Data Science Language Analysis

Final Report

Avinash Prabhakaran, Nazli Ozum Kafaee, Prash Medirattaa

2018-04-22

Contents

1 Introduction

It is common to hear from people working with data that they have a clear choice between R and Python. Almost everyone who has worked with both R and Python have one or the other as their favorite. We were curious if there might be a specific reason underlying such choice. After some brainstroming, we came up with the hypothesis that a person's choice between R and Python might be due to their preference in a specific data science task such as data visualization, data wrangling and machine learning. This hypothesis was based solely on observations and personal experience, but we set the goal to explore such causal relationship (if there exists one) with data in our data analysis project.

2 Methodology

2.1 Data collection

Our primary data source was an online survey created on Google Forms and distributed to the MDS cohort, faculty and teaching assistants through Slack channels as well as to people in the authors' Linkedin and Whatsapp network. In the end, we managed to collect 85 responses.

One of the concerns we had to address was regarding the storage of the collected data. Since we used Google Forms for our survey, the data was hosted in the US. Assuming that a great majority of our respondents were Canadian residents, we made sure to inform them of the fact that the data being collected is hosted in

the US and to get their consent before proceeding further in the survey. More information on this matter can be found in the UBC Office of Research Ethics - Using Online Surveys document.

2.2 Study Design

In our survey, we wanted repondents to answer two main questions:

- Which of the following programming languages do you prefer more?
  Possible answers: "R" / "Python"
- What is your favorite data science task?
  Possible answers: "Data wrangling" / "Data visualization" / "Machine Learning"

In the former, respondents were required to choose one of "R" or "Python" and in the latter, they could choose one of three options which were "Data wrangling", "Data visualization" and "Machine Learning". The answers to these two questions would provide us the information for the dependent and independent variables in our analysis, respectively.

In order to fully discover the causal relationship between task preference and language preference, we also collected data about factors that could have an effect on both of our dependent and independent variable. The primary goal in collecting information on possible confounding variables was to ensure that we can control for these in our analysis later on. We determined five possible confounding variables for which we asked the following questions:

- What is your academic background?
  Possible answers: "Computer Science/Computer Engineering" / "Mathematics/Statistics" / "Other"
- How many years of coding experience do you have prior to using Python/R?
  Possible answers: "Less than 1" / "1 to 5" / "More than 5"
- Do you enjoy/love coding?
  Possible answers: "Yes" / "No" / "Indifferent"
- Which programming language did you learn first?
  Possible answers: "Python" / "R" / "SAS" / "Matlab" / "C" / "Java" / "Other"
- How many programming languages do you use actively?
  Possible answers: "1" / "2" / "3" / "4" / "5 or more"

We thought that academic background would be a confounding variable as people with Computer Science/Computer Engineering background would have been introduced to Python as part of their degree and people from Mathematics/Statistics degrees would have been introduced to R in general. However, we did not anticipate any bias towards R or Python by graduates of any other degrees. We also believed that the amount of coding experience could be a confounder as it can be indicate how open the user is in selecting a statistical programming language over a general-purpose programming language. However, we also realized that it is possible that a user can become highly opinionated when they have greater experience, and they might prefer Python. Therefore, we wanted to include this variable in our survey as it would be interesting to analyze. Another variable we wanted to collect information on was the user's attitude towards coding. The outlook towards coding could be a confounder as Python is a general-purpose programming language and it can be used in various areas, and its application is not limited to Data Science/Statistics whereas R is a statistical programming language and is mainly used only in the fields of Data Science and Statistics. Again, a person's first programming language would be very influential as it dictates their style of coding and would also be a deciding factor in what they seek for in other languages. Some of the programming languages are more closely related to Python whereas some others are more related to R. The number of programming languages a person actively uses could be a deciding factor too as it can dictate how comfortable the user is in using different syntaxes and will also be indicative of how flexible the user.

Our survey can be accessed fully here.

