# EDAV HW1 Class Survey

*Team Excel*

*February 11, 2016*

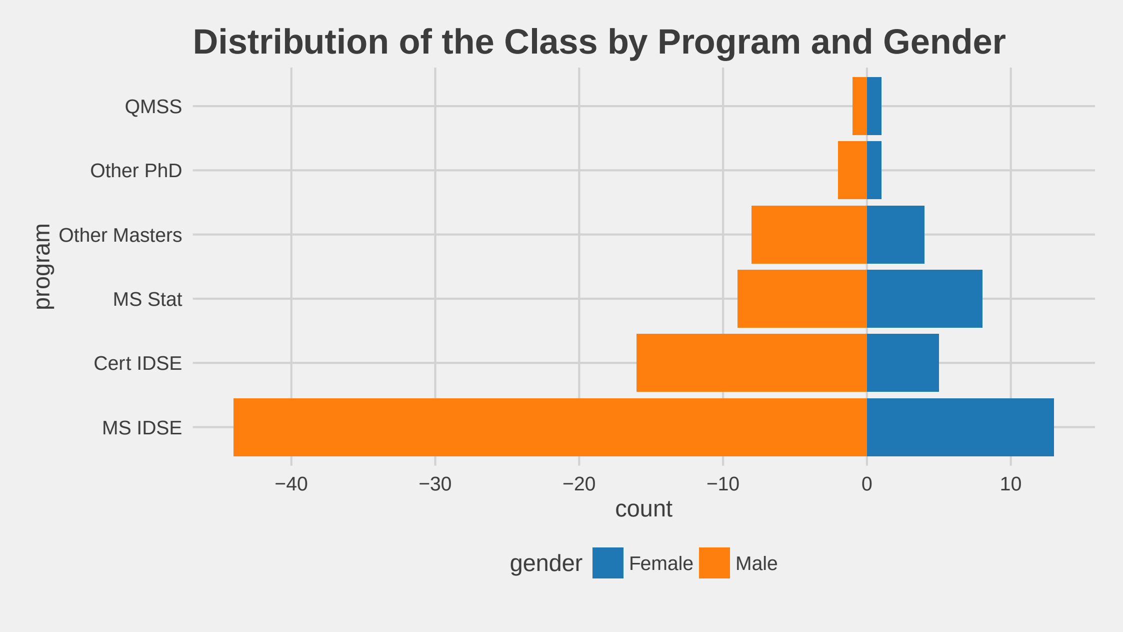
## 1 Introduction

A survey was conducted on the class which consisted of nine questions about students technical competency across a variety of data tools. This is team Excel's analysis of the 140 observations generated by the survey. We aim to provide insights and a better understanding of data through the use of data visualization. At the same time we will share some of the reasons of why we have done these respective analysis.

## 2 Respondent Information

In this section we will provide an overview of the respondents based on gender and program.

The butterfly bar plot below shows a comparison in the number of male and female students for each of the programs. Also it provides a good idea of the total counts of students from each program that participated in the survey.



The gender imbalance in our class (29% women, 71% men) is representative of a larger cultural issue. Below, we compare the gender diversity in our course with the tech workforces of several of the Internet giants, who publicly released diversity reports in 2015.

## 3 Tools

Now we will explore tools used by the students by looking at distributions, relations, and comparisons to external data.

