_categorical <fct>
```

```
# Make a ggridges plot so we can see all of the decades
ggplot(ordered_data, aes(x = logit, y = english_start_categorical,
                      fill = english_start_categorical, color = english_start_categorical)) +
  geom_density_ridges(stat = "binline", binwidth = 0.1, draw_baseline = FALSE) +
  scale_x_continuous(breaks =seq(0, 100, 20)) +
  labs(y = "") +
  theme_ridges(grid = FALSE, font_size = 14) +
  theme(legend.position = "none") # don't display the color legend. it's redundant
```



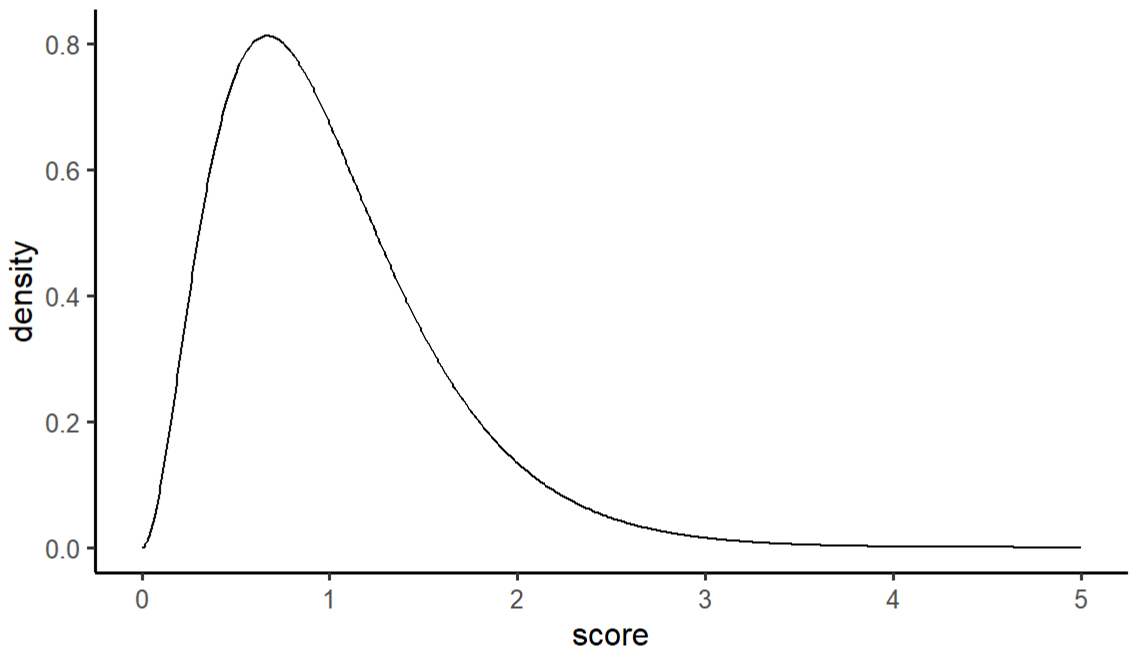
There clearly is a difference between the logit means of different decade categories in this sample. Let’s look at just 3 decades to understand what ANOVA is doing

```
three_categories <- language_data_analysis %>%
  filter(english_start_categorical %in% c("First Decade", "Fourth Decade", "Seventh Decade"))


grand_mean <- three_categories %>%
  summarise(logit = mean(logit)) %>%
  pull()

group_means <- three_categories %>%
  group_by(english_start_categorical) %>%
  summarise(logit = mean(logit))


ggplot(three_categories, aes(x = logit, fill = english_start_categorical, color = english_start_
       categorical)) +
  facet_wrap(~ english_start_categorical) +
  geom_dotplot() +
  coord_flip() +
  scale_fill_brewer(palette = "Set1") +
  scale_color_brewer(palette = "Set1") +
  geom_vline(aes(xintercept = grand_mean), linetype = "dashed", size = 1.2) +
  geom_vline(aes(xintercept = logit, color = english_start_categorical), size = 2,
             data = group_means)
```



Plot the F-distribution we'll use

```
df1 <- language_data_analysis %>%
  distinct(english_start_categorical) %>%
  nrow() - 1

df2 <- nrow(language_data_analysis) - df1 -1

fdist <- tibble(score = seq(0,5,.01),
                density = df(seq(0,5,.01), df1, df2))

ggplot(fdist, aes(x = score, y = density)) +
  geom_line()
```



Use ANOVA to determine if logit means vary across decades

```
english_start_categorical_anova <- aov(logit ~ english_start_categorical, data = language_data_a
         nalysis)

summary(english_start_categorical_anova)
```

```
##                               Df Sum Sq Mean Sq F value Pr(>F)
## english_start_categorical      6  12538  2089.7    2533 <2e-16 ***
## Residuals                 350433 289065     0.8
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Let's pull the f-value out of this analysis. I'll use the `tidy` function from the `broom` package which will give me back a tibble version of that same output