2.3 Analysis Methods

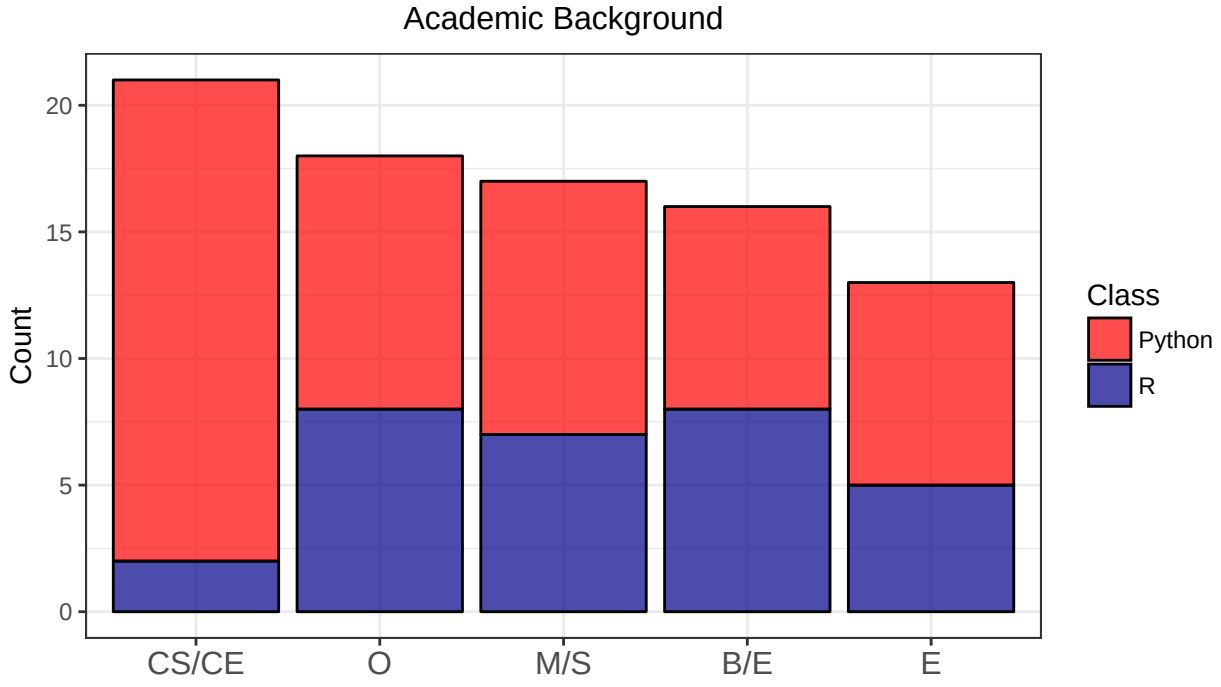
The data collected as a result of our survey was downloaded as a `csv` file and imported into R for analysis. All data wrangling and visualization were done in the R computing environment. The code chunks that download data, apply wrangling and create plots can be found in read_data.R, clean_data.R and get_plots.R, respectively.

3 Exploratory Data Analysis

3.1 Wrangling

The data collected from the survey required some initial wrangling in order to be prepared for exploratory and statistical data analysis. The main goal in wrangling was to organize answers that came from the "Other" answer option which enabled respondents to freely type their answer for a specific question if their answer did not correspond to any of the answer options provided.

The first question in our survey was "What is your academic background?". This question had three main options: "Computer Science / Computer Engineering", "Mathematics / Statistics" and "Other". We saw that "Other" comprised a lot of different answers and made the second highest in terms of share. We decided to split "Other" category and create new categories as we saw that there were some major categories that appeared repeatedly but were typed differently. For example, the answers in the form of "business", "Business", "business and economics", etc. all pointed out to the same category but appeared as distinct categories in the raw data. In fact, two such categories we observed were engineering and business studies. Therefore, we added "Engineering" and "Business/Economics" as new categories to the academic background variable and left the rest to "Other". Our final categorization of the academic background variable can be seen below together with the distribution of preference between R and Python in each category.

