# iMentor Project

*12.15.2016*

—

**W205 - 1**

Nicholas Chen
Stephanie Fan
Hyera Moon
Leslie Teo

# Executive Summary

One of the keys to success for iMentor is maintaining a strong mentor foundation to help guide their mentees. Two datasets, mentor demographics and message metrics, were analyzed for more insight on indicators of mentor dropout. Although no strong indicators were found, a successful framework for continued analysis has been established using Python, R, Tableau, and Postgres. It is expected that as iMentor's programs improve and grow, the analysis will yield more strategic results.

# Background

iMentor is a non-profit organization that facilitates mentoring relationships to ensure more students from low-income communities enroll and graduate from college. Partnering with public high schools, the iMentor program matches students with mentors for the majority of their high school career.

Today, iMentor serves more than 6,000 students through its direct-service programs in New York City, Chicago, and the Bay Area and through its partnerships with local non-profit organizations that implement its model in 39 schools.

# Goal

One of the key challenges is preventing mentor dropout. iMentor's own analysis revealed that a significant percentage of mentors do not complete the 3- or 4-year mentoring commitment, with most dropping out in the second or third year. The goal is to identify how to retain mentors by analyzing main factors and predicting possible dropouts for potential program intervention and improvement.

# Data

1. iMentor online platform data
    a. Messaging data

    iMentor's online platform tracks required communications between mentors and mentees on lessons as well as chat communications between students. To help protect student and mentor identities, only communication metrics such as frequency and length were provided. One set was provided each for mentors and mentees.

    b. Survey data

    Three times each school year, a survey is sent and collected from both mentors and mentees through the iMentor web platform. The data is then stored in the same platform and can be retrieved for analysis.

    c. Application forms

    Mentors application forms are also collected and stored in the iMentor platform. These forms are the primary resource of mentor demographic information.

2. Master data

Master data is the primary file that is compiled annually by iMentor for their annual report. It combines survey data that is taken three times a year as well as multiple files from the iMentor online platform data, academic trends for the students, and college graduation and attendance in college.

All datasets were provided by iMentor as .csv files. Additional supporting documents, such as data dictionaries, were provided as PDF documents.

# Data Processing

Data processing and transformation was built on Postgres. The entire process is summarized in Figure 1.

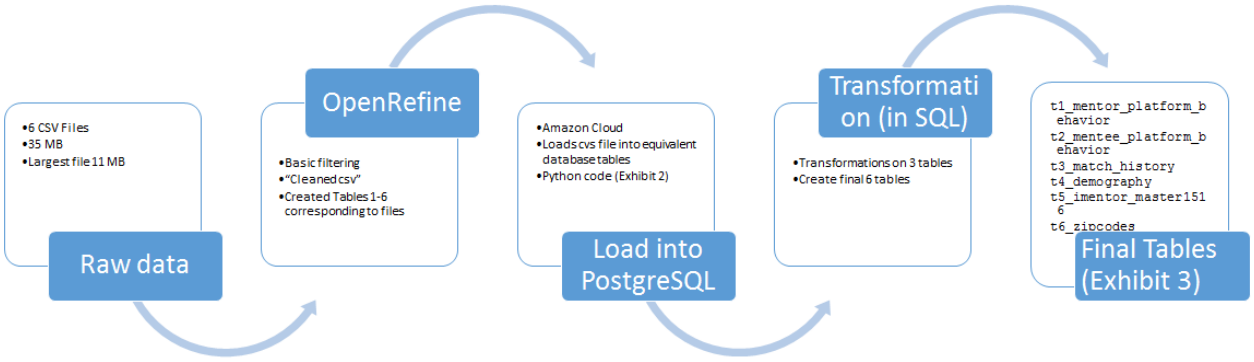


**Figure 1 Data Ingestion, Transformation, and Extraction**

Amazon Redshift was used to deploy a cloud-based Postgres server. Postgres was primarily chosen due to its basis in SQL as iMentor currently uses relational databases to store data (Figure 2). Amazon Redshift was chosen because of its ease of setup and demonstration for collaboration. We also explored Microsoft Azure as a possible means to implement but their solution relies heavily on MS SQL server. Postgres on a virtual machine is possible to set up, but setting up a permanent IP required a paid account and would incur charges. For this demonstration, we did incur some charges on Amazon, approximately $70 for two months. It is not anticipated that Amazon Redshift would be required for use by iMentor as the current data size is still small (our database was about 500 MB in size) and could be handled by internal SQL databases.

