Data Analysis Walkthrough

Advanced Topics In Linguistics Project - Data Analysis

This document is a walkthrough for analyzing data for the final project. Your data should now be organized in one .csv file with all of the data and all of the relevant information labeled in columns. Each of your setups will be slightly different, so you may need to do some configuring to get this walkthrough to fit your data. This walkthrough will be for R; if you are familiar with RStudio, you may certainly use that. If you're familiar with another stats program other than R entirely (e.g., jamovi, JASP), then by all means use what you're familiar with.

This specific walkthrough is written for the pre-packaged experiment type, but the analysis you'll have to do for your own experiment is likely very similar. You'll have to figure out the relevant conditions and comparisons, and then all else should be the same.

This walkthrough will tell you everything you have to do, but not necessarily how to do it - you'll still need to figure some things out on your own ;)

Before we get started, we may need to do some data cleaning. Maybe some of your participants zoned out on a trial here or there, or had to run to the bathroom in the middle of the experiment. We don't want that data affecting our results - these trials are considered *outliers* and we can safely discard them. So before you go further, take a look at your data, sorted by response time. If you have any trials for which the response time is greater than 10 seconds, delete away - no need to save them at all, feel free to completely delete that row and pretend it never existed! There are much more scientific approaches to determining cutoffs for outlier detection, but this very conservative approach will suffice for here. If you have a lot of trials over 10 seconds, let me know…

We'll talk about graphs first, so we won't need R just yet - graphing will be easier in Excel or Google Sheets (though you can use whatever you want).

1. Words vs. Nonwords

For this first stage, we will address a simple Word vs. Nonword comparison while ignoring the AoA and Frequency variables.

1a. Making graphs

First up, we'll make two graphs to display our accuracy and response time data. There are countless ways to do this, including using Excel/Google Sheets, base R, and the ggplot package in R. If you aren't familiar with R or ggplot, using Excel/Sheets will be the easiest way! If you'd prefer R/ggplot, there are plenty of tutorials available online.

Response Accuracy Here's what your Response Accuracy bar graph will need to include:

- Two columns - one for Word, one for Nonword
- The response accuracy as a proportion (or percentage) with a 0 to 1 (or 0 to 100) scale
- Labels where appropriate
- Error bars showing standard error (optional because Google Sheets makes this annoying)

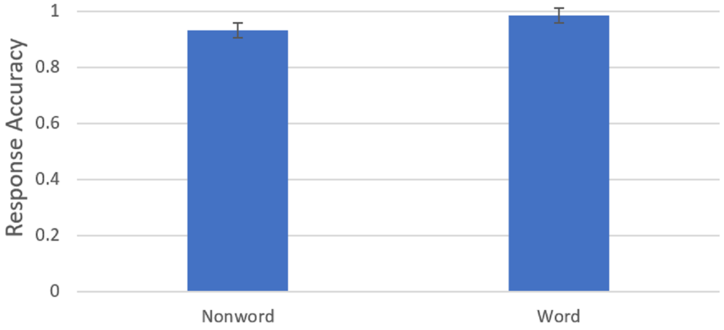


Figure 1: Example from Excel

Response Time For our response time data, we'll need a different kind of graph - specifically a boxplot. Here's what your Response Time boxplot will need to include:

- Two columns - one for Word, one for Nonword
- Response Time in milliseconds on the Y axis
- Labels where appropriate
- Median (as a line) and Mean (as a symbol) for each column
- Boxes that extend to 1st and 3rd quartile
- Whiskers that extend to 5th and 95th percentiles
- Points outside of the 5th and 95th percentiles displayed as outliers

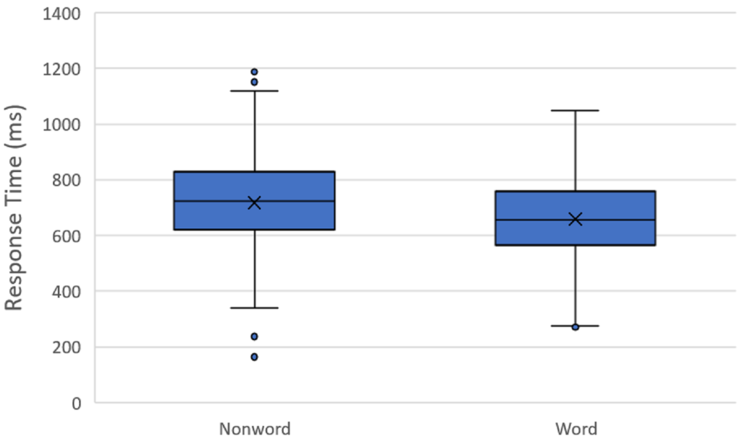


Figure 2: Example from Excel

Google Sheets doesn't have an easily-usable boxplot function, but it can be done. Here's a tutorial