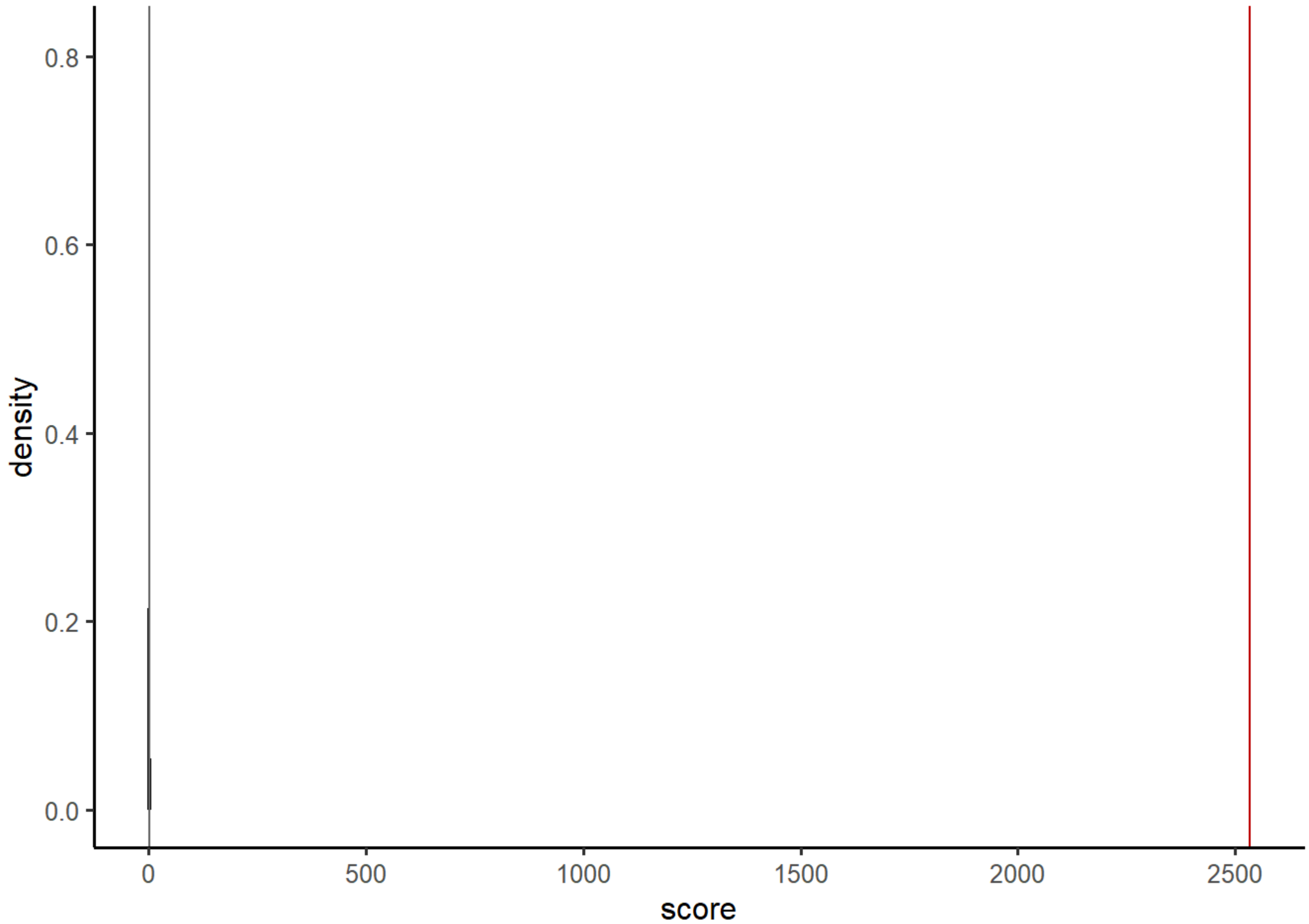
```
tidy_english_start_categorical_anova <- english_start_categorical_anova %>%
  tidy()

tidy_english_start_categorical_anova
```

```
## # A tibble: 2 x 6
##   term                          df   sumsq   meansq statistic p.value
##   <chr>                      <dbl>   <dbl>    <dbl>     <dbl>   <dbl>
## 1 english_start_categorical      6  12538. 2090.        2533.       0
## 2 Residuals                 350433 289065.    0.825       NA       NA
```

```
f_val <- tidy_english_start_categorical_anova %>%
  filter(term == "english_start_categorical") %>%
  pull(statistic)
```

Let's see where our data fall on the f-distribution

```
ggplot(fdist, aes(x = score, y = density)) +
  geom_line() +
  geom_vline(aes(xintercept = f_val), color = "#bb0000") +
  geom_vline(aes(xintercept = qf(.975, 15, 459)), color = "#666666") +
  geom_vline(aes(xintercept = qf(.025, 15, 459)), color = "#666666")
```



Since

the F_value is very small and the data line falls out of the F-distribution range, then we can reject the null hypothesis in favor of alternative hypothesis

Based on the analysis results so far it is clear that first decade they'd better in English tests compared to the 6thth decade which is the last one. Now I wanted to know in each decade category, what percent or proportion of people had lived in an English speaking country. To do so just the first decade and the last decade (the 6th) is selected For the comparison. the aim is to see whether there is a difference between the confidence interval for the proportion in each of these two categories.

```
First_Decade <- language_data_analysis %>%
  filter(english_start_categorical == "First Decade")

Sixth_Decade <- language_data_analysis %>%
  filter(english_start_categorical == "Sixth Decade")





First_Decade_prop <- First_Decade %>%
  summarise(First_Decade_lived_prop = mean(english_country_lived == "Lived")) %>%
  pull()

z_star <- qnorm(.975) #The 97.5% percentile of the normal distribution

se <- sqrt((First_Decade_prop * (1 - First_Decade_prop)) / nrow(First_Decade)) #the formula for
          the standard error

me <- se * z_star # margin of error is standard error times critical value

First_Decade_ci_95 <- c(First_Decade_prop - me, First_Decade_prop + me)

First_Decade_ci_95
```

```
## [1] 0.3912355 0.3956212
```

```
Sixth_Decade_prop <- Sixth_Decade %>%
  summarise(Sixth_Decade_lived_prop = mean(english_country_lived == "Lived")) %>%
  pull()

z_star <- qnorm(.975) #The 97.5% percentile of the normal distribution

se <- sqrt((Sixth_Decade_prop * (1 - Sixth_Decade_prop)) / nrow(Sixth_Decade)) #the formula for
          the standard error

me <- se * z_star # margin of error is standard error times critical value

Sixth_Decade_ci_95 <- c(Sixth_Decade_prop - me, Sixth_Decade_prop + me)

Sixth_Decade_ci_95
```

```
## [1] 0.2186743 0.4166198
```

with 95% confidence interval, the proportion of people lived in an English speaking country from the first decade category falls between (0.3912355 0.3956212) with 95% confidence interval, the proportion of people lived in an English speaking country from the first decade category falls between (0.3912355 0.3956212)