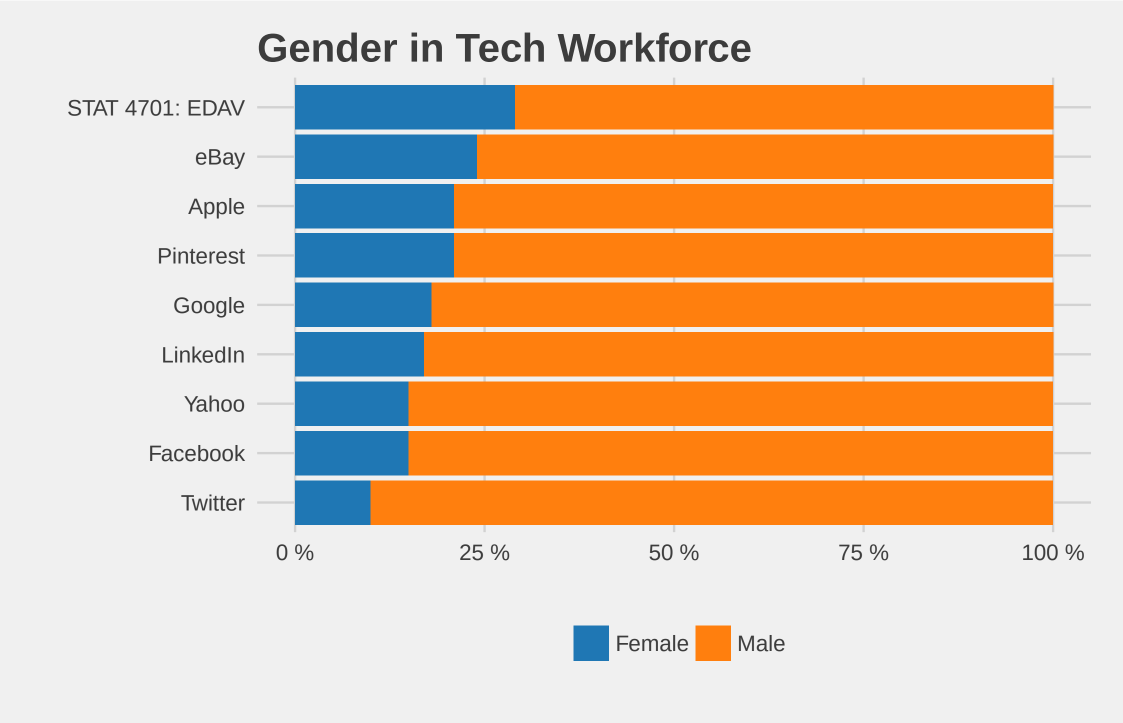
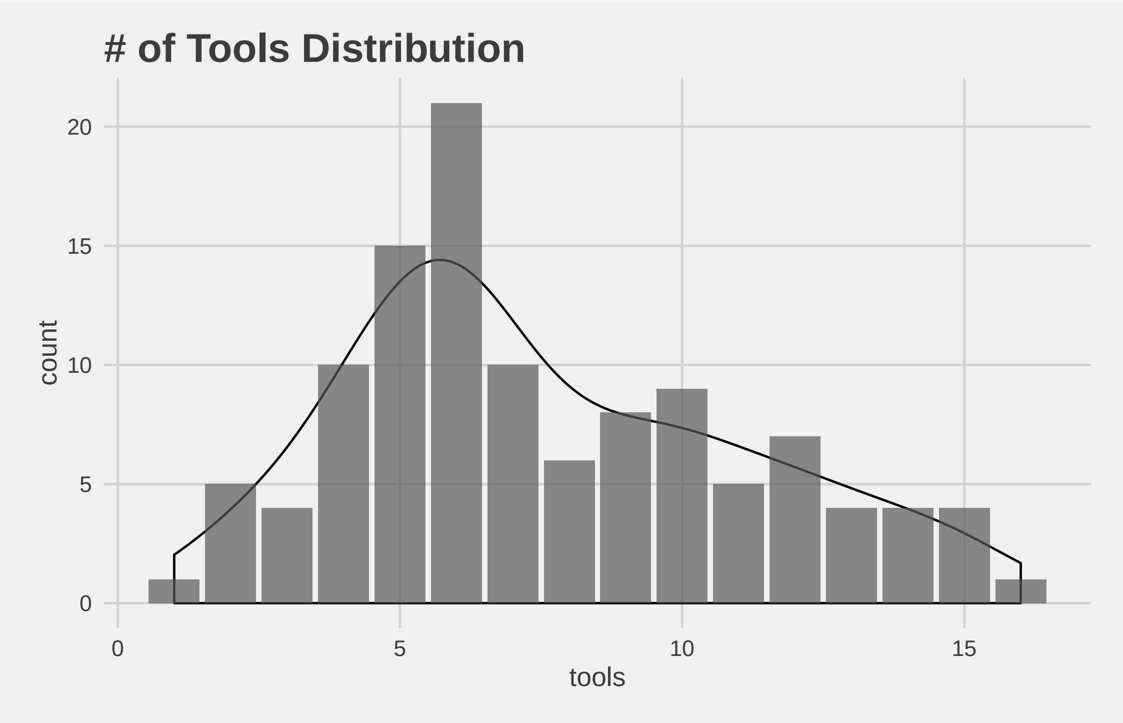
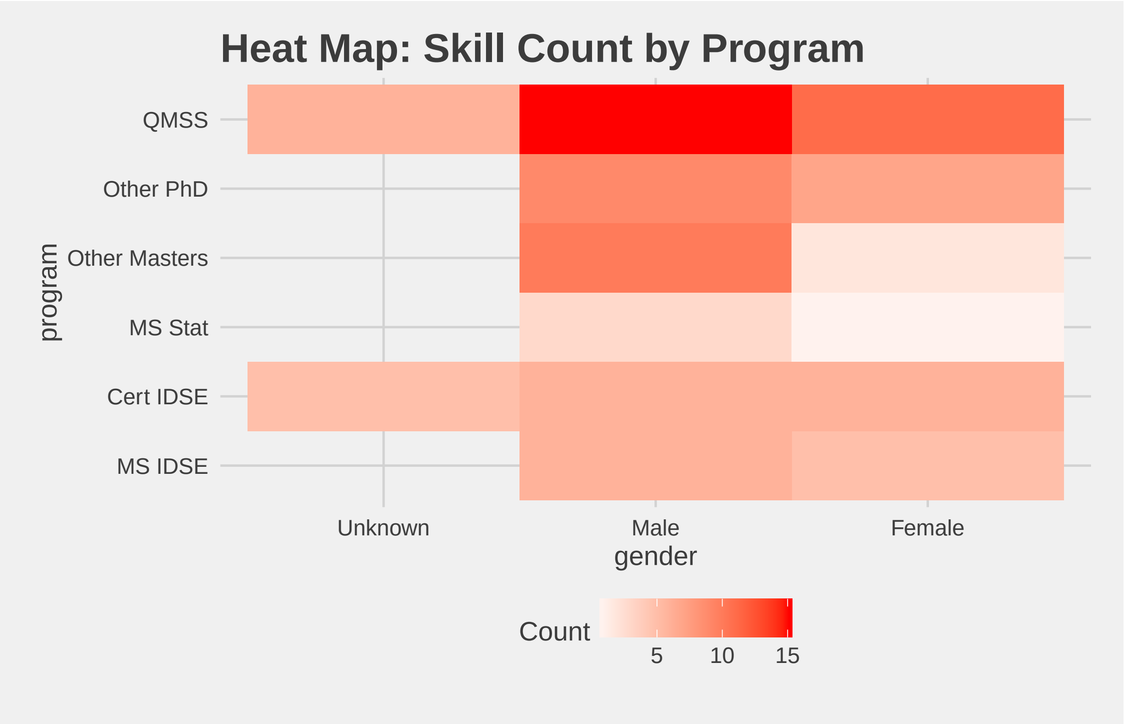


Figure 1: Data Sources: Apple, Facebook, Google, Twitter, LinkedIn, Pinterest, Yahoo, eBay

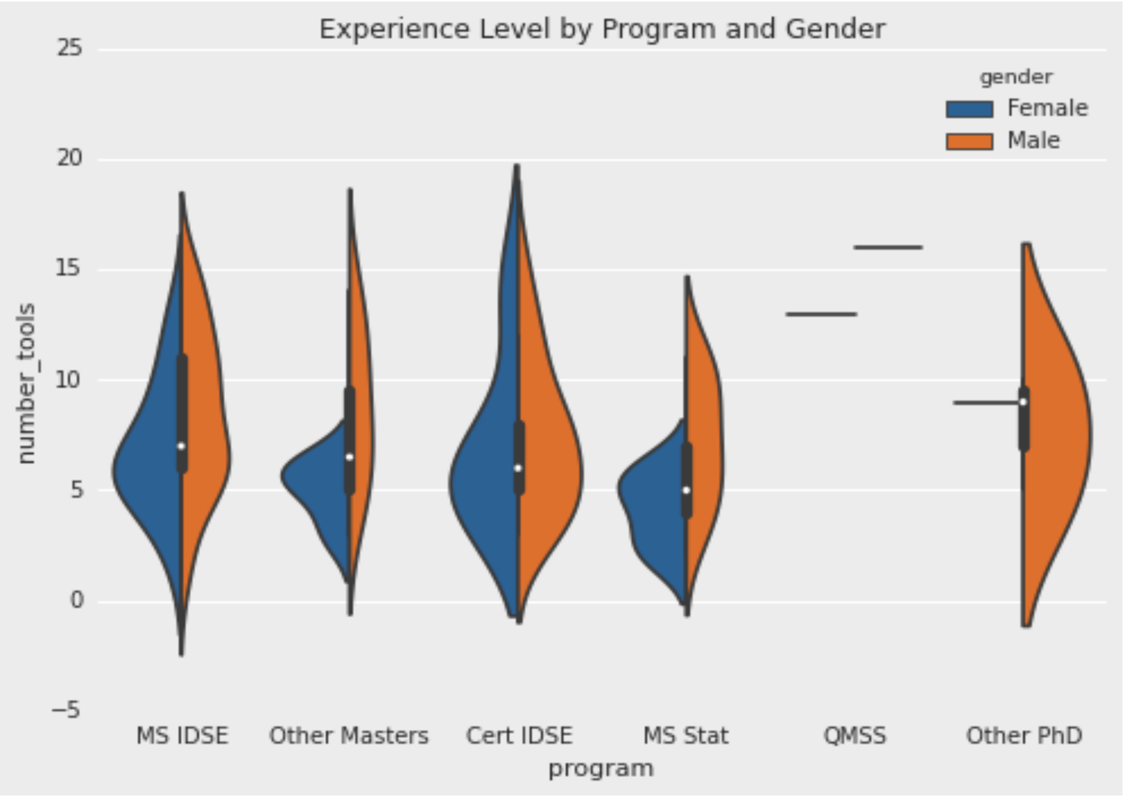
This histogram shows the distribution of the number of tools known by the class. The mean tool count is 7.605, with STD 3.52, min 1 and max 16. The data is fairly symmetrically spread around the mean, with a good deal of mass above the mean, offset by a large spike at 6 tools, with over twice as many respondents knowing that amount of tools than the next highest bucket.



