**Final Project:**

Using Predictive Analytics to Forecast Service Requests

Ambrose M Malagon Cordero

Bellevue University

DSC 630 – Predictive Analytics

Thursday August 6, 2020

**Executive Summary**

The foundation of a vibrant city life rests on well-maintained capital infrastructure. The City of Oakland (government) is responsible for maintaining the city’s infrastructure and functionality. Stewarding these assets is one of a municipality’s core responsibilities and one of its greatest opportunities.

The purpose of this project is to apply the concepts learned throughout this course into a Workforce Management scenario: predicting the total number of Service Requests in a given timespan by using historical data. By establishing a trend of when and where Service Requests might be generated, the City of Oakland can use this information to plan ahead in terms of staffing and budget to handle these requests accordingly.

This project is divided into the following segments:

1. Introduction and Business Problem
2. Data Exploration and Preparation
3. Model
4. Conclusions

Three models were built and assessed for viability in order to attain the desired results: the predicted counts of service requests. While one of the goals for this project was to build an ensemble model, it ultimately did not happen. The insights gained from working on this project will potentially help us gain a better understanding on handling similar types of projects in the future.

**Abstract**

A Workforce Analyst at my current work location was in the process of transitioning her processes from Excel over to R. She had asked if I could help since I was learning R as a result of this program, however I had little to no understanding of what she was trying to accomplish. Now that I’ve progressed further into the program, I wanted to revisit a version of this request with hopes of assisting this coworker.

The purpose of this project is to utilize the City of Oakland’s data repository on Service Requests for city infrastructure and maintenance issues reported by citizens to create a Machine Learning Model that can predict on average the total number of Service Requests generated within a specified timeframe.

We tried to closely adhere to the Cross-Industry Standard for Data Mining for this project. Once the data was collected and imported into our development environment, we proceeded to select our variables that were best suited to create our models. We built 3 models: a Poisson Regression Model, a Negative Binomial Distribution and a Random Forest Machine Learning algorithm in search of the best way to predict our results.

The payoff will be to demonstrate an understanding of the concepts presented in this course. I also hope to use the knowledge and experience from working with this project to return back to my co-worker and assist her with her initial inquiry.

**Introduction**

Workforce Management (WFM) is an institutional process that maximizes performance levels and competency for an organization. It includes all of the activities to maintain a productive workforce by:

* Forecasting of workload and required staff
* Involvement of employees into the scheduling process
* Management of working times and accounts

A key aspect of workforce management is scheduling by establishing likely demand by analyzing historical data. Current and future staffing requirements, short-term peak loads, availabilities, holidays, budget allowances, skills, labor law-related restrictions, as well as wage and contractual terms have to be integrated into the planning process to guarantee optimal staff deployment.

Workforce Management is embedded in planning infrastructure and maintenance of a community such as the city of Oakland. This project will be an attempt at analyzing the historical data for Service Requests, understand its nature, and use it to create a model that will allow us to forecast future request volume. We hope that the learnings and pitfalls of this project will provide insights into the inner-workings of how forecasting works in Workforce Management.

**Methods**

**Step I – Data Acquisition.**

* Service requests received by the Oakland Call Center (OAK 311) - Provided by the City of Oakland. <https://data.oaklandnet.com/Infrastructure/Service-requests-received-by-the-Oakland-Call-Cent/quth-gb8e>.

For this project, we want to predict the total number of Service Requests on for a specific timeframe which we will determine at the time of analysis. We download the data, load the dataset into Python, and begin our initial data transformation process. As of June 25, 2020, this dataset contained 713,569 records and continues to grow with each day.

**Step 2 – Data Exploration and Preparation.**

Provided below is the list of fields and their descriptions:

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| REQUESTID | The service request number. |
| DATETIMEINIT | The Date that the request was initiated. |
| SOURCE | Indicates if the request was received via SeeClickFix, via the Report A Problem website at www.oaklandpw.com, or via (Phone or Email). |
| DESCRIPTION | Type of issue. |
| REQCATEGORY | Request Category |
| REQADDRESS | Request Address |
| STATUS | The status as of date of upload |
| REFERREDTO | If the status is Referred, this shows who it was referred to. |
| DATETIMECLOSED | The Date that the request was closed |
| SRX | Coordinates of the issue. |
| SRY | Coordinates of the issue. |
| COUNCILDISTRICT | City Council District of the REQADDRESS. |
| BEAT | Police Beat |
| PROBADDRESS | Prob Address. |
| City | City |
| State | State |

An initial look at the variables provided by the dataset as they are:





The Summarized data provides insights on the following categorical data:

* There are 8 unique ways to submit a Service Request. The top value is “SeeClickFix”, which is the Website/App citizens can use to submit Service requests.
* There are 31 unique Request Categories. The top being ILLDUMP (Illegal Dumping).
* We have duplicate entries and misspellings for the following records:
  + COUNCILDISTRICT
  + BEAT
  + STATUS
* City and State are one unique value respectively.

We are selecting all years with the exception of 2009 and 2020 as these years do are incomplete, (2009 data starts in July while 2020 is still ongoing).

**Target Variable:**

|  |  |  |
| --- | --- | --- |
| Target Variable Name | Target Variable Type | Target Variable Values |
| Count of Request ID | Continuous | Count of Submitted Service Requests. |

Our initial goal for this project is to determine the total number of counts of service requests within a specified time frame. After looking at several possible combinations, we opted to look at the following grouping:

* Council District > Beat > Request Category > Status >Year the request was created > Month the request was created > Week Number the request was created.