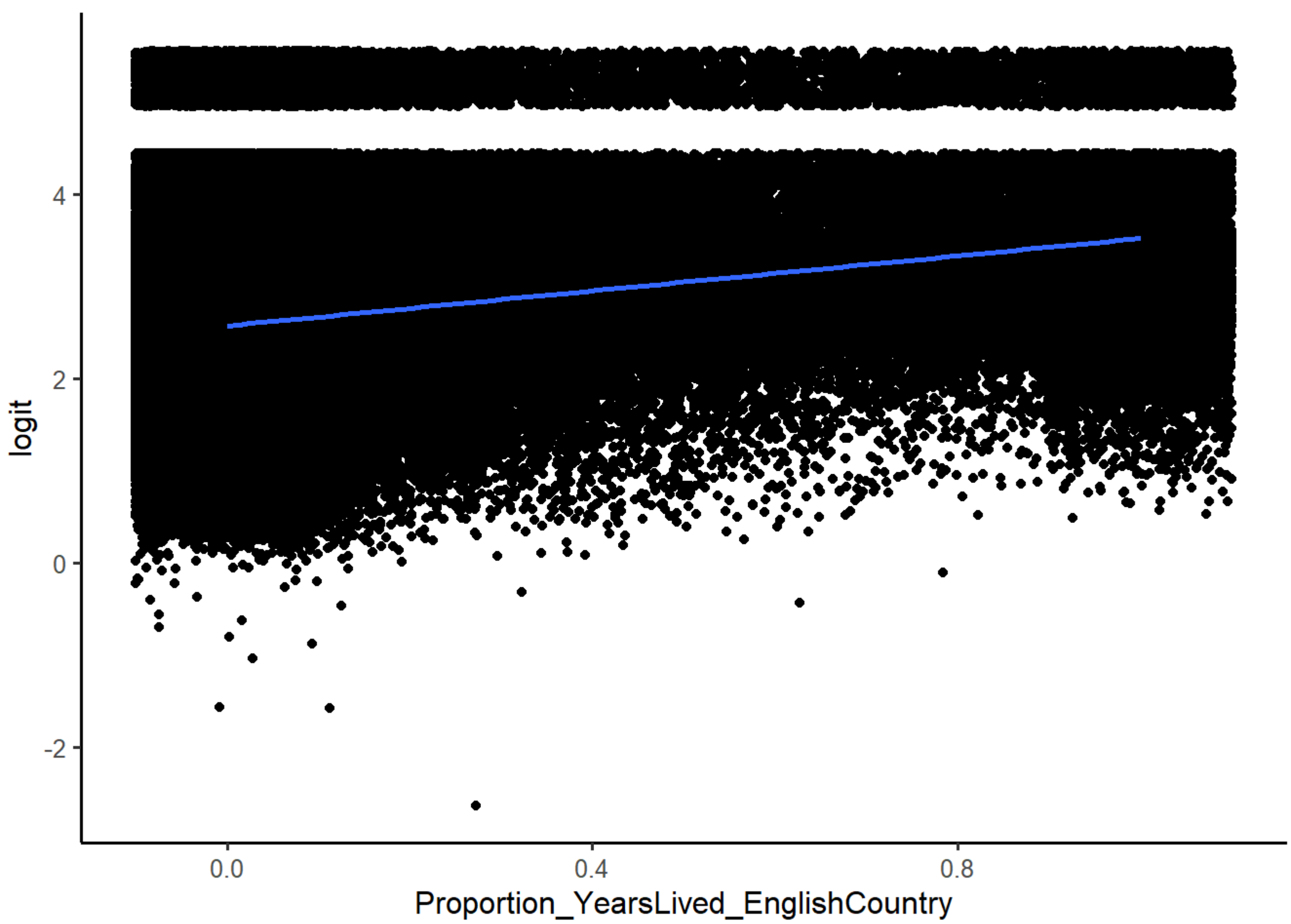
Modeling

In this section to answer to the question 3: does spending more time in an English speaking country would result in a higher score in the English test? I want to know whether there is a relation between the time spent leaving in an English speaking country and the logit (proficiency in English). to do so as simple regression model is built here.

H0: there is not a relationship between Proportion_YearsLived_EnglishCountry and logit HA: there is a relationship between Proportion_YearsLived_EnglishCountry and logit

```
#hala tu regression mitunm bepirsim ke cheghadr kharej budaneshun dar tule omreshun tasir dare?
ggplot(language_data_analysis, aes(x = Proportion_YearsLived_EnglishCountry, y = logit)) +
  geom_jitter(width = .1, height = .3) +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
#try geom_hex, this looks better
#play with geom_jitter(width = .1, height = .3)

language_data_analysis %>%
  summarise(cor = cor(logit, Proportion_YearsLived_EnglishCountry)) %>%
  pull()
```

```
## [1] 0.2585014
```