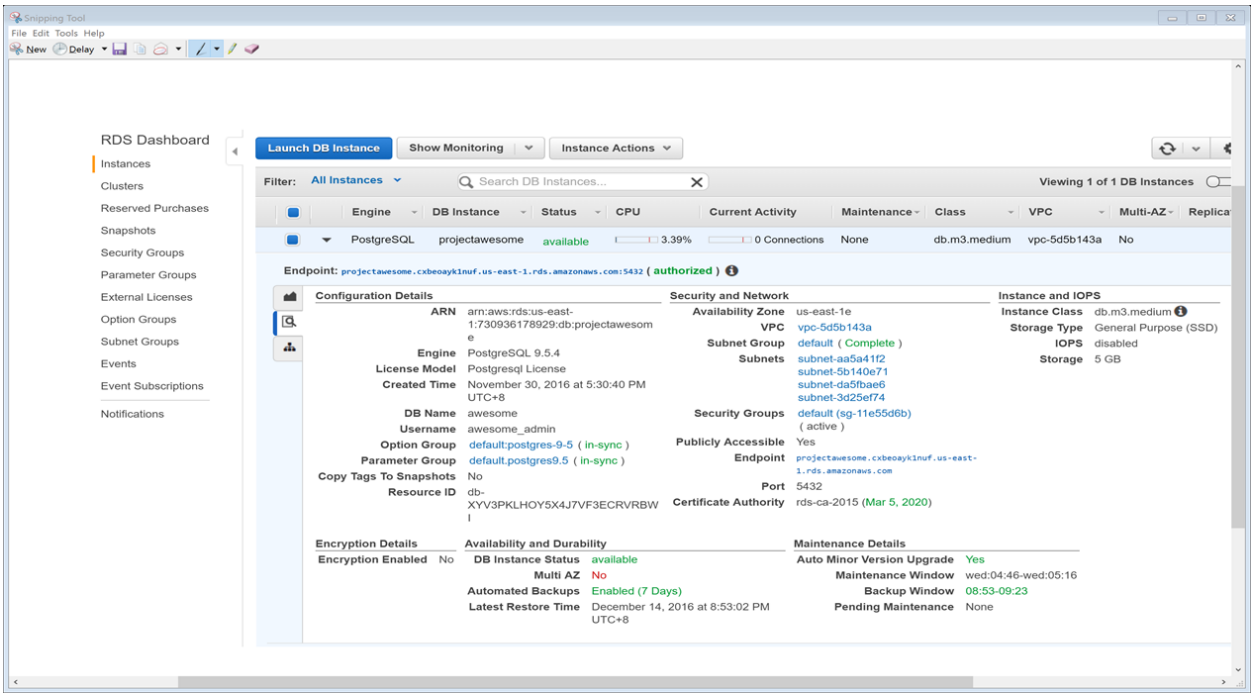


**Figure 2 Project Awesome's Database on Amazon**

Data files were acquired as comma separated value files and were loaded into the cloud via Python scripts, after filtering through OpenRefine. We did not, however, use this tool to make changes given that the process could not be automated. This means that we would be ingesting some of the data in the most simple form possible (text, for example rather than date or integers or well-defined string variables.) We do not believe this is an issue given the database size but may be if one were working on a much larger and more complicated data.

Data processing had three steps:

1. Ingestion of raw data into basic tables. This means we have access to the raw data.
2. Cleaning, transformation and processing of some of these variables
3. Construction of new final tables, which contain data used in our analysis (Figure 3).

Those new data tables were analyzed via Python and R. A Tableau connection was also used to demonstrate to the client how to visualize directly from a cloud-based database.

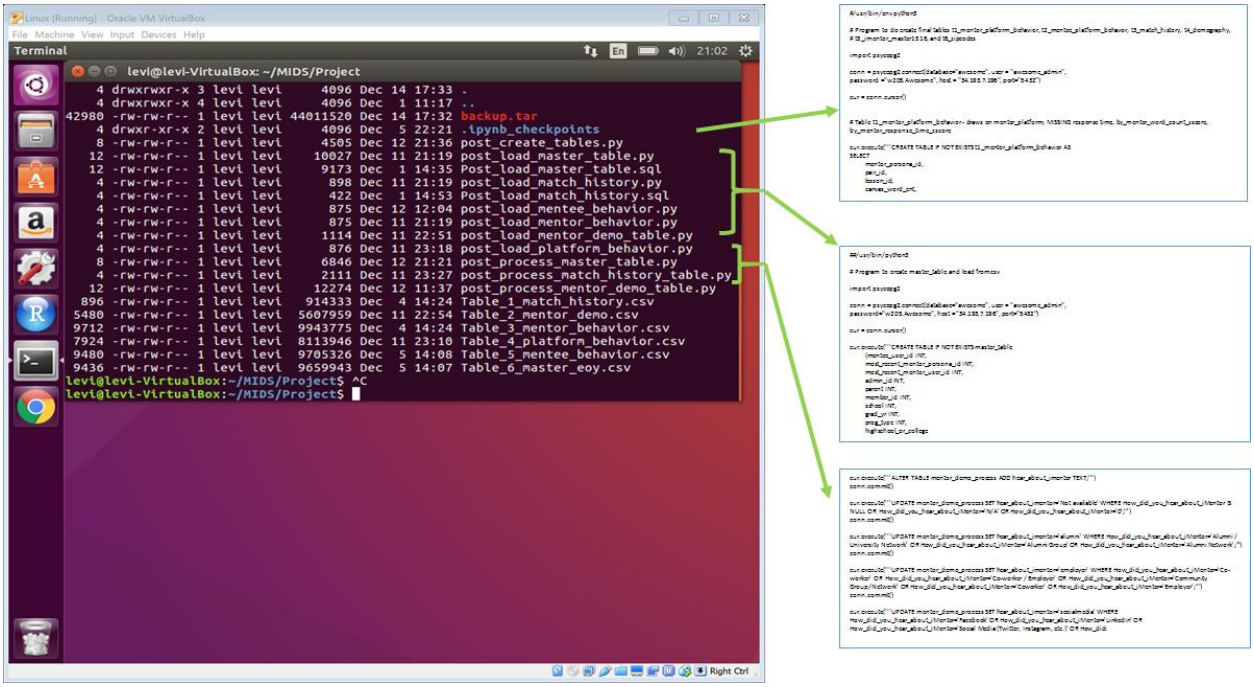


**Figure 3 Codes for Data Processing**

The final schema used is shown in Exhibit 4.