1b. Statistical Analysis

The first thing you'll need to do is load your data. The following code will bring up a file explorer to let you select your data .csv:

```
data <- read.csv(file.choose(),header = TRUE)
```

If you want to check it out:

```
View(data)
```

Response Accuracy In the paper, you'll first need to provide the *descriptive statistics*. For reponse accuracy, this will be the proportion of correct response. It should look something like this:

```
Overall, correct responses were given for 93.1% of nonwords and 98.4% of words.
```

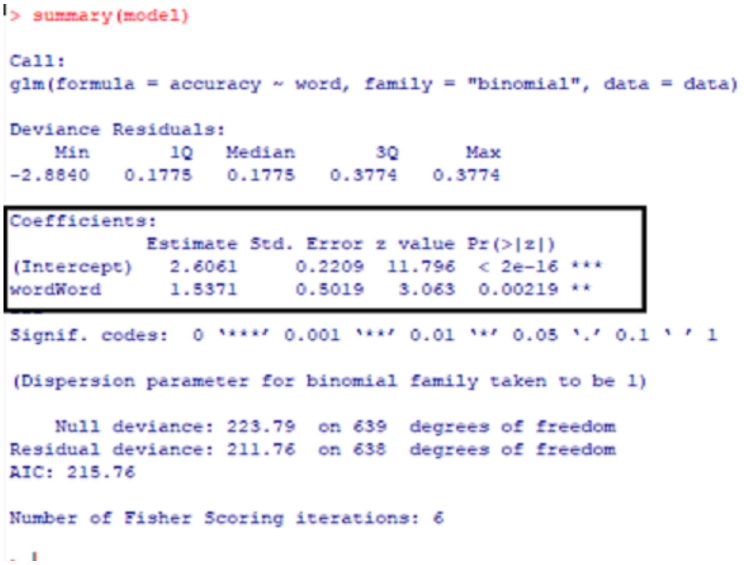
Next, we move on to inferential statistics. For our response accuracy model, we'll need to run a *logistic regression*. What we're doing here is modeling a dependent variable with a categorical distribution (here, either TRUE or FALSE). What we want here is to see whether that response is significantly affected by whether the item is a Word or a Nonword. For this to work properly, your "Accuracy" column will need to be in binary format - e.g., TRUE/FALSE, 1/0, something like that.

```
accuracy_model <- glm(accuracy ~ word, family = "binomial", data = data)
summary(accuracy_model)
```

In this code, we have the following variable elements:

- Name of the object you're creating ("accuracy_model" here, but you can call it whatever)
- Column in your data that has your binary response accuracy data ("accuracy" here)
- Column in your data that has your Word/Nonword categorization ("word" here)

When you run summary(accuracy_model), it will give you the results you need to report. You'll see two rows in the middle - (Intercept) and a second row right below it, the name of which will depend on how your data is set up. The "Coefficients" section is what you'll need here.



```
> summary(model)

Call:
glm(formula = accuracy ~ word, family = "binomial", data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8840   0.1775   0.1775   0.3774   0.3774

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   2.6061     0.2209  11.796  < 2e-16 ***
wordWord      1.5371     0.5019   3.063  0.00219 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 223.79  on 639  degrees of freedom
Residual deviance: 211.76  on 638  degrees of freedom
AIC: 215.76

Number of Fisher Scoring iterations: 6
```

Figure 3: Summary Output

What this second row shows you is the relevant comparison we're looking for. The way to interpret this depends on how your data is set up. For mine - the second row is wordWord. This means, in my setup, Value: "Word" in Category:"word". So that row gives me information about what happens when it's a real word, as compared to a nonword. If your data is set up in a different way, it may be the reverse. The farthest right column shows p-value (here shown as Pr(>|z|)), with asterisks marking significance level. The p-value above is 0.002, which is considered significant. The way to report the results would be as follows:

```
There was a significant effect of word status on response accuracy (z = 3.06, p = .002); responses were
```

Response Time Again, start with descriptive stats. Here, we need to provide the average response time and the standard deviation. Again, these are calculable in Excel/Sheets/R