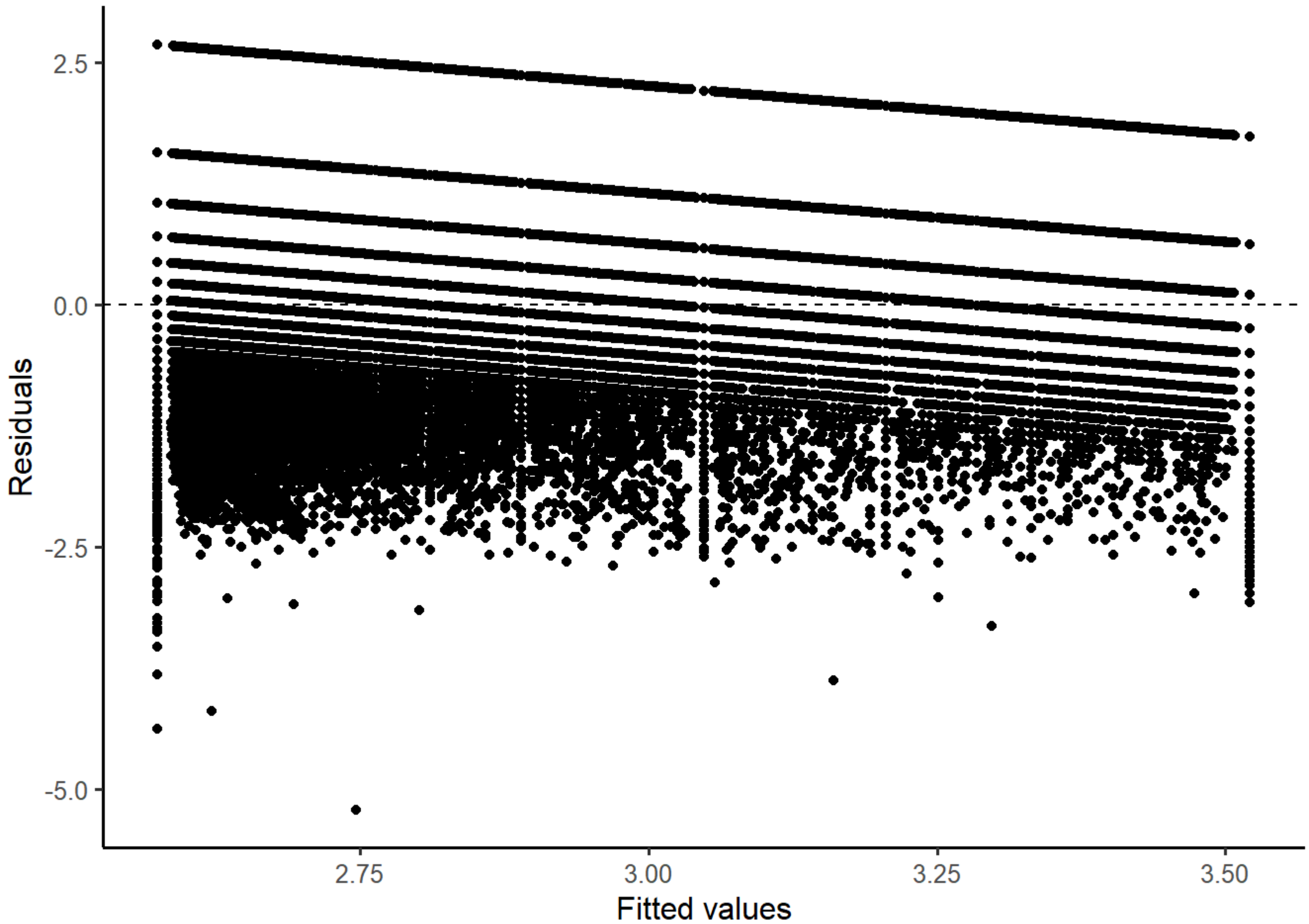
```
m1 <- lm(logit ~ Proportion_YearsLived_EnglishCountry, data = language_data_analysis)
summary(m1)
```

```
##
## Call:
## lm(formula = logit ~ Proportion_YearsLived_EnglishCountry, data = language_data_analysis)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2143 -0.6123 -0.0714  0.5063  2.6783
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          2.573996   0.001652  1558.0   <2e-16 ***
## Proportion_YearsLived_EnglishCountry 0.947321   0.005980   158.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8962 on 350438 degrees of freedom
## Multiple R-squared:  0.06682,    Adjusted R-squared:  0.06682
## F-statistic: 2.509e+04 on 1 and 350438 DF,  p-value: < 2.2e-16
```