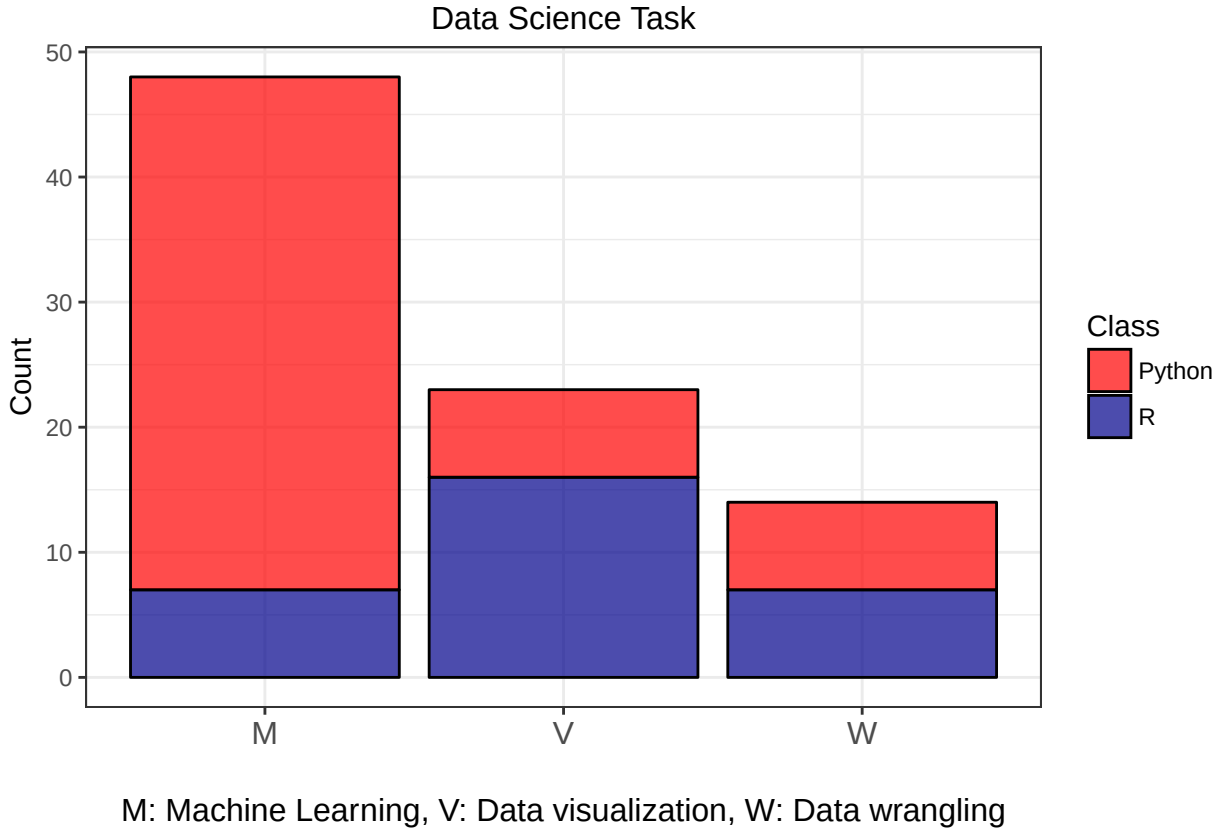


CS/CE: Computer Science/Computer Engineering, O: Other,
M/S: Mathematics/Statistics, B/E: Business/Economics,
E: Engineering