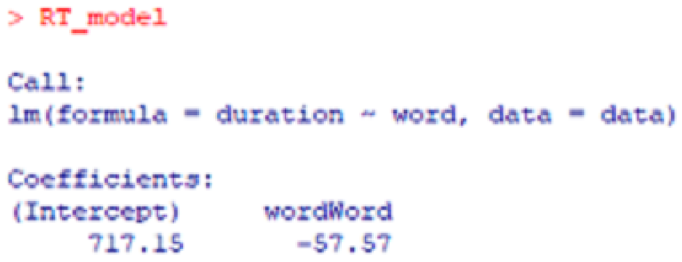
```
The average response time to nonwords was 711.49ms (sd = 167.07), compared to 655.80 (sd = 146.49) for w
```

For inferential statistics, we'll now need a linear regression, since our response time variable is linear! (Note: this is the same as a one-way ANOVA.) We want to see whether the response time is dependent on the word status (word or nonword).

```
RT_model <- lm(duration ~ word, data = data)
RT_model
summary(RT_model)
```

Again, we have, from left to right: object name, name of response time column from your .csv, name of Word/Nonword column from your .csv.

If you call "RT_model", you'll get this:

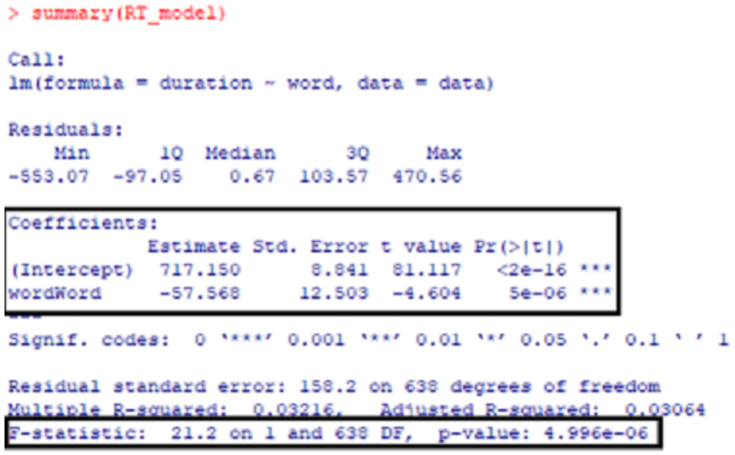


```
> RT_model

Call:
lm(formula = duration ~ word, data = data)

Coefficients:
(Intercept)     wordWord  
     717.15       -57.57  
```

Figure 4: RT_model

The way to interpret this is: according to our model, words are predicted to be -57.57ms compared to nonwords (that is, 57ms faster). If we check against our descriptive statistics, that looks pretty close! A sign that our model is correctly run.

The summary will look like this:



```
> summary(RT_model)

Call:
lm(formula = duration ~ word, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-553.07  -97.05    0.67  103.57  470.56 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  717.150      8.841  81.117   <2e-16 ***
wordWord     -57.568     12.503  -4.604    5e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 158.2 on 638 degrees of freedom
Multiple R-squared:  0.03216,	Adjusted R-squared:  0.03064 
F-statistic:  21.2 on 1 and 638 DF,  p-value: 4.996e-06
```

Figure 5: summary(RT_model)

Since it's a different kind of model, we have a bit more to report. We'll here report the F-statistic, degrees of freedom, and p-value - all of which can be found by calling the "summary(RT_model)" function above. You'll note that in the writeup below, I've just listed p < .001. If it's less than .001, you don't need to be precise - just say p < .001.

```
There was a significant effect of word status on response time (F(1,638) = 21.2, p < .001); words were 
```

2. AoA and Frequency

For this analysis, we'll group words and nonwords together and focus on the effects of AoA and Frequency. Remember we made a 2x2 setup - there are two levels of AoA (Early, Late) and two levels of Frequency (High, Low). Because of this, we'll need to test if there's an interaction. But first, visualization!

2a. Making graphs

Response Accuracy Here, repeat the same steps as you did for Response Accuracy in 1a. You'll notice it's a little more complex, since we'll need 4 columns instead of 2! (Hint: The easiest way to set up this data for graphing will be to merge your AoA and Frequency Columns, like with Excel's =CONCAT() function.)

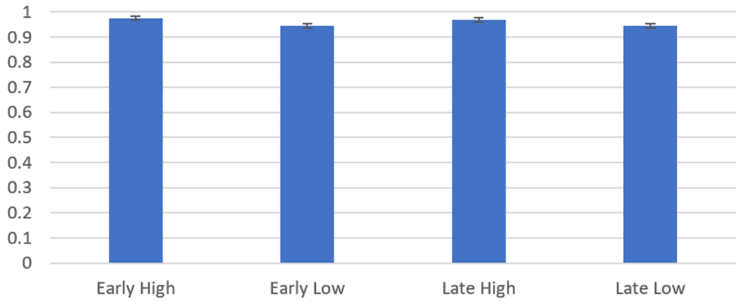


Figure 6: Graph from Excel

Response Time Same thing for RT data!