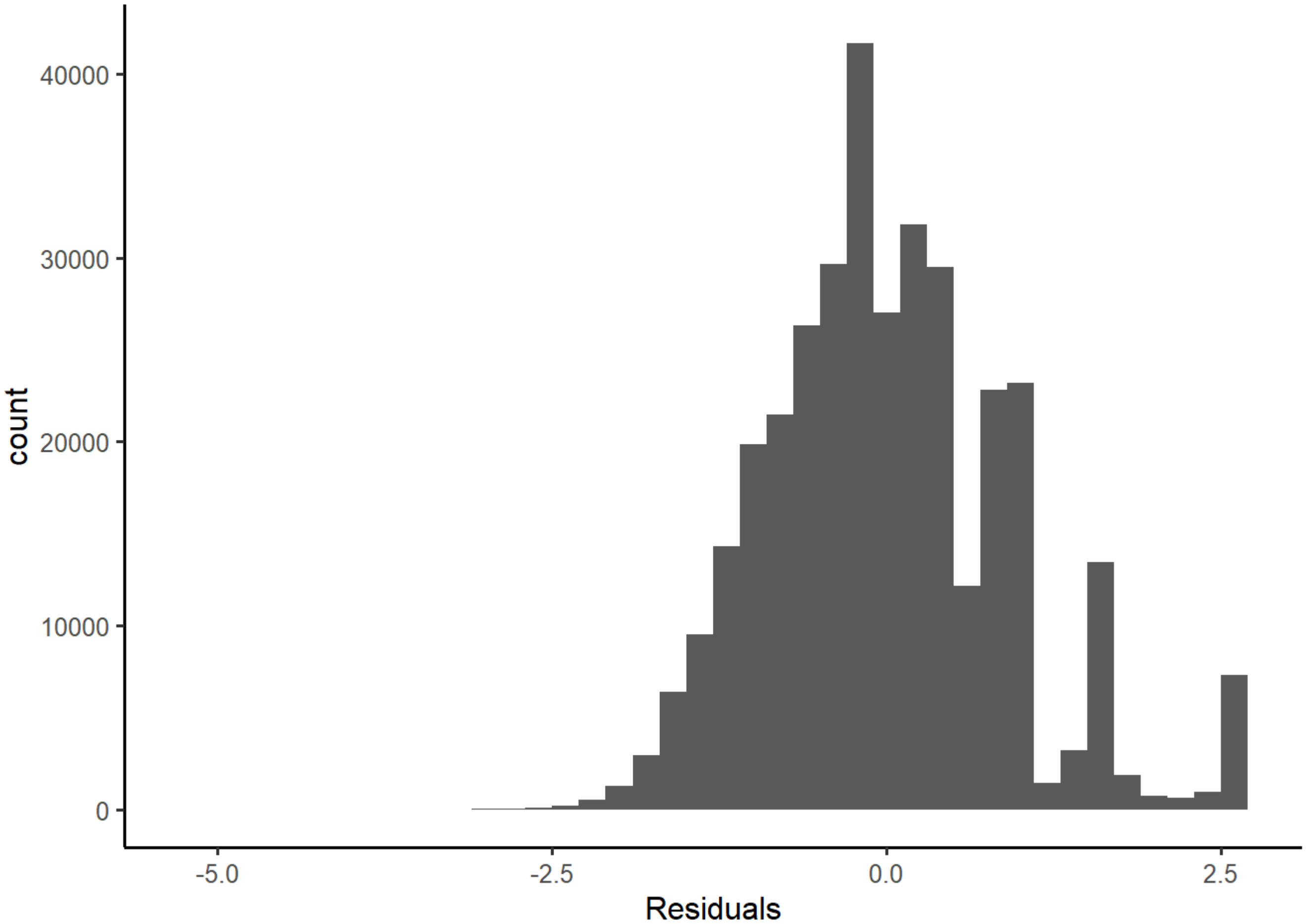
Since the slope is not zero and it's larger than zero, then there is a positive relationship between the Proportion_YearsLived_EnglishCountry and logit. Based on the blue line their relationship it's not that strong but it is positive. the correlation is 0.2585014, the slope is 0.947321, and the intercept is 2.573996. also the P-value is very small. Hence the simple regression formula would be: logit = 2.573996 + 947321*Proportion_YearsLived_EnglishCountry

However in order to use this formula for predictions, first we need to check if the conditions for the simple regression has been met.

```
#play with geom_jitter(width = .1, height = .3)

m1_residuals <- tibble(x = nrow(language_data_analysis),
                       fitted = fitted(m1),
                       resid = residuals(m1))

ggplot(m1_residuals, aes(x = fitted, y = resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  xlab("Fitted values") +
  ylab("Residuals")
```




```
ggplot(m1_residuals, aes(x = resid)) +
  geom_histogram(binwidth = 0.2) +
  xlab("Residuals")
```



The

first condition to use the simple regression is linearity Which the first model showed a clear linear relationship between logic and Proportion_YearsLived_EnglishCountry The second one is normality of the distribution of the residuals. the distribution about also clearly showed the normality. the third one is constant variability of residuals however the plot for that Didn't show a clear constant variability of residuals it seems roughly constant. so generally we may want to use this simple regression model for further predictions.

```
predict(m1, newdata = tibble(Proportion_YearsLived_EnglishCountry = 0.025))
```

```
##        1
## 2.597679
```