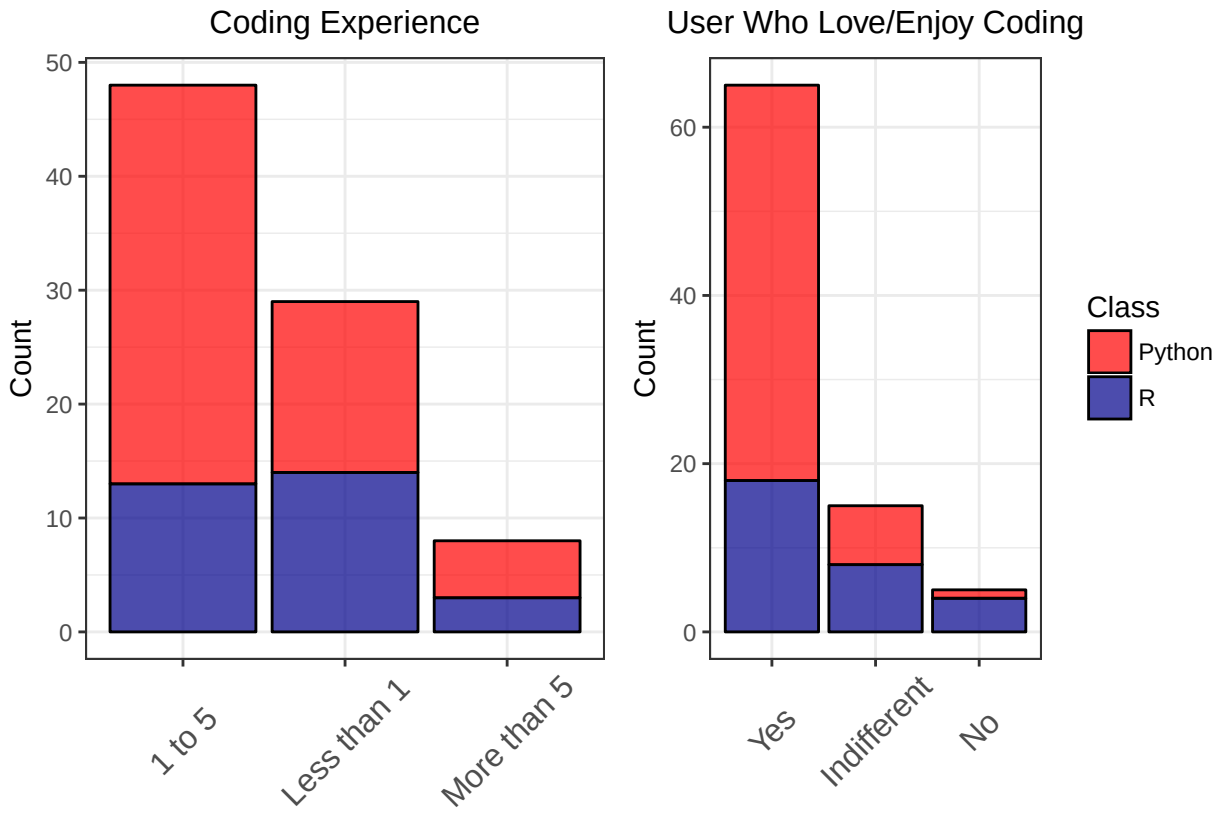
We were faced with a similar issue in the answers to the question which asked respondents about the first programming language they learned. Again, we gave respondents six main options to choose from and an "Other" option to fill in if necessary. They could freely type the name of their first programming language if it was not one of the predetermined languages listed. We observed that the "Other" option comprised of varying languages but each answer held one or two people and none of the languages that we had not listed represented a major group of people. Therefore, we aggregated all answers in "Other" and kept them together.

3.2 Visualizations

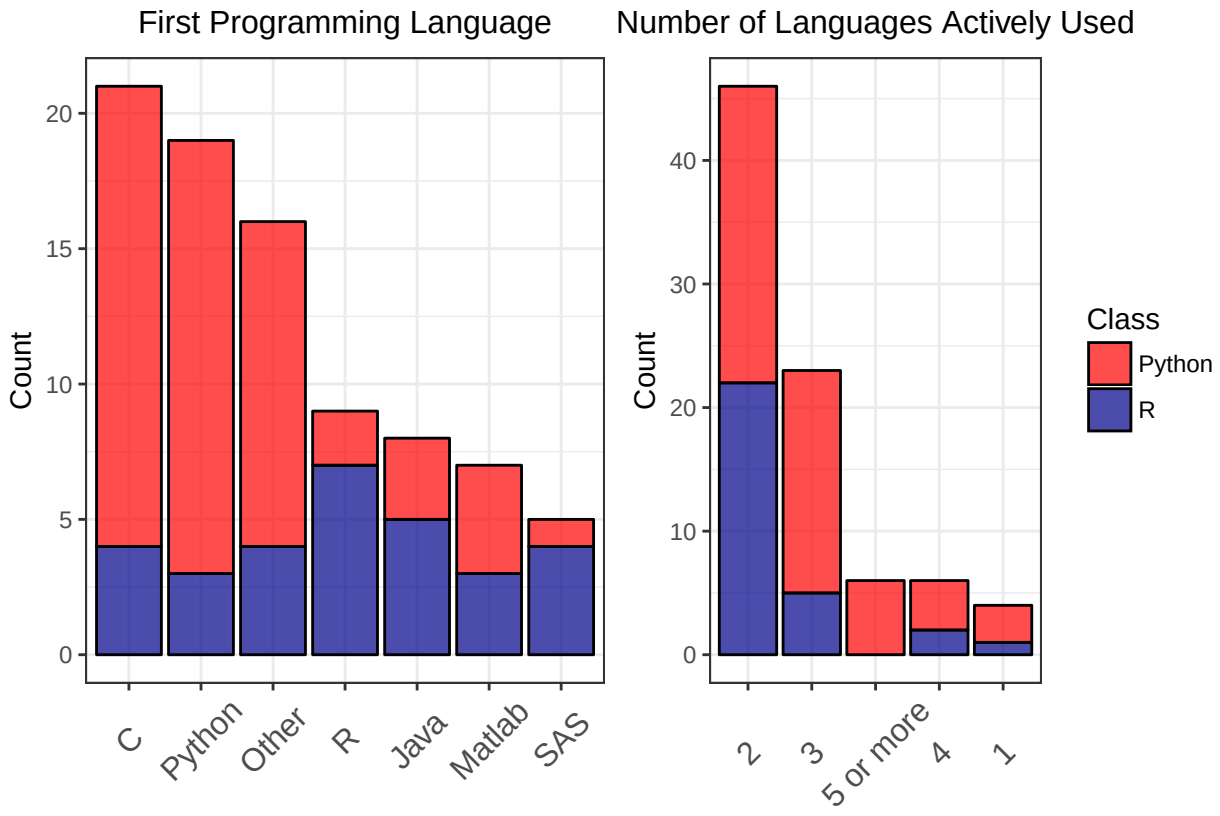
Our primary goal was to see how the preference between R and Python changed depending on a person's favorite data science task. In the plot below, we can see that Python is more popular among people whose preferred data science task is machine learning. However, R seems to be the preferred language when it comes to data visualization. In the data wrangling category, there is an equal split between R and Python.