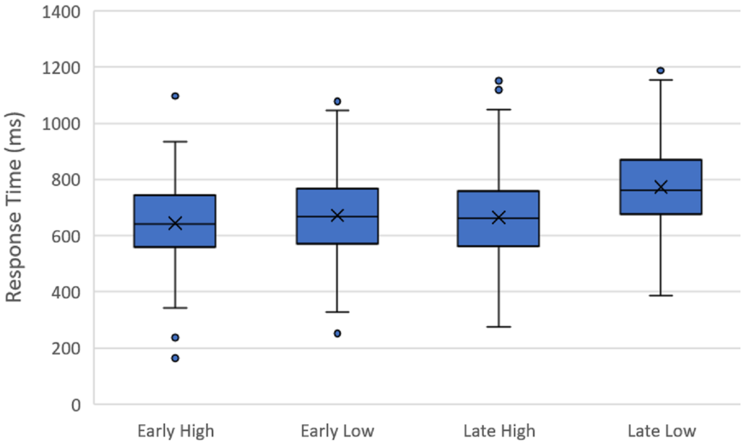


Figure 7: Graph from Excel

2b. Statistical Analysis

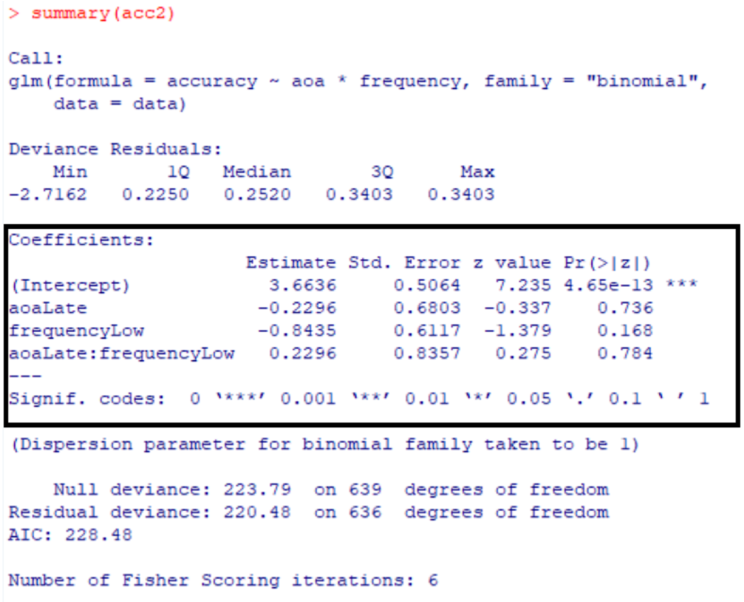
First, descriptive statistics. Provide response accuracies for all four conditions.

We can use the same principles and models as before, but these are slightly more complex because they'll now involve an interaction.

Response Accuracy We'll set up our logistic regression model in the exact same way as before; the only difference is we now have two independent variables to include and we'll want to test their interaction.

```
accuracy_model_2 <- glm(accuracy ~ aoa*frequency, family = "binomial", data = data)
summary(accuracy_model_2)
```

This is the same code before, except we've now swapped in our "aoa" and "frequency" columns, joined by "*" to include the interaction.



```
> summary(acc2)

Call:
glm(formula = accuracy ~ aoa * frequency, family = "binomial",
    data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7162   0.2250   0.2520   0.3403   0.3403

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)            3.6636     0.5064   7.235 4.65e-13 ***
aoaLate               -0.2296     0.6803  -0.337    0.736
frequencyLow          -0.8435     0.6117  -1.379    0.168
aoaLate:frequencyLow   0.2296     0.8357   0.275    0.784
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 223.79  on 639  degrees of freedom
Residual deviance: 220.48  on 636  degrees of freedom
AIC: 228.48

Number of Fisher Scoring iterations: 6
```

Figure 8: summary(accuracy_model_2)

The boxed portion above shows you the relevant statistics for aoa, frequency, and then the interaction as the bottom line. As you can see here, there are no significant effects! We still have to report the results, but we'll say there was no significant effect.

```
For response accuracy, there was no significant effect of Age of Acquisition (z = -0.34, p = 0.74), Fre
```

Response Time Again, descriptive statistics first. Provide average response times and standard deviations for each of the four conditions.

