Machine Learning Engineer Nanodegree

Capstone Proposal

Brindha Sivashanmugam

August 3rd 2018

EMERGENCY – 911 CALLS PREDICTION

DOMAIN BACKGROUND

We all know how significant the role of a 911 employee is. They are the ones who address our emergencies. The team works 24x7 non-stop. Any challenges they face in their job will directly affect the public in their life or death moment. Let us look at what happens when there is a flaw on their side, how it affects the public, and the challenges they face in their day-to-day job.

From the Headlines: (How a flaw in 911 affects the Public)

Longer Response Time: Tragic death of Shanell Anderson in Atlanta because emergency responders arrived too late to save Shanell Anderson from drowning in her SUV. *(Reference 1)*

Equipment Outage: A cooling unit low on refrigerant, caused Montgomery County, MD 's 911 system to crash for nearly two hours on the night of July 10 2016, causing about 100 callers to get busy signals when they tried to report emergencies. Delays caused deaths of a 40-year-old dialysis patient and a 91-year-old woman. *(Reference 2)*

Human error: The death of Kokomo resident Tammy Ford on July 1 2015, due to operator error. In that case, a dispatcher sent first responders to the wrong address. *(Reference 3)*

Challenges Faced by 911 Operators and Dispatchers:

- Most of the 911 employees work in 12 hour shifts in a hectic, high stress environment, not able to plan their leisure hours or vacation. They are humans too. If they could know how to plan this, they could be stress free, and operator error will no longer be an issue.
- Not able to know when they could schedule a downtime to maintain their equipment, which is very crucial, so that the equipment would to be in good shape, ready for a dire emergency.
- Not able to predict the prime location where most of the calls are expected on a particular day and time. If they had known that, they could rotate employees accordingly and cover the emergencies in a shorter response time.

PROBLEM STATEMENT

If the response time of the 911 team is minimized, lot of lives could be saved. Response time can be minimized if there are surplus amount of equipment and human resources ready to serve emergency, which is practically not possible, as it would consume a lot of money.

So, the alternate way is to effectively distribute available resources. This could be possible only when they know in advance about how many 911 calls are expected. As this is emergency call, no one knows what is expected until it actually happens. Here is where Machine Learning comes in rescue of the rescue team.

There are 3 major categories of emergencies – EMS, Fire and Traffic. The number of calls expected in each category (EMS, Fire, Traffic) on a given day, time frame and region need to be found.

DATASETS AND INPUTS

I am using "Emergency - 911 calls" dataset for this problem. This dataset was provided by montcoalert.org , and donated to Kaggle by Mike Chirico. This dataset contains more than 2 years history of 911 calls handled by Montgomery county, Pennsylvania. It contains the emergency category, location and GPS coordinates of the victim. As it is very informative, and the dataset has enough values to do the analysis, I thought I could work on this for my Machine Learning Capstone Project. I have downloaded the dataset from Kaggle Datasets *(link given in reference section – Reference 4)*.

The dataset has 326425 records and 9 columns. The columns are described below:

S.No	Column Name	Column Description
1.	"lat"	Latitude
2.	"lng"	Longitude
3.	"desc"	Description of emergency
4.	"zip"	ZIP code
5.	"title"	Title of emergency
6.	"timeStamp"	Date and time of the call
7.	"twp"	Town
8.	"addr"	Address
9.	"e"	This column is filled with a constant 1 for all rows

The below is a sample row from the dataset.

```
data.head(1)
```

	lat	lng	desc	zip	title	timeStamp	twp	addr	e
0	40.297876	-75.581294	REINDEER CT & DEAD END; NEW HANOVER; Station ...	19525.0	EMS: BACK PAINS/INJURY	12/10/2015 17:10	NEW HANOVER	REINDEER CT & DEAD END	1

SOLUTION STATEMENT

With the available timestamp and location information, regression model would be developed which predicts the number of calls in each emergency category on a given day, time frame and location.

Gradient Boosting Regressor model *(sklearn.ensemble.GradientBoostingRegressor)* will be developed to predict the number of calls. Also, XGBoost *(xgboost)* will be implemented for predicting the number of calls. Evaluation is done based on R2 score, which will be explained under "Evaluation Metrics" section. Performance of both the algorithms will be compared.