This heatmap compares students across different programs based on an average count of the number of tools they have learned, split out by gender. The QMSS students have experience in the highest number of tools, followed by PhDs. Both figures are likely skewed by how few QMSS and PhD students there are in the class. The MS Stat students seem to know the fewest amount of tools.There does not seem to be much variation in skill level between men and women.

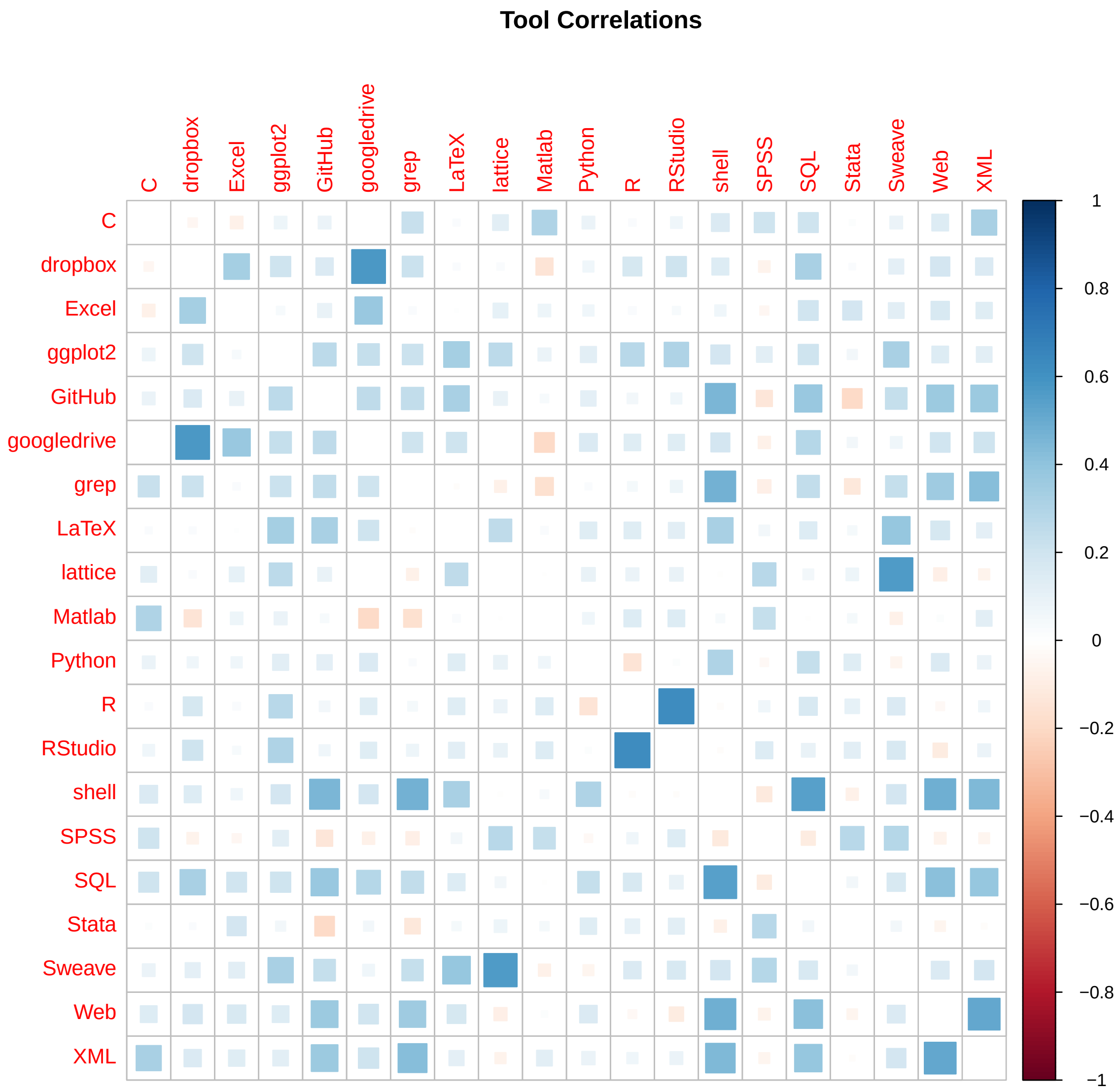


This violin plot compares the distribution of the number of tools known by respondents between programs, and within programs between genders. We were interested in seeing if there were differences between programs in the number of tools each respondent knew. From the plot it appears as though there's higher variance amongst Male respondents even though the means between genders and programs are similar.

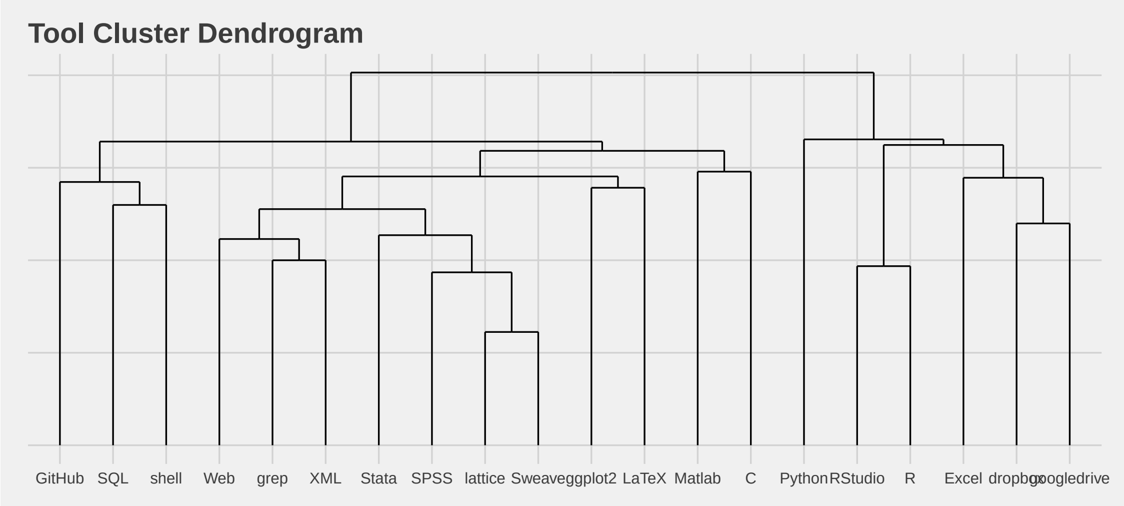


This is colored matrix shows the positive or negative correlation between tools. Here are some of the correlations that can be observed:

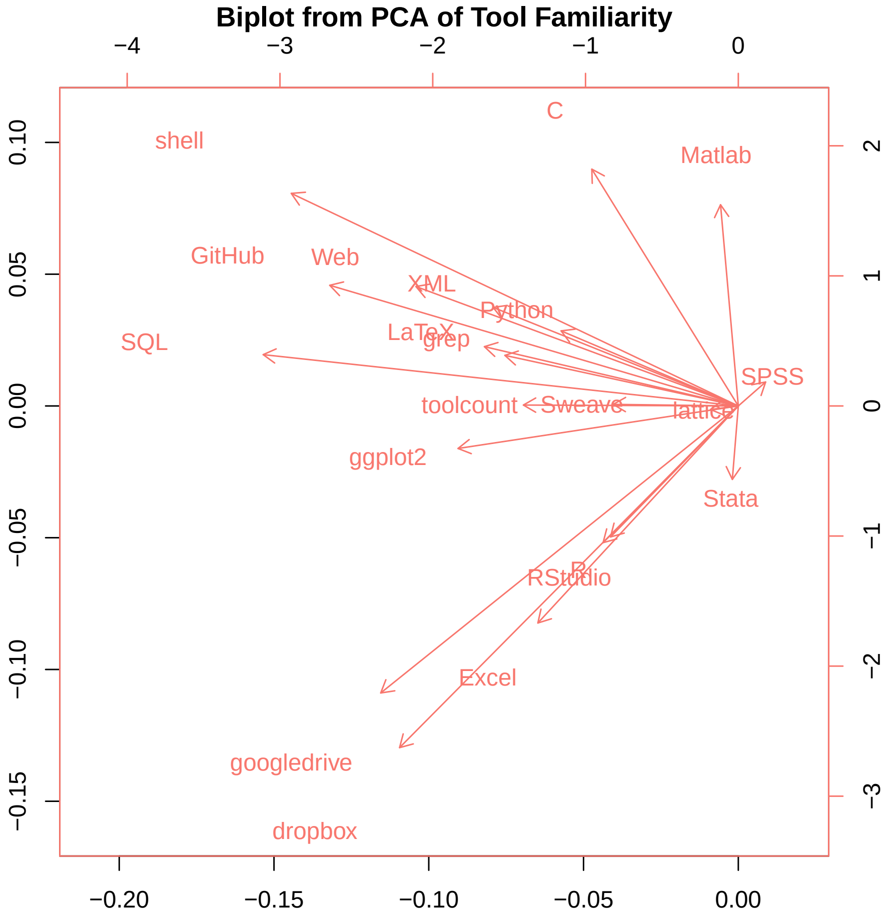
- weak negative correlations between stata and github
- strong positive correlation between r and rstudio (of course)
- strong positive correlation between dropbox and googledrive
- strong positive correlation between shell and github, SQL, grep, SML, Web
- no correlation between r and python
- no correlation between r and excel
- no correlation between c and r



This table displays the pairwise correlation of familiarity with each tool. A positive correlation between a pair indicates that an individual that knows one tool means that it is more likely the individual knows the second language then the class as a whole. The table can help reveal what tools are often learned together or that may rely on each other. For example, R and RStudio has the highest (positive) correlation which is unsurprising since familiarity with RStudio would suggest familiarity with R, and the two are typically used in combination. Other highly correlated tools are google drive and dropbox (both collaboration/cloud storage services), lattice and sweave (both R libraries), and shell and SQL. Looking at whole rows (or columns) reveals which tools tend to be known by individuals with a larger set of tools. For example, SQL and shell have multiple tools with a positive correlation suggesting that if someone knows SQL or shell, they probably know a larger number of tools than the class average. Conversely, someone who knows Stata, SPSS, or MatLab probably does not know more tools than typical. Overall, most correlations are positive suggesting knowing a specific tool means that you are more likely to know a greater number of other tools.



This dendrogram shows the agglomerative clustering using average linkage for the twenty tools. It reveals which tools are most similar by the individuals they share in common. The dendrogram includes some expected pairings based on the correlation table with highly correlated pairs grouped next to each other, but it is surprising that the very first split separate R from the different R libraries. It seems like the first split may be a split between common and uncommon tools since the six tools on the right branch are the six most commonly known tool in the survey.



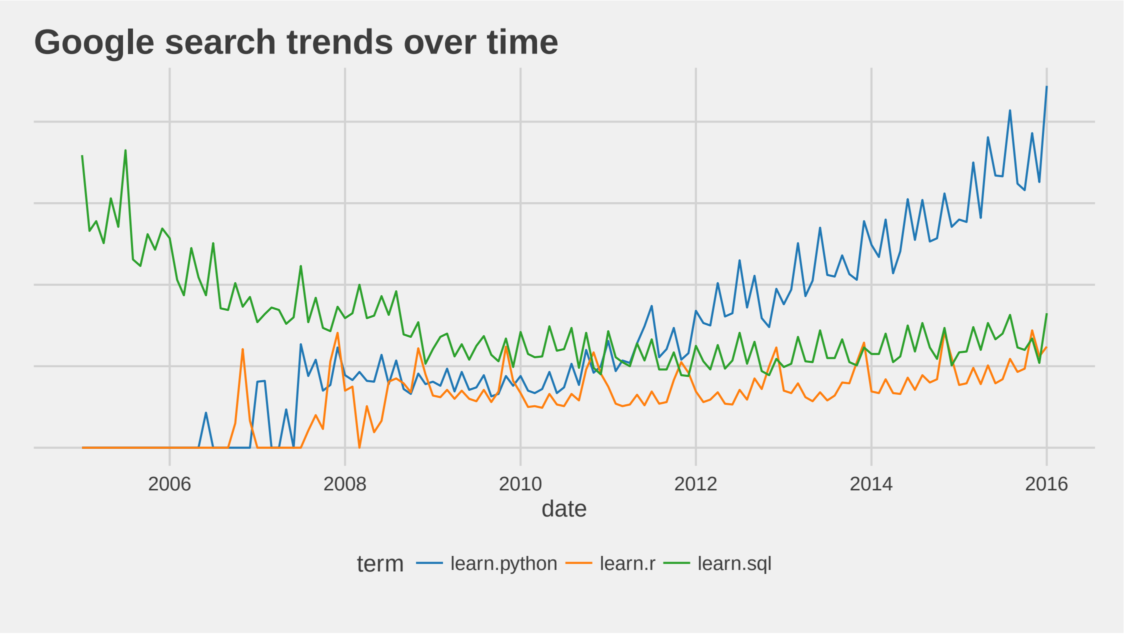


Figure 2: Data Source: Google Trends

This biplot is formed from a principal components analysis of the twenty tools as well as a total count of the tools. Most of the tools either point toward the top left or the bottom left, and the split is similar to the first split in the clustering with Python being a noticeable exception. The first principal component seems to be primarily overall tool expertise. This is supported by the fact that toolcount points directly to the left, and most tools point to the left half since knowing one tool means that you are more likely to know other tools. The exceptions to that are the same as from the correlation plot with Matlab, Stata, and SPSS all suggesting that familiarity with these tools doesn't make it more likely to know other tools. SQL and shell have the most negative scores on the x-axis which also agrees with the correlation plot in that knowing one of these tools suggest knowledge of a wider array of other tools. The second principal component is more difficult to interpret. Tools with high scores on the second principal component (C, shell, and Matlab) are all scripting languages that may be more common among students with an engineering background. The tools with the lowest score (googledrive, dropbox, Excel) are not scripting languages and are known by a more general group of people with or without programming experience. Then, in the middle, are statistical languages and packages such as R, Stata, and ggplot2.