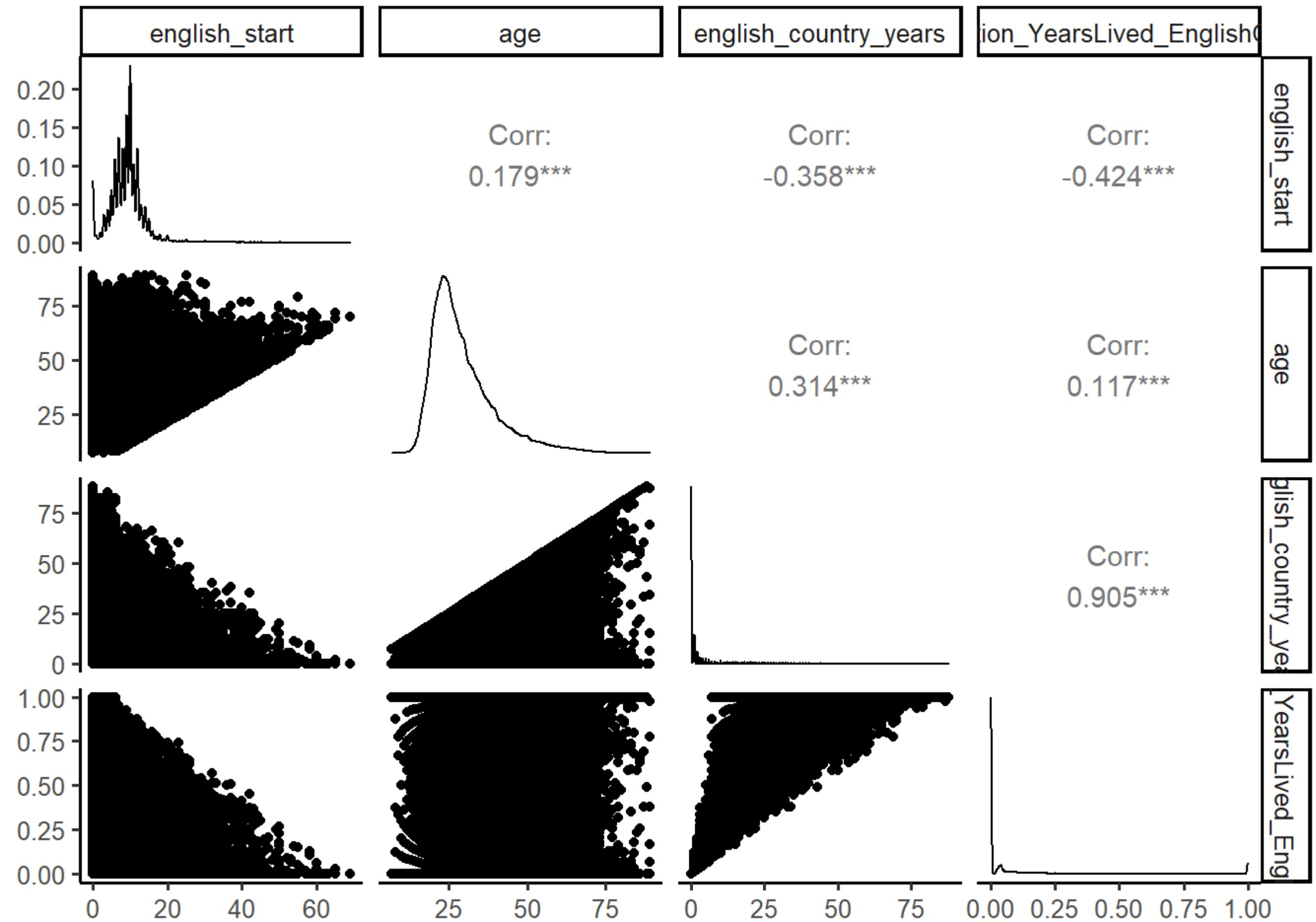
```
empirical_value <- filter(language_data_analysis, Proportion_YearsLived_EnglishCountry == 0.025)
         %>%
  pull(logit)

empirical_value
```

```
##   [1] 2.0971411 3.2744457 2.8006018 2.8006018 3.2744457 3.2744457 2.8006018
##   [8] 3.2744457 1.6474172 2.8006018 3.2744457 2.6224364 3.2744457 1.6474172
##  [15] 3.0122616 5.2522734 4.1431347 3.0122616 2.4680995 2.6224364 2.8006018
##  [22] 2.0971411 2.6224364 3.2744457 2.3315726 3.2744457 3.0122616 2.2088526
##  [29] 3.2744457 4.1431347 1.2427462 1.6474172 3.2744457 2.3315726 2.6224364
##  [36] 3.6216707 2.8006018 3.0122616 3.6216707 1.4325072 3.2744457 1.0169343
##  [43] 2.4680995 4.1431347 4.1431347 4.1431347 2.4680995 3.2744457 3.2744457
##  [50] 4.1431347 2.0971411 5.2522734 2.2088526 5.2522734 3.6216707 2.4680995
##  [57] 3.2744457 2.2088526 4.1431347 1.9944045 2.8006018 3.0122616 2.2088526
##  [64] 1.9944045 2.3315726 1.8991180 3.0122616 2.3315726 2.3315726 5.2522734
##  [71] 1.8101086 3.0122616 2.6224364 1.6474172 3.0122616 1.9944045 2.6224364
##  [78] 2.4680995 1.5008977 2.8006018 2.2088526 4.1431347 5.2522734 2.6224364
##  [85] 2.2088526 3.6216707 1.6474172 3.2744457 3.2744457 2.8006018 3.2744457
##  [92] 3.6216707 3.6216707 2.4680995 3.2744457 2.0971411 3.2744457 2.6224364
##  [99] 4.1431347 5.2522734 2.3315726 4.1431347 2.4680995 2.6224364 2.4680995
## [106] 5.2522734 4.1431347 2.6224364 2.6224364 3.6216707 2.2088526 2.3315726
## [113] 4.1431347 3.2744457 3.6216707 2.8006018 2.2088526 3.0122616 2.3315726
## [120] 3.0122616 3.0122616 3.2744457 4.1431347 1.8101086 3.0122616 2.8006018
## [127] 2.6224364 2.6224364 2.2088526 2.8006018 2.4680995 2.6224364 3.2744457
## [134] 2.8006018 4.1431347 3.0122616 1.9944045 1.8991180 3.0122616 2.2088526
## [141] 3.0122616 1.8101086 2.6224364 3.0122616 1.9944045 5.2522734 1.8991180
## [148] 4.1431347 2.4680995 3.6216707 1.1265861 3.6216707 3.6216707 2.8006018
## [155] 1.5008977 2.0971411 2.8006018 0.2301776 2.6224364 2.6224364 3.2744457
## [162] 2.4680995 4.1431347 2.0971411 1.4325072 1.8991180 3.6216707 1.1837701
## [169] 2.4680995 3.6216707 3.6216707 3.2744457 2.8006018 3.2744457 2.3315726
## [176] 1.1837701 3.0122616 2.4680995 1.5008977 1.5008977 2.8006018 4.1431347
## [183] 1.1837701 1.9944045 1.5008977 2.2088526 3.6216707 3.0122616 2.3315726
## [190] 2.8006018 2.3315726 2.3315726 4.1431347 1.8991180 2.8006018 4.1431347
## [197] 1.0169343 0.9126477 4.1431347 1.4325072 3.2744457 2.8006018 3.0122616
## [204] 3.6216707 3.0122616 1.9944045 3.2744457 2.8006018 3.6216707 3.0122616
## [211] 2.8006018 3.6216707 3.0122616 3.2744457 2.8006018 2.8006018 2.6224364
## [218] 2.4680995 2.8006018 2.4680995 1.8991180 3.0122616 3.0122616 3.2744457
## [225] 3.6216707 3.2744457 2.3315726 1.8991180 3.2744457 2.3315726 3.6216707
## [232] 1.7264544 3.2744457 3.2744457 3.2744457 3.6216707 2.2088526 1.0169343
## [239] 2.6224364 1.8991180 2.4680995 2.8006018 3.6216707 3.2744457 2.0971411
## [246] 2.6224364 1.8991180 4.1431347 2.8006018 3.0122616 2.6224364 2.6224364
## [253] 3.6216707 2.6224364 2.4680995 1.9944045 5.2522734 2.3315726 2.4680995
## [260] 2.4680995 