We end with the resulting Data Frame:



The following plots below help illustrate how the number of service requests grows with each passing year, the total variation with each passing year and the distribution of the counts:







|  |  |  |
| --- | --- | --- |
| **Year** | **Total Requests** | **Percentage Growth** |
| 2010 | 33,647 | - |
| 2011 | 37,995 | 11% |
| 2012 | 47,294 | 20% |
| 2013 | 56,888 | 17% |
| 2014 | 61,826 | 8% |
| 2015 | 66,889 | 8% |
| 2016 | 75,932 | 12% |
| 2017 | 80,740 | 6% |
| 2018 | 77,851 | -4% |
| 2019 | 110,928 | 30% |
|  |  | 12% |

The total count of service requests for the City of Oakland grows with each passing year, in this 10-year timespan an average of 12%. The outlier here would be 2018, which saw an overall decrease in terms of counts from 2017. This raises questions as to what events or other factors we are not accounting for that have an effect on counts.

**Explanatory Variables:**

Provided below is the Pearson’s correlation for all of the numerically encoded variables in R:



Next, Spearman’s correlation in Python for all numeric/categorical data points:



Provided below is the contingency table between the City Council and Police Beat variables:



“NR1” and “nc” under the City Council and Police Beat respectively stand for missing/values that were not captured or entered at the time the Service Request was issued. Ultimately, we are trying to pinpoint accurate numbers using real locations, so we are going remove records that have these values.

Next, we look at the contingency table between Request Type and Status:



We quickly notice how we only have 5 requests total under PENDING status and these fall under the ELECTRICAL request category. We will also omit the following request categories for overall small counts to the point of becoming outliers:

* GIS
* LAB
* SURVEY
* OPD
* VEGCONTR

**Step 3 – Predictive Models.**

**1st Model – Poisson Regression Model:**

The initial project goal is to predict overall Service Request counts, our first attempt at a model will be to assume a Poisson Distribution. Provided below are the model parameters as in R:









Aggregated Counts – Actuals vs Predicted:



To assess the quality of the model, we need to test for overdispersion. Overdispersion describes the observation that variation is higher than would be expected. Some distributions

do not have a parameter to fit variability of the observation. We can test for this in R by using the “dispersiontest” function of the AER library:



In this case, our dispersion value is greater than 1 so this a clear indication of overdispersion.

**2nd Model – Negative Binomial Distribution Model:**

We next try our hand at a Negative Binomial Regression, which is often considered the next step when dealing with over-dispersed Poisson Regression models:







Looking at the model’s overall dispersion using Python:





Aggregated Counts – Actuals vs Predicted:



**3rd Model – Random Forest.**

Another way to predict counts is via the use of the Random Forest Algorithm. Unlike Poisson and the Negative Binomial distributions, Random Forest is a ‘black box’ algorithm, meaning we do not have key metrics or any other information that allows us to infer how it arrives at its predictions.

Because of Random Forest’s complexity and requirements, we used a small sample size of 10,000 observations that were divided to train and validate the model. The predictions (red-dotted) appear to be worse than the previous algorithms. However, the model in itself could potentially improve if provided with more data.



Aggregated Counts – Actuals vs Predicted:



While the aggregate totals look somewhat accurate, the individual predictions are only 58% accurate:



**Discussion/Conclusion**

The importance of forecasting when and how demand cannot be understated. Using Historical data to forecast future events to meet future demands from customers and clients can is not just critical to workplace environments like a Call Centers but also cities that need to manage their resources in the most effective manner. One item that does come to mind is how entities such as City Governance and Workforce/Labor departments are coming about their forecasts: specifically, which tools and techniques they are implementing, and how often are the models that make said predictions revisited/revised.

Topic recaps aside, there were some learning lessons with this project. First off, I don’t believe I had the right data, or rather, I chose the wrong research question. Having worked with this data over the last couple of weeks has dawned upon me this realization that data itself was ripe for a classification problem. However, I fear this would’ve been the easy way out in terms of working with this data and so I stood the course of my decision.

One thing that could’ve potentially enhanced this project would’ve been the addition of “external event data” surrounding these Service Requests. Referencing back to my days as a Call Center Representative, every time there was a major announcement made by Disney, we were in for an influx of calls. In my preliminary analysis, I was looking for a correlation between Holidays and Service Requests but that did not pan out as I had hopped. At one point, I asked myself if this data even existed to begin with. After a quick search in the City of Oakland’s data repository, I found nothing and gave up.

**Acknowledgements**

* Krysten Riker – Fellow co-worker who sent me down this path by asking how to use Linear Regression in R. I hope to present this project to her in hopes that it will allow her provide her with insights.
* Amirfarrokh Iranitalab – Thank you for being my course instructor and for answering all my questions – Even the redundant ones.
* Natallia Casey – Thank you for providing your peer review and providing suggestions.
* Alan Danque – For providing his feedback for this project.

**References**

[1] Margaret Rouse. workforce management (WFM). Whatis.com. Article. Retrieved Sunday June 28, 2020 from <https://searchhrsoftware.techtarget.com/definition/workforce-management>.

[2] Statistics How To. Poisson Regression / Regression of Counts: Definition. Article. Retrieved Sunday June 28, 2020 from <https://www.statisticshowto.com/poisson-regression/>.

[3] Carsten F. Dormann. Overdispersion, and how to deal with it in R and JAGS. GitHub. PDF. Published December 7, 2016. Retrieved Friday August 7, 2020 from <https://biometry.github.io/APES/LectureNotes/2016-JAGS/Overdispersion/OverdispersionJAGS.pdf>.