We had mentioned earlier that we should not forget about confounders and just look at the relationship between our dependent and independent variables without taking them into account. These confounders might be the real reason for the proportional difference we observe in the plot above. Therefore, concluding a causal inference without a careful consideration of the possible confounders would be naive. We made use of stacked bar plots in order to see the difference in the preference between R and Python depending on each category in our possible confounding variables. As can be seen in the plots below, we have observed that the proportion between the two languages changes depending on a person's experience in coding. Novice coders (`Less than 1`) prefer Python around 50% of the time whereas this proportion increases to around 60% for Intermediate (`1 to 5`) and experienced coders (`More than 5`). We can observe a proportional difference but this difference does not seem to be significant. Also, we should keep in mind that the number of respondents in each category are not equal and these proportions might have been similar or more different if we could have collected more data equally in each category.



We have observed that the first programming language has some effect on the choice between R and Python. People who have learned a statistical programming language (`R`, `Matlab`, `SAS`) as their first language seem to be leaning more towards R whereas this is the reverse for those groups that have learned a general-purpose programming language (`C`, `Python`) as their first language.



4 Statistical Analysis

Our dependent variable has two categories as `R` and `Python`, so it is a binary variable (Python is coded as 1) and our independent variable is categorical with three categories. Therefore, we found it appropriate to use a `glm` model with logit link function as logistic regression is useful with binary random component and mixed systematic components.

With the type of model specified, we tried fitting the model using different variables and assessed all of them before deciding on one. The models we have explored and their AIC scores can be seen below and further results can be found in the analysis in this repository.

```
kable(bind_cols(Sno =c("1","2","3","4","5","6","7"),Models = c("preference ~ task", "preference ~ task +
```

Sno	Models	AIC
1	preference ~ task	93.555
2	preference ~ task + background + experience + attitude + first + active	90.394
3	preference ~ task + background + experience + first + active	88.420
4	preference ~ task + background + first + active	86.694
5	preference ~ task + background + first	90.428
6	preference ~ task + background + active	90.411
7	preference ~ task + first + active	89.657

```
#Fitting a GLM without any confounding variables.
model.1 <- glm(preference ~ task, family = binomial(link = 'logit'), data = data)

#Fitting GLM with all the confounding variables.
```

```
model.2 <- glm(preference ~ task + background + experience + attitude + first + active,
               family = binomial(link = 'logit'), data = data)

#Final Model with only first language as confounder
model.3 <- glm(preference ~ task + first, family = binomial(link = 'logit'), data = data)

#Releveling the data to obtain the comparison between Machine Learning and Data Wrangling.
model.4 <- glm(preference ~ task + first, family = binomial(link = 'logit'),
               data = data_relevel)
```