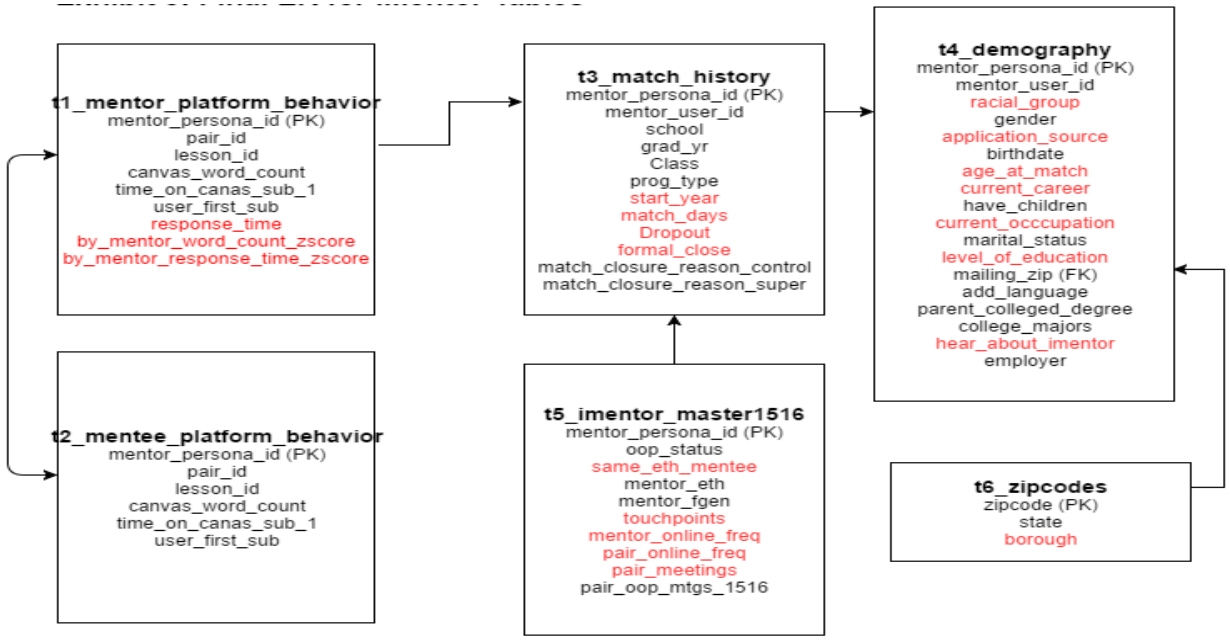


**Figure 4 Final iMentor Tables**

# Findings

## A. Demographics

Demographics were taken from the mentor demographic data file provided by iMentor. The demographic information was collected by iMentor through their online platform from mentors application forms.

iMentor has two programs, a 3-year program and a 4-year program. By comparing the mentors in the two programs, the number of dropouts was substantial in year 2 and year 3 in both programs, and the number of mentor dropouts was significantly higher in the longer program, which suggest the length of the program may be an important factor in influencing dropout rates. In this analysis, beside the length of the program, our goal is to explore whether the mentors' demographic attributes can also be a significant indicator of dropout and whether a predictive model can be built from these features.

Several analysis techniques were used to assess whether a dropout prediction model could be built from solely mentors' demographic attributes. As certain algorithms limited the number of outcomes that could be used, the analysis was performed against formal closure (i.e. mentor-mentee pair completed the commitment) and mentor dropout groups only. The dataset was split into a training set and test set with a 2:1 proportion of data, respectively. In some cases, bootstrapping was used to balance the number of dropouts and formal closures for a more accurate analysis.

### *Unsupervised learning*

First, principal component analysis (PCA) was used to reduce and group the number of demographic attributes into a more manageable number of attributes and for easier visualization of the results. When plotted in a 2D graph against the first and second principal components, mentor dropouts could not be distinguished from the formal closure group (Figure 5).


**Figure 5 Principal Component Analysis by Mentor Dropout or Formal Closure**

In fact, collapsing the demographic features into only 2 components could not explain the variance within the dropout and formal closure groups attributes, and thus, using more than 2 PCA components would be necessary.

Using the two first PCA components, several clustering methodologies (k-means clustering and Gaussian Mixture Model (GMM)) were applied. As expected, the combination of clustering algorithms and principal component analysis could not accurately identify the two groups. However, it is anticipated that as the program is further refined, fewer principal components will be needed to differentiate between the formal closures and mentor dropouts, and clustering via this method could become useful.

An algorithm optimization script was written to optimize the number of PCA components and GMM components needed for successful clustering. The optimized algorithm yielded a 48% success rate in predicting mentor dropouts and formal closures correctly. Thus, this model was not selected given its complexity and low accuracy.

### *Supervised learning: Decision Tree*

A separate model was built using a decision tree classification model. The Pearson correlation and Chi squared test were performed to implicate highly contributing factors to use for the model. Using the statistical significance, factors were ranked in order of impact.

Most demographic features did not show a strong relationship with the mentor dropout, which led us to expect that the prediction model based on these features would not be as accurate as a model with additional features.

The final mentor dropout predictive ability of the decision tree model was 65%. The low accuracy could be due to the small number of formal closures in the dataset; iMentor had its first class finish the 4-year program in 2016. In addition, demographic attributes of mentors may only partially explain the likelihood of mentors dropping out. There are additional external factors that could also influence dropout but are not captured in the dataset, such as moving for work or relationship with mentee. A detailed decision tree flowchart can be viewed in the project GitHub repository.