*In order to make this analysis more relevant to the overall field of Data Science we have brought in data from google search and information from a survey provided by O'reilly.* Below is a comparison of google search trends for three terms: “learn python”, “learn R”, and “learn SQL”. The time period of the graph goes from 2004 to January 2016 and the left axis shows a scale with units relative to the highest point in the graph. It can be observed that from the three term only SQL has a negative trend, but it appears to trail off and stay consistent towards the end. Also Python has an exponential increase VS R which has a small gradual increase.

O'reilly hosted an online survey about Data Science which was open to their audience from November 2014 to July 2015.The survey had 820 respondents from 47 countries, 38 states and across multiple industries. One quarter of the of the respondents have job titles that fall under Data Science and the rest of the sample comprised mostly of students, postdocs, professors, and consultants.The image below is the distribution of the responses to the questions: Which of the following tools do you use?

A similar questions with fewer options was asked in our class survey and produced the following distribution. Although the two surveys differ in population size and experience levels, it is easy to see that tools such as R, Python and Excel are the most used by both populations.

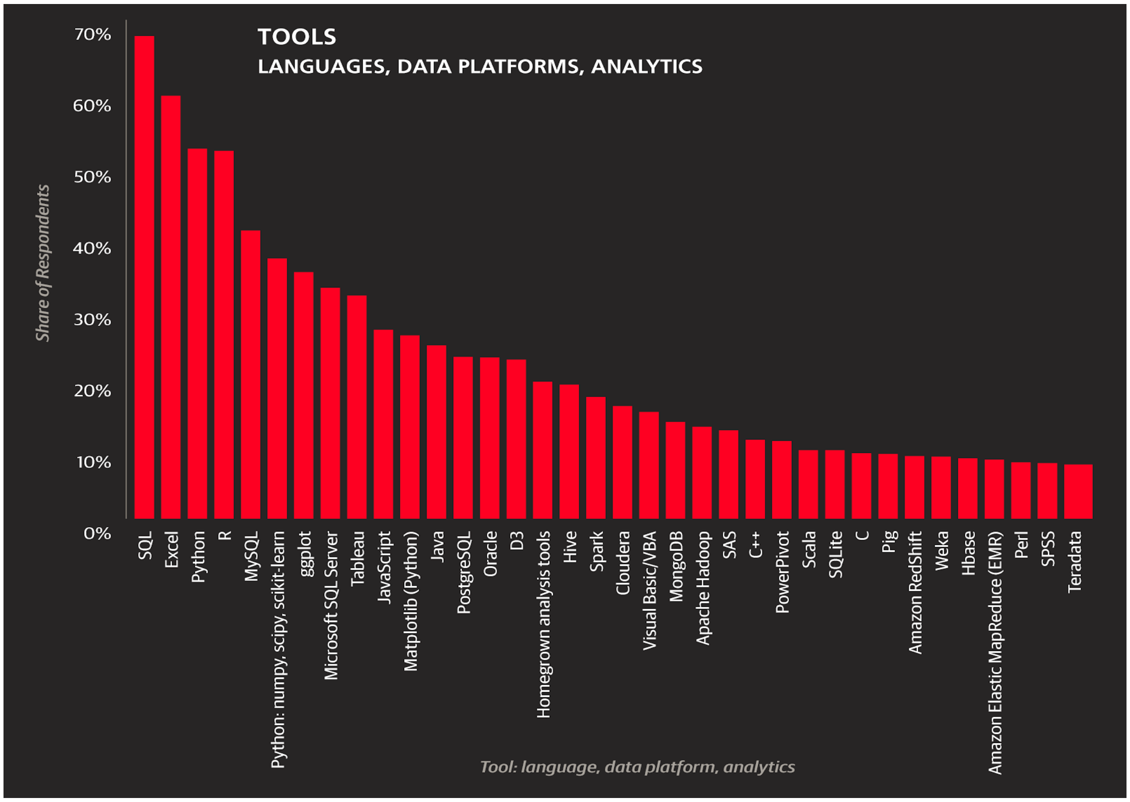
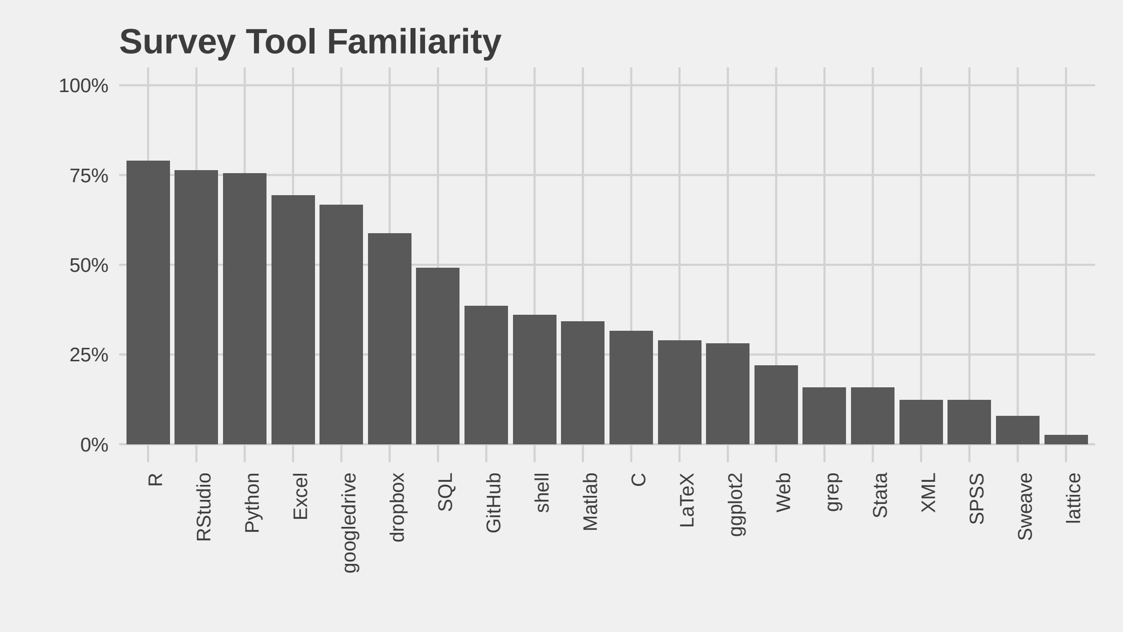
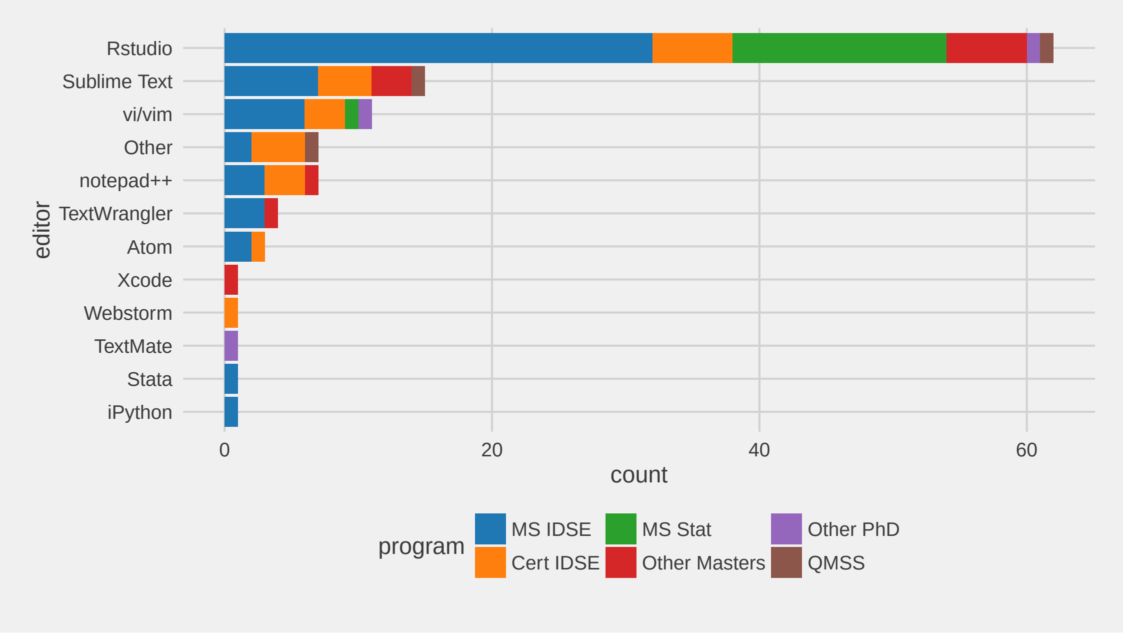