term	estimate	std.error	statistic	p.value
(Intercept)	-0.8614963	2.6527822	-0.3247520	0.7453688
taskData visualization	-4.1142174	1.1776112	-3.4936976	0.0004764
taskData wrangling	-1.9905944	0.9898468	-2.0110125	0.0443241
backgroundComputer Sc/Eng	3.8720513	1.6799990	2.3047938	0.0211781
backgroundEngineering	-0.3728615	1.2799724	-0.2913043	0.7708186
backgroundMaths/Stats	0.4743432	1.0137671	0.4679016	0.6398550
backgroundOther	0.4333682	1.2456112	0.3479161	0.7279032
firstJava	-2.0700504	1.4489828	-1.4286232	0.1531126
firstMatlab	-0.6066711	1.5277983	-0.3970885	0.6913022
firstOther	0.6676056	1.2053382	0.5538741	0.5796650
firstPython	3.0142724	1.6048857	1.8781850	0.0603559
firstR	-0.9333257	1.4231468	-0.6558183	0.5119411
firstSAS	-0.6027816	1.7587936	-0.3427245	0.7318058
active2	1.4009353	2.1619754	0.6479886	0.5169924
active3	3.8921539	2.3830329	1.6332775	0.1024107
active4	2.7215844	2.6835120	1.0141875	0.3104933
active5 or more	20.2088781	2047.0819987	0.0098720	0.9921234

```
#Feature selection using stepwise
sw_model <- step(model.2, direction = "both", trace=FALSE)
```

preference ~ task + background + first + active

The complete analysis can be found in analysis.pdf in this repository. Some of the main models along with variables can be be found above with their respective `AIC` scores.

We did manual backward selection as we wanted to ensure the `task` variable to be present in the final model. Selection criteria used was `AIC` being `86.694`. Just as an extra step we did stepwise methodolgy to vaildate the results.

```
summary(model_level)
```

```
##
## Call:
## glm(formula = preference ~ task + background + first + active,
##     family = binomial(link = "logit"), data = data_relevel)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.71388  -0.41985   0.05763   0.36812   2.86063
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)                -0.8615     2.6528  -0.325 0.745369
## taskData visualization     -4.1142     1.1776  -3.494 0.000476 ***
## taskData wrangling         -1.9906     0.9898  -2.011 0.044324 *
## backgroundComputer Sc/Eng   3.8721     1.6800   2.305 0.021178 *
## backgroundEngineering      -0.3729     1.2800  -0.291 0.770819
## backgroundMaths/Stats       0.4743     1.0138   0.468 0.639855
## backgroundOther             0.4334     1.2456   0.348 0.727903
## firstJava                  -2.0701     1.4490  -1.429 0.153113
## firstMatlab                -0.6067     1.5278  -0.397 0.691302
## firstOther                  0.6676     1.2053   0.554 0.579665
## firstPython                 3.0143     1.6049   1.878 0.060356 .
## firstR                     -0.9333     1.4231  -0.656 0.511941
## firstSAS                   -0.6028     1.7588  -0.343 0.731806
## active2                     1.4009     2.1620   0.648 0.516992
## active3                     3.8922     2.3830   1.633 0.102411
## active4                     2.7216     2.6835   1.014 0.310493
## active5 or more            20.2089  2047.0820   0.010 0.992123
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 110.372  on 84  degrees of freedom
## Residual deviance:  52.694  on 68  degrees of freedom
## AIC: 86.694
##
## Number of Fisher Scoring iterations: 17
```

5 Results

comparison	reference	estimate	std.error	p.value	odds-ratio	lowerCI	upperCI
Data wrangling	Data visualization	2.123623	1.1196597	0.0578717	8.3613763	0.8907456	78.4877408
Machine Learning	Data visualization	4.114217	1.1776112	0.0004764	61.2042965	5.8065990	645.1222014
Data wrangling	Machine learning	-1.990594	0.9898468	0.0443241	0.1366142	0.0188680	0.9891585
# Final Model							