Based on the decision tree model, the top 10 most important features for predicting the mentor likelihood of dropout were:

1. Career - not specified
2. Parent college degree - not specified
3. Heard iMentor from employer
4. Level of education - 2-yr college
5. Career in Government
6. Racial group - African American
7. Racial group - Asian
8. Career in Tech
9. Level of education - not specified
10. Age at match - 30's

A predictive score was applied to each current mentor by using the results of the decision tree analysis. This data was fed into a Tableau dashboard along with other demographic data from the database to demonstrate how predictive analysis could be visualized by iMentor administrators (Figure 6).

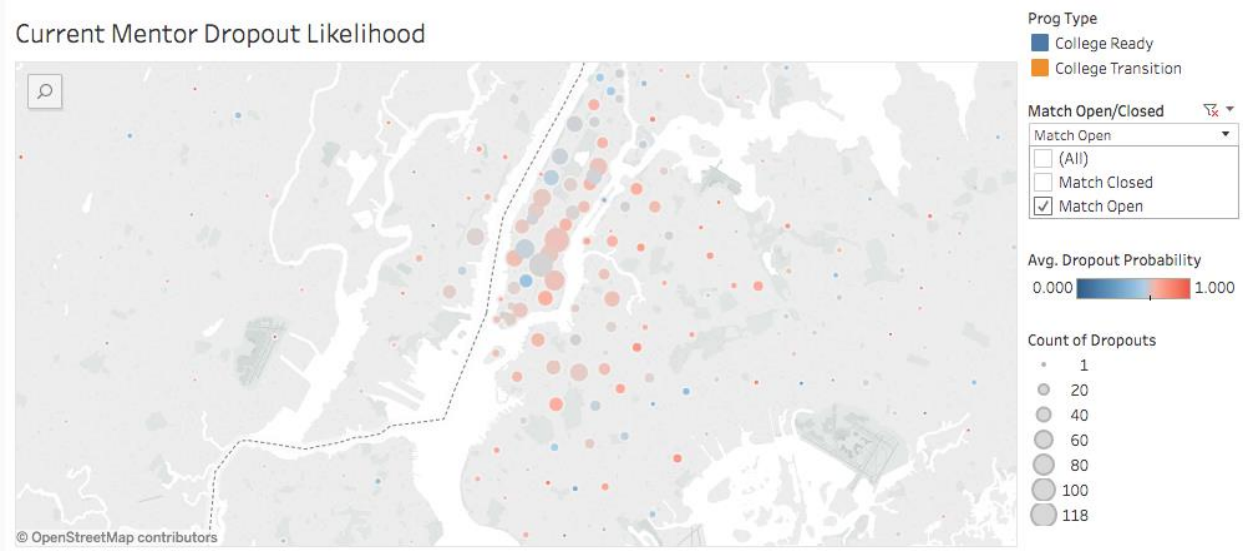


**Figure 6 Tableau Visualization of Current Mentor Dropout Likelihood based on Demographic Analysis**

Ultimately, we recommend that iMentor move forward with the decision tree analysis as this method is easier to conceptually grasp and explain to key stakeholders than unsupervised learning of combining PCA components and clustering methods.

## B. Message Analysis

Two datasets were provided for message analysis – one for mentors and one for mentees. The datasets provided aggregated message metrics, such as wordcount and time since the last message. Contents of messages were not provided to comply with student and mentor privacy concerns.

All datasets were first data cleaned prior to analysis. This largely consisted of removing negative values and filtering the dataset to include mentor-mentee pairs that were active at the beginning of the 2015-2016 school year and where data was available for both mentor and mentee. The dataset was categorized into 5 outcomes -- formal closure, match open, mentee can no longer participate, mentor can no longer participate, and program partnership ended. Based on the analysis, no practically significant predictor was found that would allow iMentor to predict mentor dropout.

The most promising predictor for mentor dropout rate in the message analysis was based on ignore rates. A mentor ignore was defined as when the mentee wordcount was non-zero and the mentor's response had wordcount of zero or was missing for the same message sequence. Mentee ignores were not analyzed.

Mentor dropouts showed a statistically significant higher ignore rate compared to other groups using a pairwise t-test with Bonferroni correction (Figure 7).

| | combined.dropout.flag.mentor | ignore.rate.by.group | count.obs |
|---|---|---|---|
| 1 | Formal closure | 0.2341904 | 1439 |
| 2 | Match Open | 0.2243848 | 60267 |
| 3 | Mentee can no longer participate | 0.1630114 | 4742 |
| 4 | Mentor can no longer participate | 0.2871042 | 5653 |
| 5 | Program Partnership ended | 0.2564103 | 39 |

**Figure 7 Ignore Rates by Match Closure Types**

All mentor groups showed an increase in ignore rate over time (Figure 8). Note that data points on this line graph of less than five messages were removed to remove erratic and high ignore rates at the beginning of each series. While on average statistically higher, the mentor dropout group was sometimes indistinguishable from the other groups, and over time, ignore rates varied widely. Therefore, the analysis did not find that ignore rate could be used to predict mentor dropout from a practical sense.