3.2744457 3.6216707 3.0122616 1.9944045 2.3315726 2.8006018
## [267] 3.0122616 3.6216707 3.0122616 2.6224364 3.0122616 3.2744457 1.9944045
## [274] 3.2744457 2.8006018 3.6216707 1.9944045 1.6474172 3.6216707 5.2522734
## [281] 3.2744457 3.0122616 2.6224364 2.8006018 4.1431347 5.2522734 2.8006018
## [288] 2.8006018 2.8006018 3.6216707 3.2744457 3.6216707 2.8006018 2.4680995
## [295] 3.2744457 3.6216707 2.4680995 3.0122616 4.1431347 2.2088526 0.7643235
## [302] 1.8991180 3.6216707 4.1431347 3.0122616 1.8991180 3.6216707 2.4680995
## [309] 4.1431347 3.6216707 1.9944045 3.2744457 2.6224364 3.2744457 2.4680995
## [316] 1.5008977 1.6474172 3.0122616 3.6216707 2.8006018 1.0169343 1.3037078
## [323] 3.0122616 2.0971411 3.6216707 2.4680995 3.2744457 2.8006018 2.8006018
## [330] 3.0122616 2.6224364 2.4680995 2.8006018 4.1431347 3.2744457 1.1265861
## [337] 2.3315726 5.2522734 1.3668763 2.2088526 2.4680995 2.3315726 1.8991180
## [344] 2.3315726 3.6216707 2.4680995 3.2744457 4.1431347 2.6224364 2.8006018
## [351] 2.6224364 2.6224364 3.2744457 2.6224364 3.6216707 5.2522734 2.4680995
## [358] 2.6224364 3.0122616 1.8991180 2.0971411 2.2088526 3.2744457 2.8006018
```

```
## [365] 3.0122616 2.8006018 1.5723966 4.1431347 2.3315726 2.6224364 2.6224364
## [372] 3.2744457 1.6474172 5.2522734 3.2744457 1.5008977 4.1431347 4.1431347
## [379] 1.5723966 0.9126477 2.3315726 1.7264544 4.1431347 2.8006018 3.2744457
## [386] 3.2744457 2.6224364 2.4680995 2.8006018 1.3037078 2.8006018 3.0122616
## [393] 2.8006018 3.0122616 2.3315726 3.6216707 2.4680995 1.9944045 3.6216707
## [400] 3.0122616 3.0122616 2.8006018 5.2522734 2.6224364 3.2744457 3.6216707
## [407] 3.6216707 3.6216707 4.1431347 2.4680995 3.2744457 2.6224364 5.2522734
## [414] 3.6216707 3.2744457 3.2744457 0.9641820 3.2744457 2.6224364 3.0122616
## [421] 2.6224364 1.5723966 3.2744457 1.8991180 1.8991180 3.6216707 2.6224364
## [428] 3.2744457 2.6224364 5.2522734 2.3315726 1.8991180 3.6216707 3.2744457
## [435] 3.6216707 2.2088526 2.4680995 1.9944045 3.0122616 4.1431347 1.8991180
## [442] 1.0710243 2.8006018 1.3037078 3.0122616 2.2088526 2.3315726 1.0169343
## [449] 3.0122616 1.8991180 1.4325072 2.6224364 1.5723966 1.8101086 1.3037078
## [456] 1.3668763 2.8006018 2.0971411 1.0710243 2.0971411 3.0122616 2.6224364
## [463] 1.9944045 2.4680995 1.0169343 2.8006018 1.9944045 3.0122616 4.1431347
## [470] 3.0122616 4.1431347 1.5723966 1.3668763 1.8101086 2.4680995 2.8006018
## [477] 1.8101086 3.2744457 2.8006018 2.2088526 3.6216707 2.6224364 2.8006018
## [484] 1.9944045 2.3315726 1.8101086 3.0122616 3.6216707 1.6474172 4.1431347
## [491] 4.1431347 1.6474172 1.9944045 2.4680995 4.1431347 2.8006018 1.6474172
## [498] 1.0169343 1.7264544 3.6216707 2.6224364 2.3315726 2.2088526 5.2522734
## [505] 3.2744457 1.8101086 3.2744457 2.8006018 1.9944045 2.0971411 1.5723966
## [512] 3.2744457 2.8006018 2.8006018 2.2088526 3.2744457 1.0169343 1.6474172
## [519] 1.1837701 3.6216707 1.1265861 0.8128117 2.6224364 1.8101086 2.8006018
## [526] 1.7264544 2.2088526 2.3315726 1.9944045 2.2088526 1.3668763 3.2744457
## [533] 2.3315726 4.1431347 1.8991180 0.9126477 1.4325072 1.0169343
```