BENCHMARK MODEL

Linear regression model *(sklearn.linear_model.LinearRegression)* will be implemented to predict the number of 911 calls expected in each emergency category on a given day, time window and location. This model will be evaluated using R2 score as the evaluation metric. This model will be used as a benchmark model.

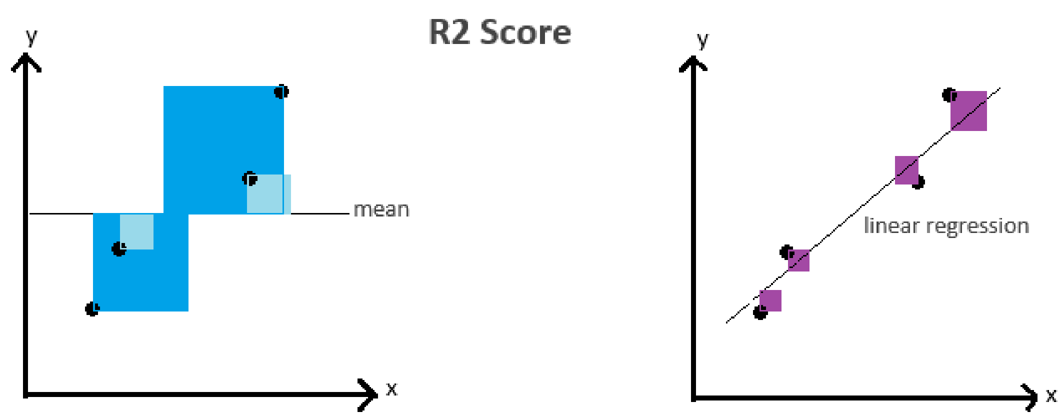
EVALUATION METRICS

R2 score is used for evaluating all the models developed in this project. R2 score is a very common regression metric. It is based on comparing our model to the simplest possible model (the mean of the data points).

The simplest possible model is the average of all data points, represented as a horizontal line that goes through them. The sum of squares of residuals of this simple model is calculated (SS_{mean}). And, the sum of squares of residuals of our model, for example, linear regression model is calculated (SS_{linear_reg}). Then, R2 score would be defined as:

$$\text{R2 score} = 1 - (SS_{linear\_reg} / SS_{mean})$$

This will be clearly given in the below plot.



The areas of the blue square represent the squared residuals with respect to the mean.

The areas of the purple square represent the squared residuals with respect to the linear regression.

$$\text{R2 score} = 1 - (SS_{linear\_reg} / SS_{mean})$$

The value of R2 score lies between 0 and 1. In order for a model to be good, the R2 score should be closer to 1.