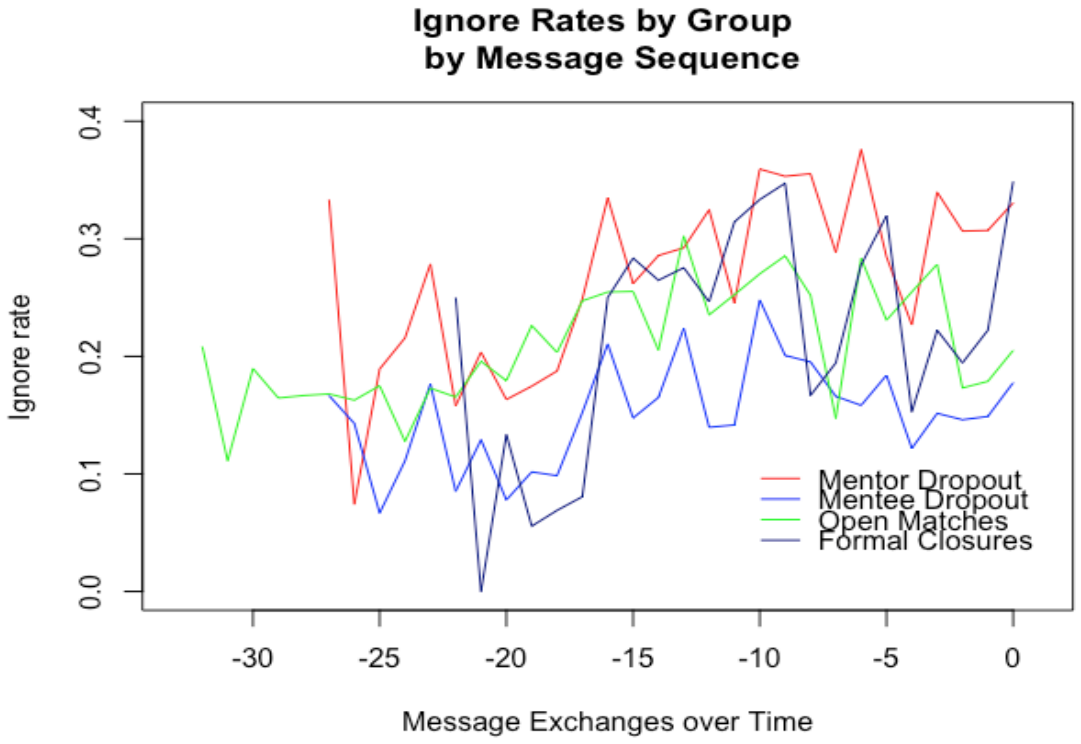


**Figure 8 Ignore Rates by Match Closure Types over Time**

In addition to the ignore rate, several other interesting highlights were found. These included:

### *Mentor Investment*

On the whole, mentors write more and spend more time writing their messages, indicating a higher level of investment in the program than the mentees. Both mentor message length and writing time was almost more than double those of menteess.

### *Blank Messages*

There was an extremely high prevalence of blank messages. Approximately, 43.5% of mentor messages and 30.7% of mentee messages were blank. Blank messages could be due to attachments or images being sent in lieu of actual words.

Strangely, formal closures seemed to have the highest prevalence of blank messages, which was found to be statistically significant using a pairwise t-test with Bonferroni correction. A very distinct time trend was found across all groups (Figure 9). Note that the few observations were removed from Figure 9 included less than five messages in order to remove erratic spikes at the beginning of each series.

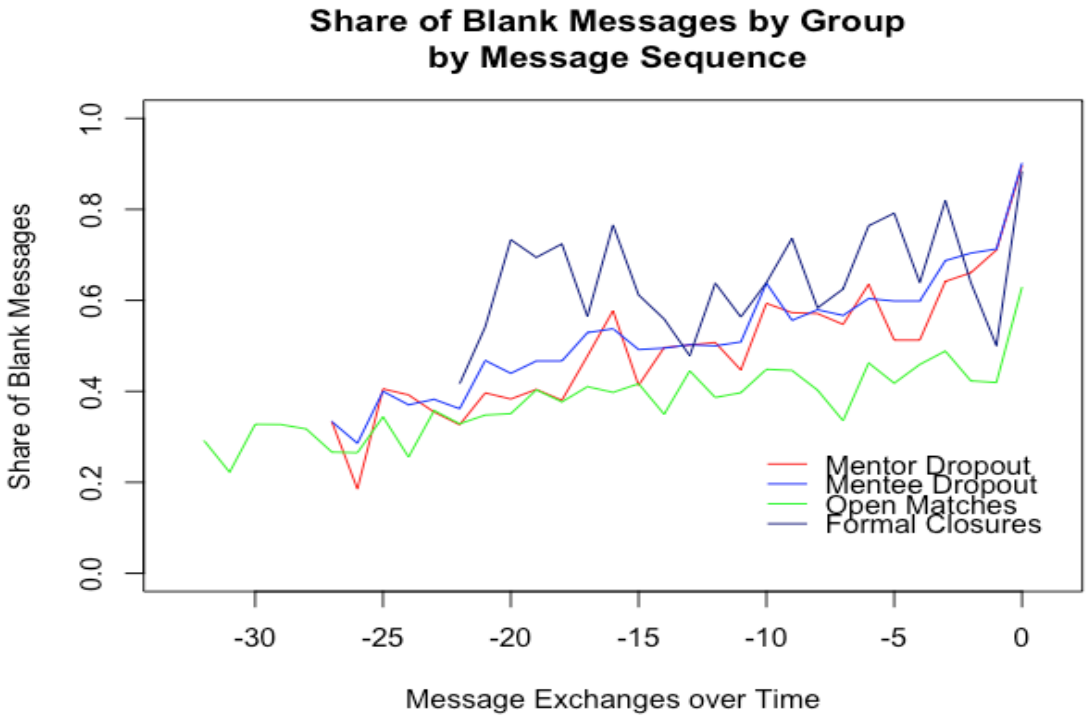


**Figure 9 Proportion of Blank Messages by Match Closure Type over Time**

### *Response Times*

Response time showed an interesting distribution showing peaks for each day of the week, and a significant number of ‘uh oh’ responses of mentors sending their responses just before the end of a lesson (Figure 10). Note that observations in this section were limited to responses with content and response times that were between zero to ten days. There was no statistically significant difference between the mean response time and any outcome; given the distribution of response times though, the mean may not be the most appropriate method of analysis. Response time showed no clear trend among any of the groups as mentors are involved in the program longer.