Figure 3: Data Source: O'reilly 2015 Data Science Survey



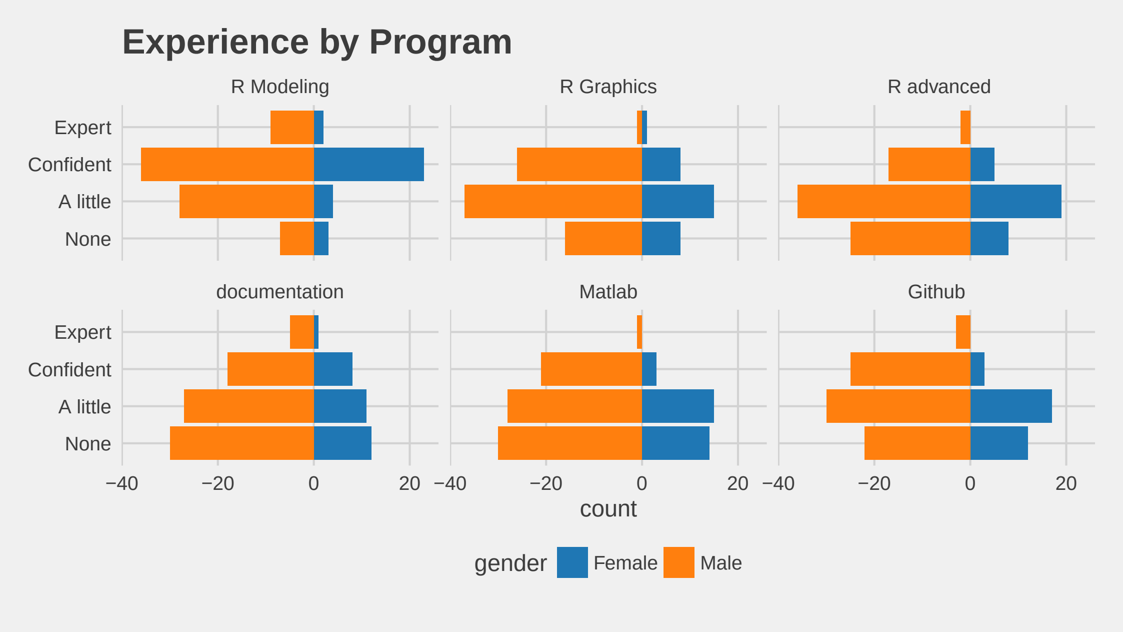
Below is a barplot of the number of students that use each respective text editor. We also added a color to differentiate student from different programs. To achieve these groupings we combined term that referred to the same text editor. RStudio is the most frequently used text editor by our class, based on the survey results. This could be due to the fact that R is the main language used in this course. Often, a person's preferred text editor depends on the task at hand: RStudio for R, iPython for Python, Sublime for JavaScript. RStudio and iPython are also computational environments, which fall into a different category than TextWrangler, Atom, Sublime, etc.



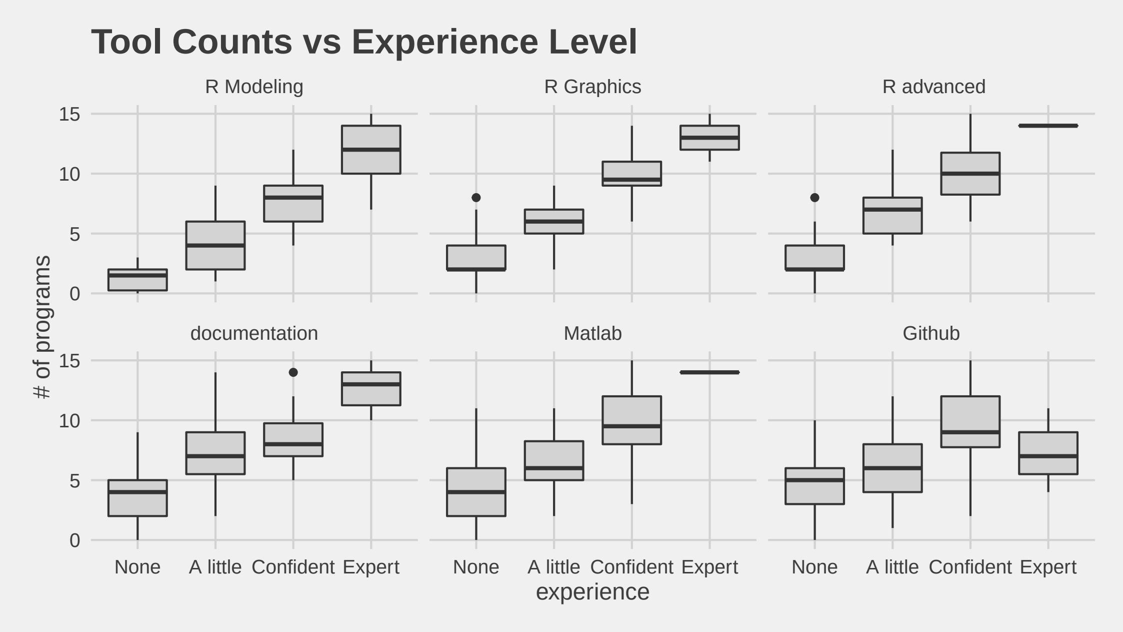
## 4 Experience

In this section we will focus on observing the relationship between specific technical experience and the rest of the variables.