Although previous simple regression analysis showed Proportion_YearsLived_EnglishCountry has a linear relation with logit, now I want to know what are the other factors that has this relation with logit. Since I created multiple columns based on the ones that are already there in the language_data_analysis data set, I wanted to first create a GG pairs plot, to see whether there is a correlation between those two, so that I can delete the ones with high correlation to make it ready for the multiple regression model.

```
#based on above plot, I only selected corr :
language_data_analysis %>%
  select(english_start, age, english_country_years, Proportion_YearsLived_EnglishCountry) %>%
  ggpairs()
```



Based on the analysis so far, for the multiple regression, the following data is selected: education + gender + age + Proportion_YearsLived_EnglishCountry + english_start_categorical

```
#begu motanaseb ba WWWW, and I know my data, inaro negah dashtam va vase hamin chiziam kam nasho
        d bade stepwise regression.


lm_full <- lm(logit ~ education + gender + age + Proportion_YearsLived_EnglishCountry
            + english_start_categorical, data = language_data_analysis)

summary(lm_full)
```

```
##
## Call:
## lm(formula = logit ~ education + gender + age + Proportion_YearsLived_EnglishCountry +
##     english_start_categorical, data = language_data_analysis)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1349 -0.6020 -0.0538  0.5115  3.4677
##
## Coefficients:
##                                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                1.2827980  0.0503701  25.467  < 2e-16
## educationhigh school degree               -0.0739895  0.0047455 -15.591  < 2e-16
## educationincomplete highschool            -0.2237434  0.0082623 -27.080  < 2e-16
## educationsome graduate                    -0.0868706  0.0055202 -15.737  < 2e-16
## educationsome undergraduate                0.0282368  0.0053596   5.268 1.38e-07
## educationundergraduate degree              0.0474204  0.0039556  11.988  < 2e-16
## gendermale                                -0.1718650  0.0029642 -57.980  < 2e-16
## age                                        0.0076892  0.0001607  47.847  < 2e-16
## Proportion_YearsLived_EnglishCountry       0.7641245  0.0061425 124.400  < 2e-16
## english_start_categoricalFirst Decade      1.2983131  0.0497619  26.091  < 2e-16
## english_start_categoricalFourth Decade     0.1925719  0.0548581   3.510 0.000448
## english_start_categoricalSecond Decade     1.0759587  0.0497333  21.635  < 2e-16
## english_start_categoricalSeventh Decade -0.2446534  0.2016236  -1.213 0.224971
## english_start_categoricalSixth Decade     -0.1003497  0.1069783  -0.938 0.348226
## english_start_categoricalThird Decade      0.4660398  0.0508119   9.172  < 2e-16
##
## (Intercept)                               ***
## educationhigh school degree               ***
## educationincomplete highschool            ***
## educationsome graduate                    ***
## educationsome undergraduate               ***
## educationundergraduate degree             ***
## gendermale                                ***
## age                                       ***
## Proportion_YearsLived_EnglishCountry      ***
## english_start_categoricalFirst Decade     ***
## english_start_categoricalFourth Decade    ***
## english_start_categoricalSecond Decade    ***
## english_start_categoricalSeventh Decade
## english_start_categoricalSixth Decade
## english_start_categoricalThird Decade     ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.874 on 350425 degrees of freedom
## Multiple R-squared:  0.1125, Adjusted R-squared:  0.1125
## F-statistic:  3174 on 14 and 350425 DF,  p-value: < 2.2e-16
```

some of the inferences are for example men scored 0.17 less in English proficiency test compared to women. In terms of English start categories, the reference level is 5th decade, so orders should be compared to the 5th decade for example, people who started English in their first decade, scored 1.2983131 higher in English proficiency test compared to people who started learning English in their 5th decade. also people who started learning English in their 6th decade, scored 0.1003 list in English proficiency test compared to 5th decade.

also we see that in previous linear model the intercept was 2.573996, but now the intercept is 1.2827980 which shows the effect of added factors and the colinearity.

overal the formula woud be:

Logit = 1.2827980 + (-0.0739895 * high school degree) + (-0.2237434* complete highschool) + (-0.0868706* some graduate) + (0.0282368* some undergraduate) + (0.0474204* undergraduate degree) + (-0.1718650* gendermale) + (0.0076892* age) + …. For example if in future predictions the gender is female, in the above formula for the gender male we will put 0, if it is male we will put 1.

```
lm_step <- step(lm_full)
```

```
## Start:  AIC=-94397.8
## logit ~ education + gender + age + Proportion_YearsLived_EnglishCountry +
##     english_start_categorical
##
##                                        Df Sum of Sq    RSS    AIC
## <none>                                               267665 -94398
## - education                             5    1259.0 268924 -92763
## - age                                   1    1748.7 269413 -92118
## - gender                                1    2567.7 270232 -91054
## - english_start_categorical             6    8791.5 276456 -83084
## - Proportion_YearsLived_EnglishCountry  1   11820.5 279485 -79256
```

Backward Stepwise regression is also done here to remove any factor that is not that relevant or important in predicting the logit, however after the end results it is obviously shown that the remaining factors are all important.

Conclusion

Finally we can conclude that in terms of the three main questions, people who started learning English in early stages of his or her life has better proficiency in English. proficiency in English is also higher among the people who have had lived in an English speaking country. moreover people who has spent a larger proportion of his or her life in an English speaking country has better proficiency in it. also proficiency in English is related to other factors such as the education level, age, and gender.