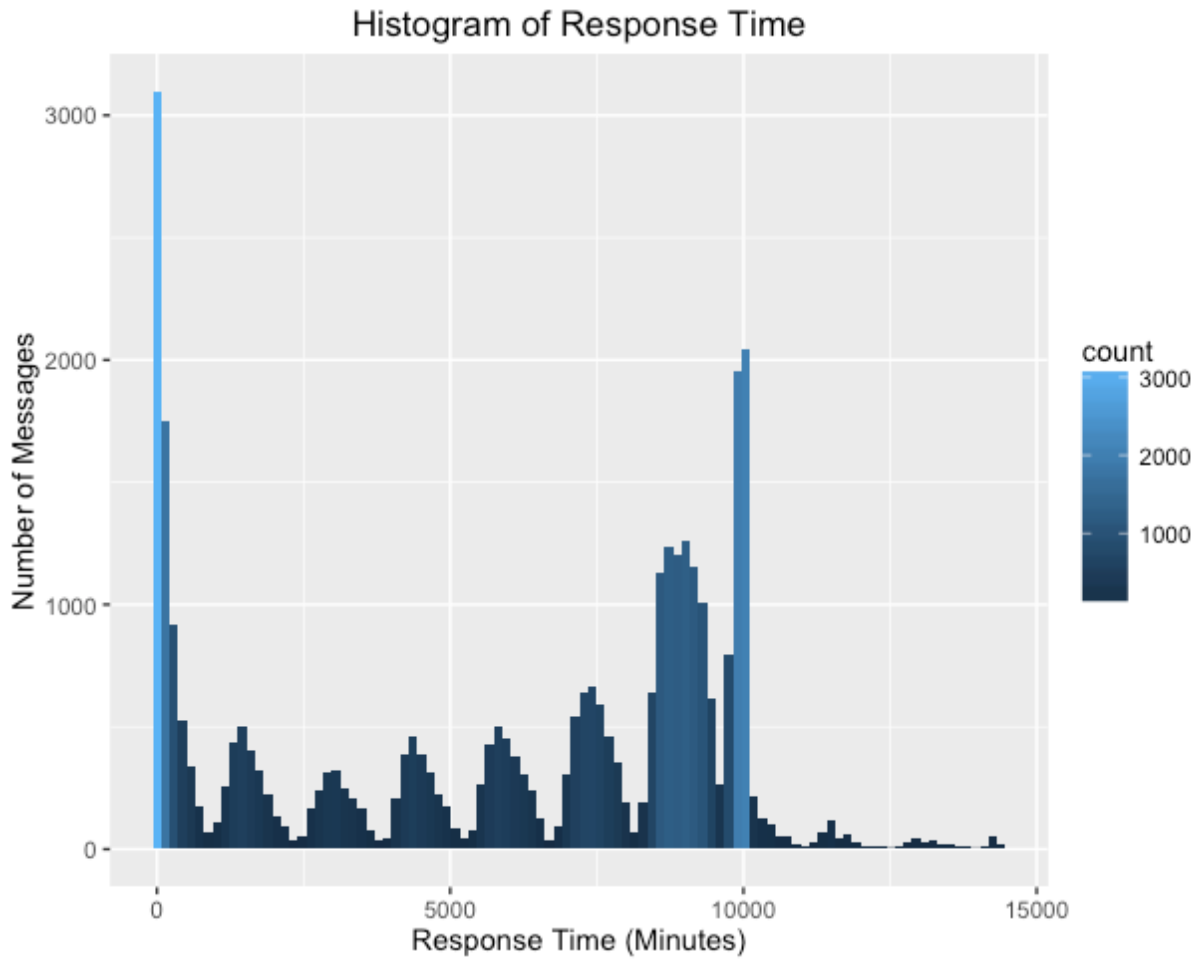


**Figure 10 Number of Responses by Time per Message Sequence**

# Data Visualizations

Two dashboards were put together using the cleaned data after data processing and from the demographic analysis in Tableau. Tableau was chosen because iMentor already uses this program for data visualization and are already familiar with how to link data sources, create, and modify visualizations.

These dashboards are mockups of potential visualizations that could be used to further explore and communicate the data and analysis. Screenshots of these two dashboards are shown in Figure 11 and Figure 12.