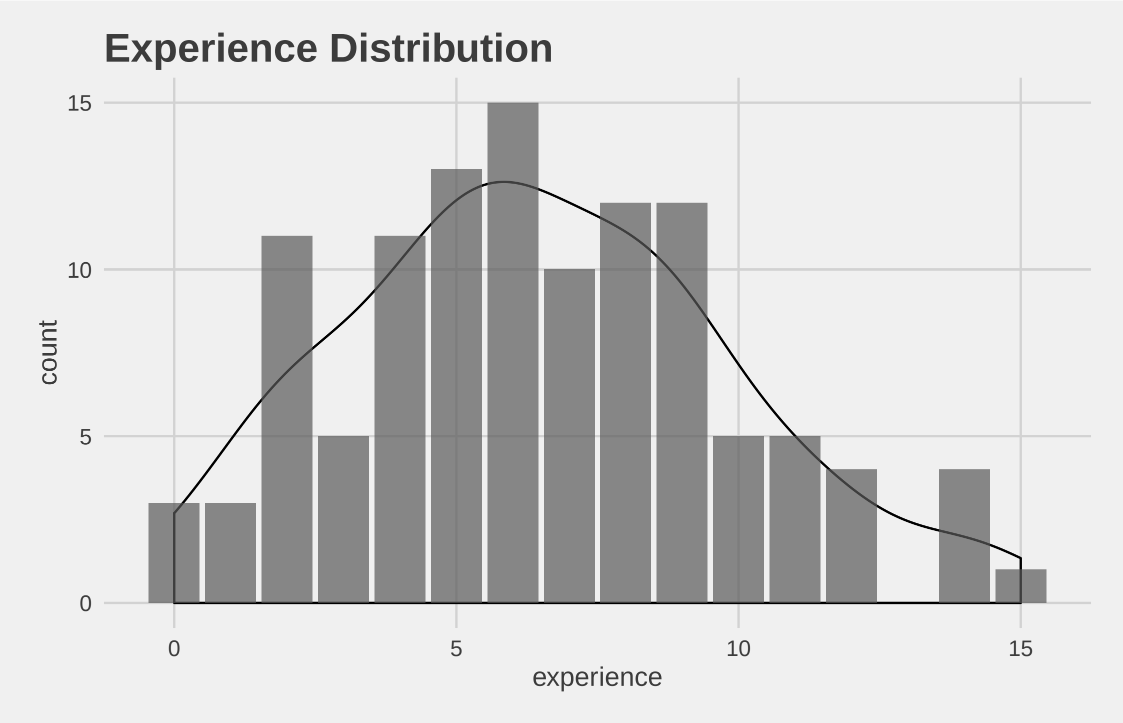
This series of butterfly bar plots provides a general idea for the amount of male and female students that felt they have experience across six different areas and tools.



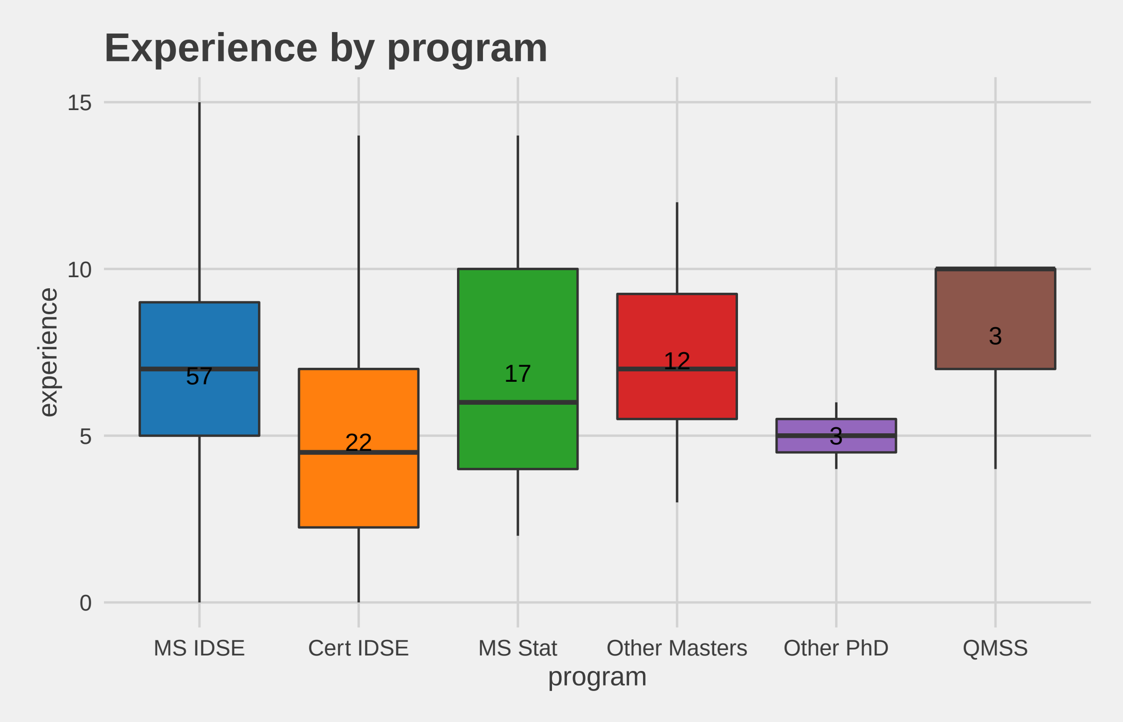
The panel of box plots below compares the confidence level of a specific skill(modeling, graphics, documentation, etc.) to the number of tools one has experience with. As the number of tools increases the confidence level with respect to a skill also tends to increase. For example, students who claim to be “Expert” in R-modeling knew on average 12 total programs, while students who claimed to know “A little” with respect to R-modeling averaged about 5 total programs. One exception is seen in the Github experience boxplot. Respondents who claimed to be experts in Github had not necessarily been exposed to a higher number of tools compared to respondents who claimed to be confident in Github.