PROJECT DESIGN

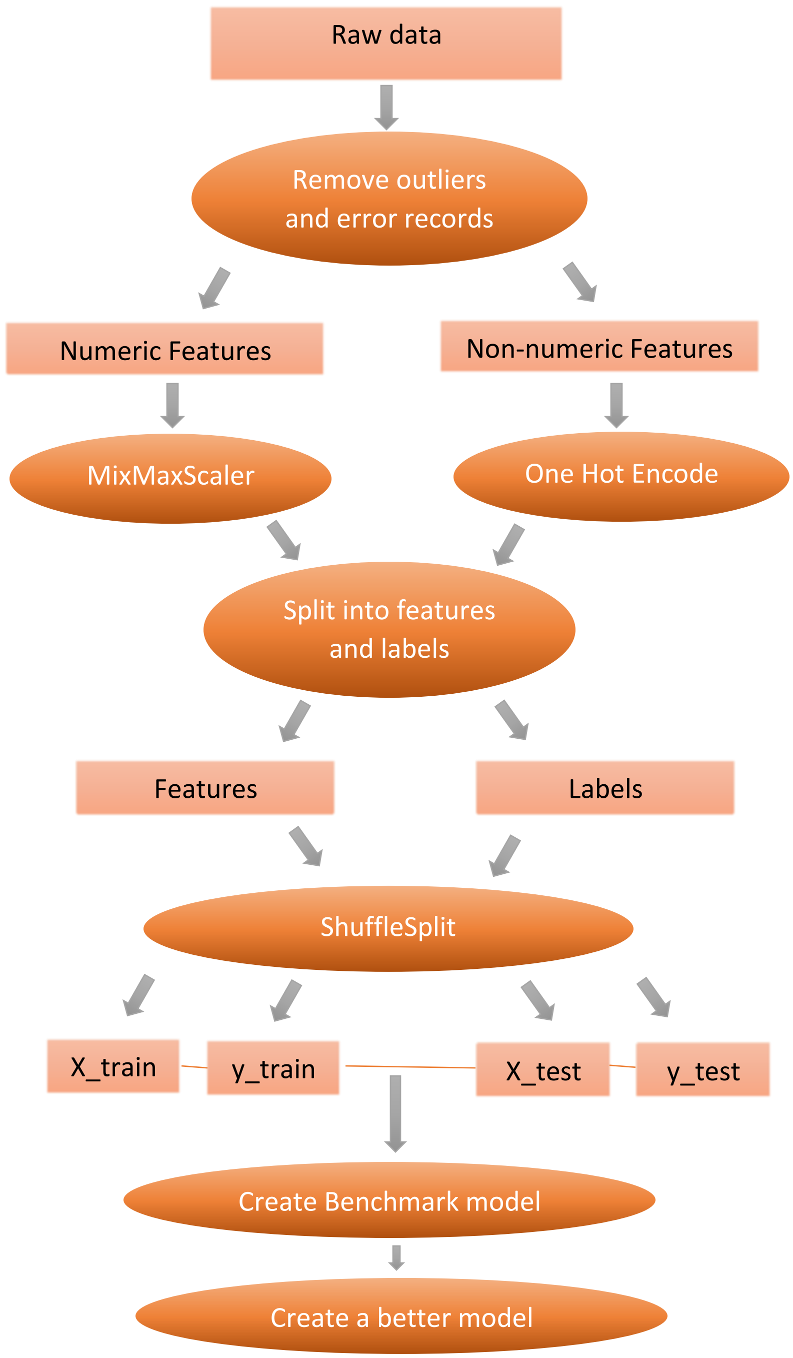


Figure: Steps involved in project design

Data cleanup

Analyzing the town names and position of the latitude longitude to check for any geographic outliers and removing them. Records which are wrongly entered by human error will be analyzed and removed if any. Then, the dataset will be checked for missing values in any of the features, and those will be cleaned. Then, the features that are suitable to this context and problem will be identified and only those will be retained.

Data Preprocessing

Data needs to be scaled before applying learning algorithms on them, so each feature will be treated equally by the learning algorithm. So minmax scaler will be applied on all the numeric features. The other non-numeric features need to be converted into numeric before applying learning algorithm. So, one hot encoding will be done on the non-numeric features.

Split the Dataset

The dataset will be split for features and labels. The number of calls per Emergency title (EMS, Fire, Traffic) will be the labels. Then, the dataset will be shuffle split for training and testing. 80% of the data will be used for training, and 20% will be used for testing.

Setting the Benchmark

A linear regression model will be developed and applied, and the R2 score of the testing set will be recorded. This will be used as the benchmark score.

Developing a better model

Gradient boosting regressor and xgboost regressor will be developed. Corresponding R2 score of the testing set will be recorded. The performance of both the algorithms will be compared, and discussed.

REFERENCES

Reference 1: https://www.ravemobilesafety.com/blog/911-system-failures

Reference 2: http://www.kokomotribune.com/news/dispatchers-discuss-challenges-of-the-job/article_85753cb0-379a-11e5-8a06-b74ef6ca24dc.html

Reference 3: https://www.washingtonpost.com/local/md-politics/montgomery-officials-try-to-explain-911-outage-that-should-never-happen/2016/08/02/798c77d8-58d2-11e6-9767-f6c947fd0cb8_story.html?utm_term=.cb08532efcaf

Reference 4: https://www.kaggle.com/mchirico/montcoalert