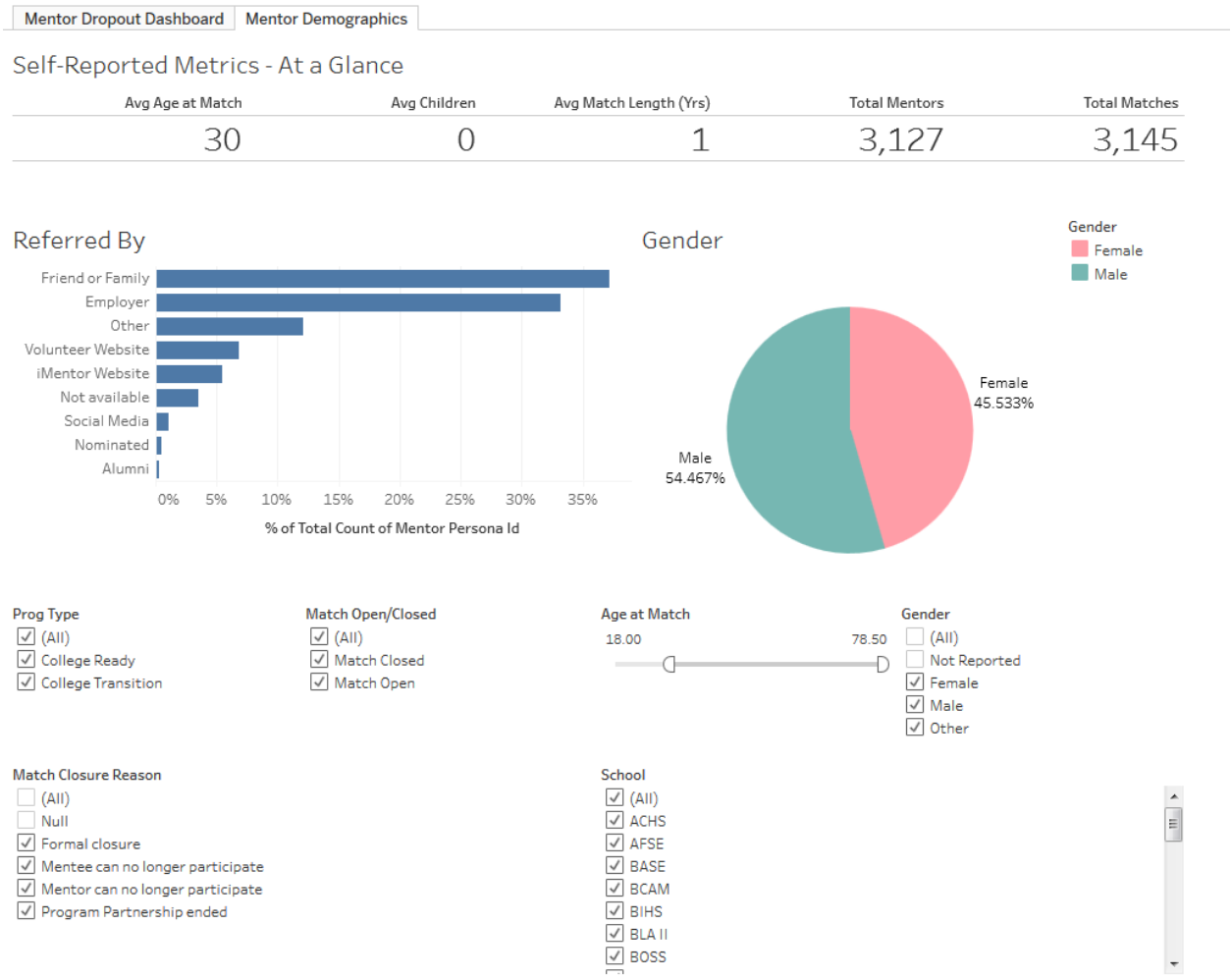


**Figure 11 Mentor Demographics Exploration Visualization**

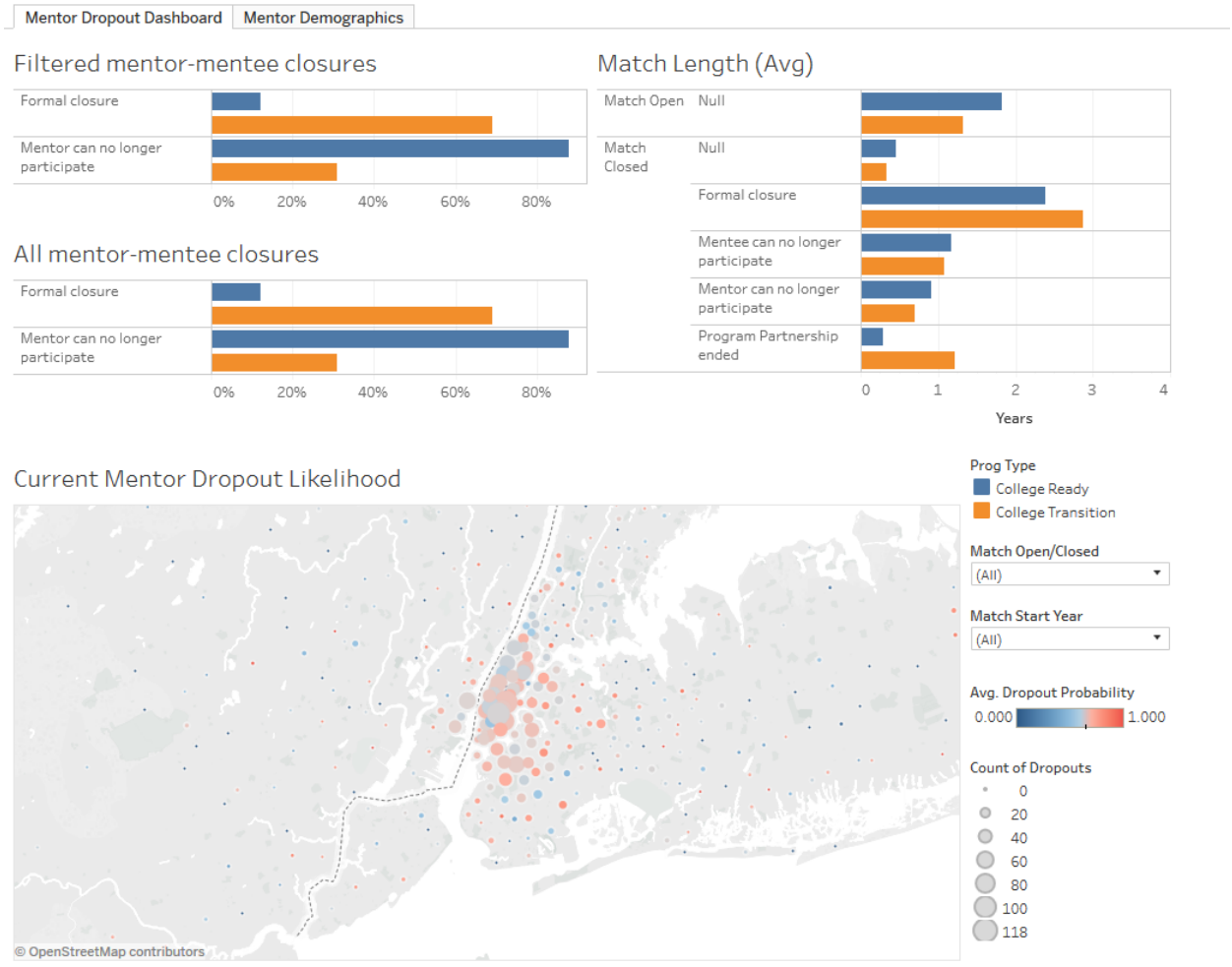


**Figure 12 Mentor Dropout Dashboard**

# Architecture & Implementation

The main tools used are Python, R, Tableau and Postgres. See the readme.txt file in each folder of the Github repository for step-by-step instructions on how to setup and run the data processing, analysis, and visualizations.

***Python libraries:***

numpy
pandas
matplotlib
sklearn
psycopg2
scipy
sklearn
seaborn

***R libraries:***

dplyr
ggplot2
RPostgreSQL

# Next Steps

***Data Processing***

- Data growth
  It is anticipated that much of the iMentor platform data will remain relatively small, such as the tables for match history and mentor demographics, so these tables could remain in relational databases.

  On the other hand, message logs and content as well as the new chat logs and content have the potential to grow exponentially as iMentor approaches its goal of 200,000 current matches. Aggregated metrics, such as the wordcount and times that were analyzed in this report, would likely be small enough to be ported into relational databases for matching, but data storage of content and logs would need to be moved into a Hadoop file system or other redundant storage means to preserve data quality. It is not anticipated that real-time analysis would be needed, so message and chat data does not need to be processed through streaming analytics platforms like Apache Spark.

- Data parsing
  The master data file provided for this report was already compiled by hand from multiple data sources such as the Department of Education, individual school data, and the iMentor platform. Eventually, this analysis should be adapted to pull from the true raw data sources as compiling the master data file will become errorprone as the number of menttes and mentors grow.

- Data flexibility
  The current analysis has largely been hardcoded to filter data. Future flexibility should be added to allow the analysis to be run based on user inputs, such as a date range of interest or a particular school year or class.

- SQL Server selection
  For this project, we used Postgres because of its low cost and ease of implementation and similarity to iMentor's current reliance on relational databases. We anticipate some amount of work may be necessary to convert all code to the final SQL server selection of choice.

***Data Analysis***

- Analysis data