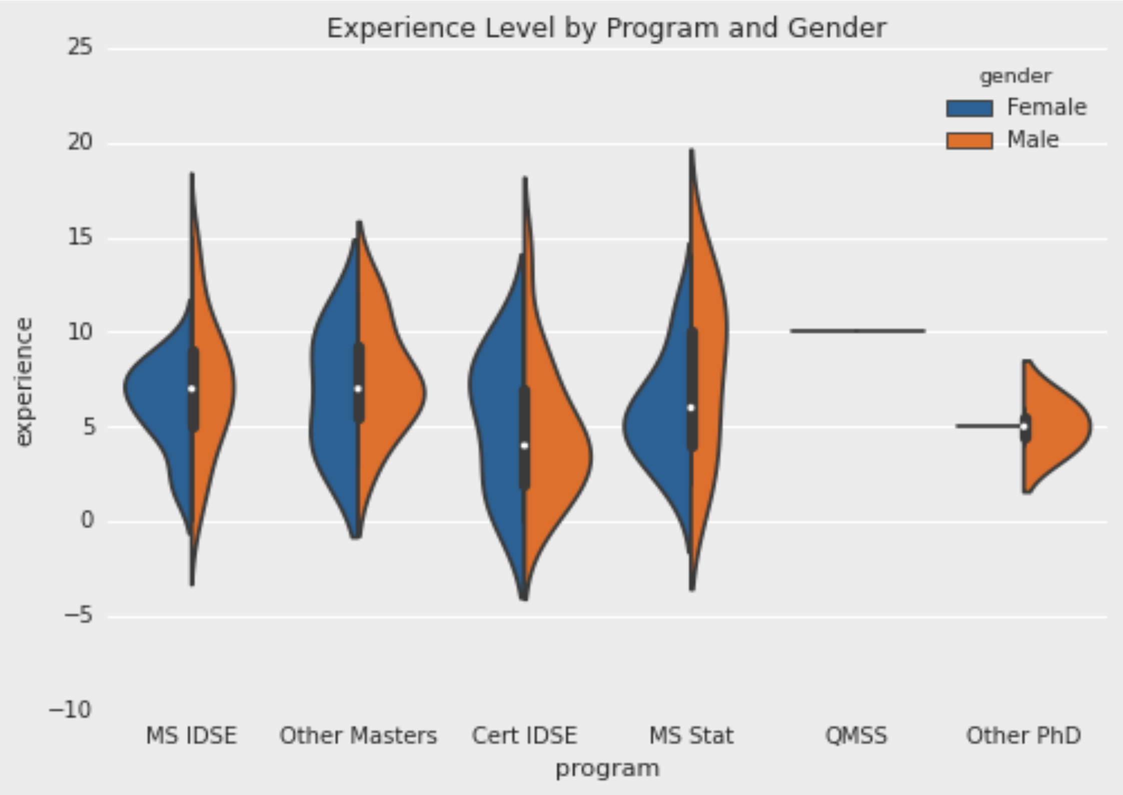
This histogram plots the distribution of experience levels in the class. We converted experience levels (None, A little, Confident, Expert) to values 0 - 4 for the experience-related columns so we could sum them and get a sense of the overall experience level of respondents. Given there were four experience columns, the maximum possible ‘experience’ value is 16. The mean for the class was 6.46, with std of 3.37, min of 0 and max of 15. The distribution is fairly symmetric around the mean, with a slightly longer upper tail indicating that most respondents have some experience with analytical tools and techniques, but not extensive experience.



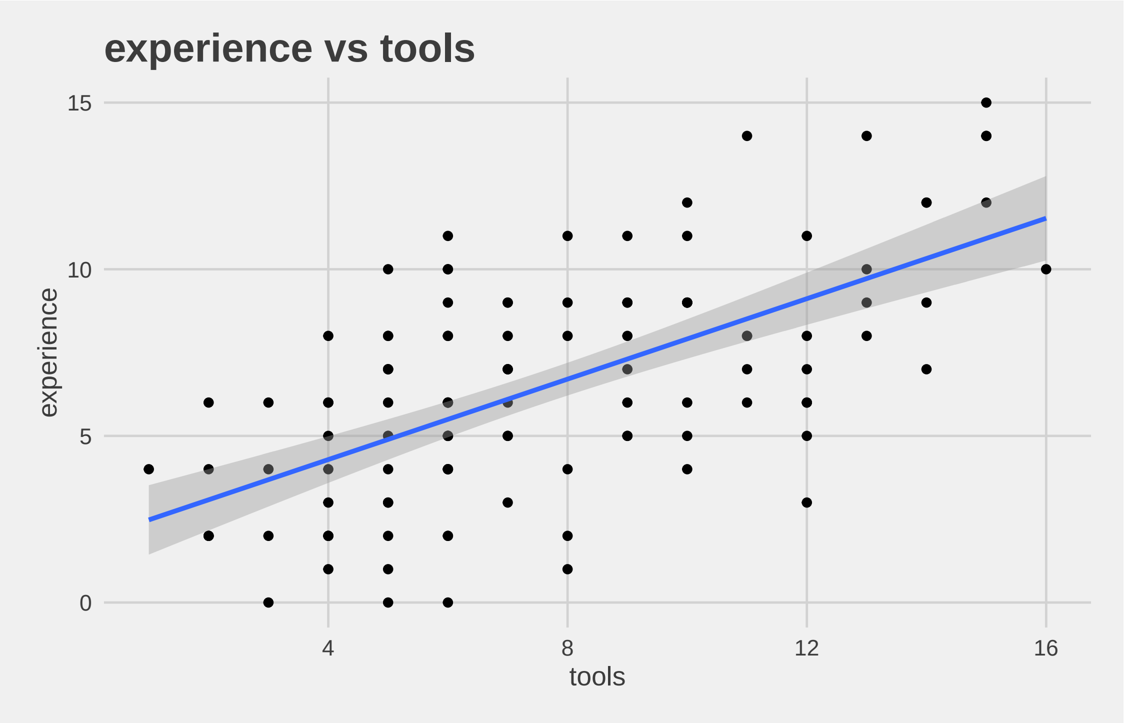
This box and whisker plot also shows the distribution of the experience levels (as described above) within different programs. Because some programs have less than five respondents it's difficult to make comparisons across groups. That said, there are some interesting things that we noticed about the plot. First, the Data Science programs have the widest ranges in experience levels. They both have the lowest and highest experience levels, though the certification mean lower than the MS (and lowest of all the programs.



Like the boxplot above, this violin plot also compares distributions of experience levels between programs. This plot, however, also compares experience levels within groups. Further, instead of bounding the box at the 25th and 75th percentiles, the violin plot uses kernel density estimation to estimate the distribution of the experience data. Distributions for Female and Male variables are reflected over the experience axis to show differences in distributions within those categories (and the program category). This plot also provided some interesting insights: the Male category generally had greater variance in experience levels. There also appeared to be an inclination towards bimodality in many of the distributions, with a higher and lower skill groups seen in many groups.



This scatter plot compares the number of tools the respondent indicated having experience with and their overall experience (tools, languages, etc.). This plot layers a linear regression line on top of the scatter plot, along with error bands, to show the linear relationship. It's immediately apparent from the linear model and error bands that there's a strong positive relationship between the two variables.



## 5 Conclusion