**1. Tweet classification is a project from Kaggle.**

**Data given :** Keywords,location, text and target variable (0 or 1) for each tweet.

**To classify :** Target variable is a binary classifier to classify if a tweet is about a real disaster or not.

**Approach :**

Used regex functions in python using the 're' package and removed

* numbers
* punctuations
* extra space
* non- printable words like \n, \r, \t and such.
* urls
* converted the text to lower for consistency.

Next, used a language model, in this example fasttext from gensim to train the text and then apply logistic regression to classify the labels. Metrics used to measure goodness of fit: Accuracy, precision\_score, recall\_score and cross\_val\_score

**2. Anomaly detection for data ingestion**

**Data given :** date, and record count

**To classify :** Whether a record is an anomaly or not.

**Approach :**

This is a time dependent problem, hence used time series model.

Initially used ARIMA models, but due to scalibility issues, had to transition to a different approach. Next, used Prophet model developed by facebook.

Used 90 day time period as training data, predicted the values for the next 1 week. When new data is loaded, compare the predicted data with the observed data point, and classify if the record is an anomaly or not. Define anomaly: If an observed value of record falls outside the predicted threshold, then it is classified as an anomaly.

**Results :** Using prophet model,

* was able to scale the model for different data trends
* added a holiday effect
* overall reduced false positive rate of anomalies by 32%
* reduced number of anomlies resulted in reduced amount of time spent in resolving anomalies by 25%.

![image](https://user-images.githubusercontent.com/12262825/128132756-2a489b6a-149c-4336-ba6a-ccfaefe8a6ff.png)