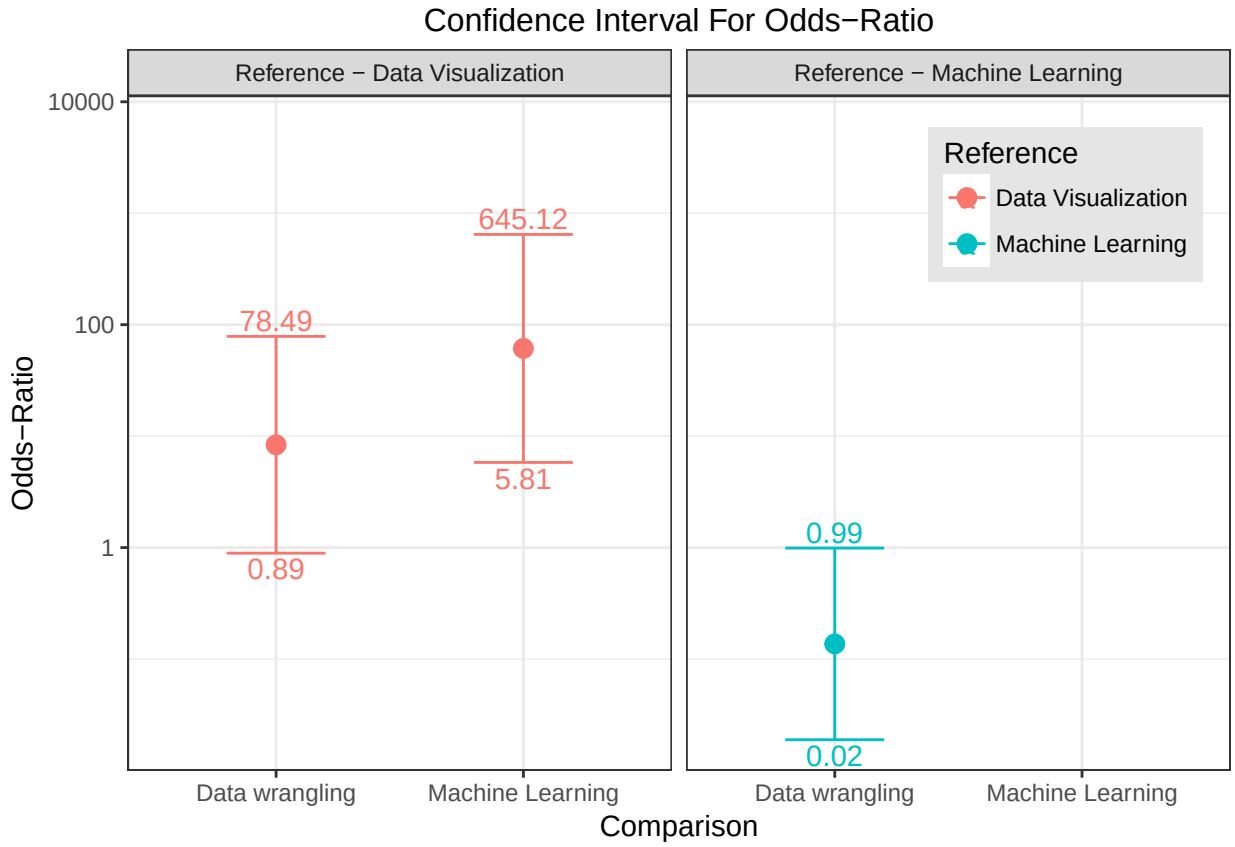
$logit(p) = log(\frac{p}{1-p}) = \beta_0 + \beta_1 * \text{task} + \beta_2 * \text{background} + \beta_3 * \text{first} + \beta_4 * \text{active}$

We can say these things about the above fitted model in terms of odds ratio.

- People whose favorite task is Data Wrangling are 8.3 times more likely to select Python as their favorite language compared to people whose favorite task is Data Visualization.
- People whose favorite task is Machine Learning are 61 times more likely to select Python as their favorite language compared to people whose favorite task is Data Visualization.
- People whose favourite task is Machine Learning are 5.8 times more likely to prefer Python as their favourite language compared to people whose favorite task is Data Wrangling

6 Assumptions

- In our study there are three confunding variable(Academic background,first programming language and number of languages used actively).
- No interactions was there between expalanatory variables and the confunding variables in the model.
- There is also no interaction between the confounding variables themselves.



7 Conclusion