Also again, our model will look like what we've done before!

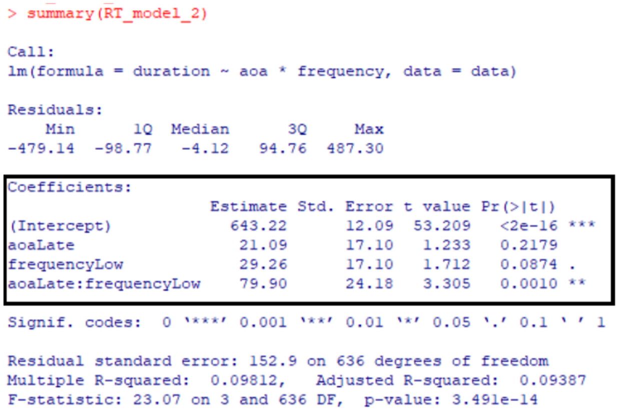
```
RT_model_2 <- lm(duration ~ aoa*frequency, data = data)
summary(RT_model_2)
anova(RT_model_2)
```

This is the same code before, except we've now swapped in our "aoa" and "frequency" columns, joined by "*" to include the interaction.

Hey look, there's a significant interaction! We'll have to report it as such. Since there is an interaction, we don't necessarily need to report the main effects of AoA and Frequency themselves. Another view, from the anova(RT_model_2) function, is useful here:

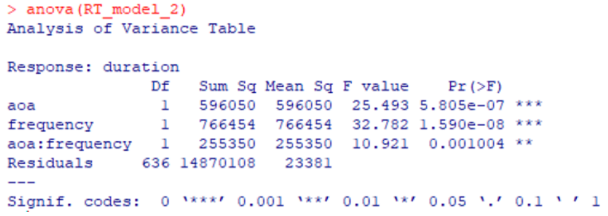
```
There was a significant interaction between Age of Acquisition and Frequency (F(1, 636) = 10.92, p = .0
```

Again, use "RT_model_2" to make sure your model aligns with your descriptive statistics. This can also help you interpret the direction of your effects.



```
> summary(RT_model_2)

Call:
lm(formula = duration ~ aoa * frequency, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-479.14  -98.77   -4.12   94.76  487.30 

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)            643.22      12.09  53.209   <2e-16 ***
aoaLate                 21.09      17.10   1.233   0.2179    
frequencyLow            29.26      17.10   1.712   0.0874 .  
aoaLate:frequencyLow    79.90      24.18   3.305   0.0010 ** 
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 152.9 on 636 degrees of freedom
Multiple R-squared:  0.09812,	Adjusted R-squared:  0.09387 
F-statistic: 23.07 on 3 and 636 DF,  p-value: 3.491e-14
```

Figure 9: summary(RT_model_2)



```
> anova(RT_model_2)
Analysis of Variance Table

Response: duration
               Df   Sum Sq Mean Sq F value    Pr(>F)    
aoa             1   596050  596050  25.493 5.805e-07 ***
frequency       1   766454  766454  32.782 1.590e-08 ***
aoa:frequency   1   255350  255350  10.921  0.001004 ** 
Residuals     636 14870108   23381                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 10: anova(RT_model_2)

3. Closing Notes and Tips

- Make sure your graphs are clear! Provide labels on and captions under every graph so someone reading along knows exactly what's being shown in each graph.
- Provide descriptive statistics first (mean, standard deviation) followed by the inferential statistics.
- When you write up your inferential statistics, specify what model was used and what the variables are.
- Be sure to describe every result in words as well as numbers. If you find, for example a significant effect of Frequency on Response Time, provide the relevant statistics *and* describe the direction of the effect (i.e., "High frequency words were responded to more quickly than low frequency words").
- Save your code and graphs! Don't accidentally lose stuff and have to start over again.
- Remember, the point of this project is not to get the "right" results. Even if you have no significant effects at all (honestly, with our sample sizes, this is probably likely), still write up everything in detail. The actual results from your experiment

4. Extra Credit

I will offer an extra credit opportunity for this project involving data analysis. In addition to the analysis described above, figure out how to run a linear mixed effects model on the data as well, with random effects for "item" (the word itself) and "participant" (the individual participant in the experiment). You'll need the "lme4" package in R for this. I can help by making sure the goal and the desired analyses are clear, but you'll be on your own for figuring out how to do it. Plenty of tutorials online ;)