  For the purposes of this project, data was suffiently small that the analysis could be performed offline and data be fed manually into visualizations. Eventually, this sort of analysis should be performed automatically and results stored back into the cloud-based database for analysis. This would also allow for auto-updating of visualizations, so that users would not need to manually load datafiles and could remain permanently connected to the cloud-based database.

- Analysis robustness
  One of the weaknesses of this analysis was that the proportion of successful formal closures is small in comparison to the number of mentor dropouts. As the program continues to grow, more formal closure data points will become available and make the analyses more robust.

- Analysis depth
  *Demographics*
  For this analysis, many of the mentor factors examined were static (i.e. would not change over time – like race, college major, and parental education status). For a more robust predictive analysis and insights about mentors' attrition likelihood, future analysis should combine these static demographic attributes with features that reflect mentors behavior within program (e.g. message frequencies), mentors interaction and relationship with mentees (same interests, hobbies, family background, racial group, languages, etc.) and other features (proximity of work/home to school, etc.).

  *Messages*
  Repeated message traffic analyses for every school year and comparisons between these analysis to detect differences or similarities would improve analysis depth. It would also be interesting to follow the progression of mentors who drop out from year to year to detect patterns of behavior.

  As the program improves, analysis could be used to identify areas for improvement or to verify effectiveness of changes.

  Message content was not accessible in this research. However, if accessible, analysis of the content could shed light on some potential signs of mentor/mentee relationship and dropout likelihood using some natural language processing techniques, such as sentiment analysis.

# Conclusion

Based on the data analysis, the current strongest factor implicated in mentor dropout is the length of the program. Outside of this factor, no strong factors in the demographic or message analysis was apparent.

A well-structured framework was established for repeating the data processing, analysis and visualization presented in this report, and key improvements and considerations for the future are identified.