KHARAGPUR DATA ANALYTICS GROUP

Presents

KHARAGPUR DATA SCIENCE

HACKATHON

Sponsored By

redBus

Team : D-WING (DTC4872)

Members :

- Vishal Kumar
- Soumyo Bhattacharjee
- Harsh Sharma
- Shivam Sood

Contents

Problem Statement

Dataset Description

Data Pre-processing : Feature Engineering

Data Pre-processing : Grouping Instances

Data Pre-processing : Feature Distribution

Curse of Dimensionality : PCA

Algorithm

Confidence Score & Leader-Follower

Intuition & Generalization

Price Setter & Follower Detection :

- Problem Statement

redBus is an online platform where bus operators offer their services and sell seats. These bus operators vary from single service operators to ones which have scores of services.

Pricing is a complex decision for Bus Operators, who continuously adjust the prices seeing the demand and supply signals. But not all operators have the same level of visibility on these signals or the capability to price effectively. Given the pricing data, we had to reveal the market dynamics.

Hence, our challenge was to find out which services is/are pricing independently (the price leaders) and which services are the followers (and following whom), given the history of prices set by the operators.

Price Setter & Follower Detection :

- Dataset

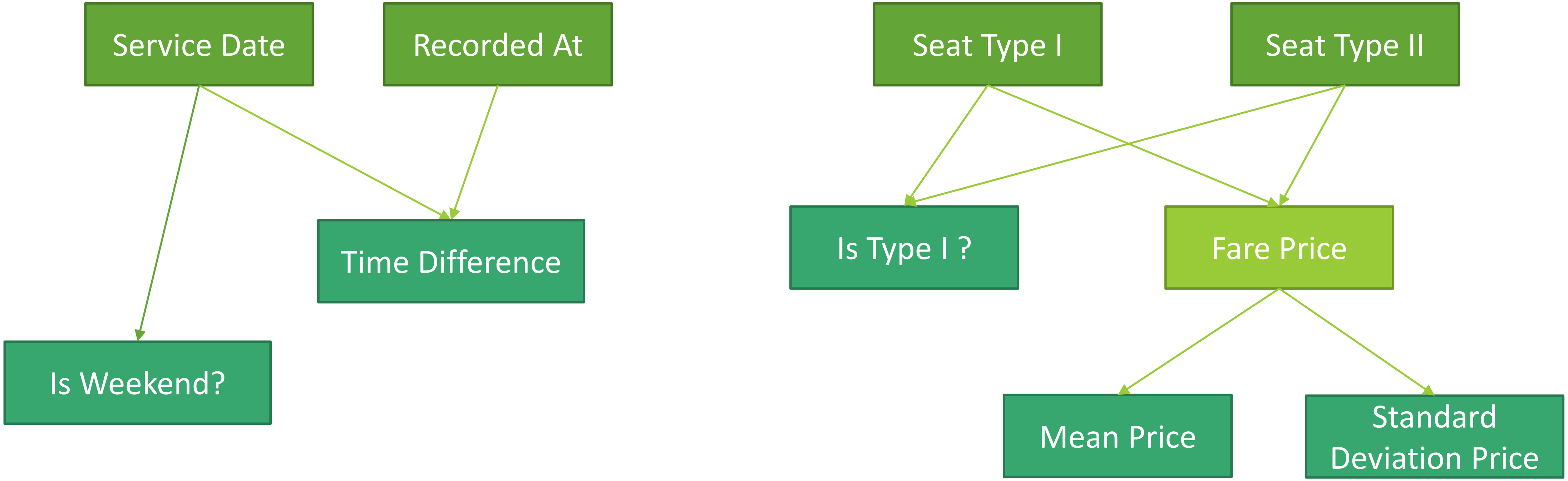
The following fields are data fields are available (as provided by the problem setter) –

- Seat Fare Type I : Prices of all categories of this type of for a particular trip
- Seat Fare Type II : Price of all categories of this type of seat for a particular trip
- Bus ID : ID assigned to a particular bus
- Service Date : The date of journey
- Recorded At : The time at which prices were recorded

With the given fields, the data was analysed to determine the features which could be used for the given problem.

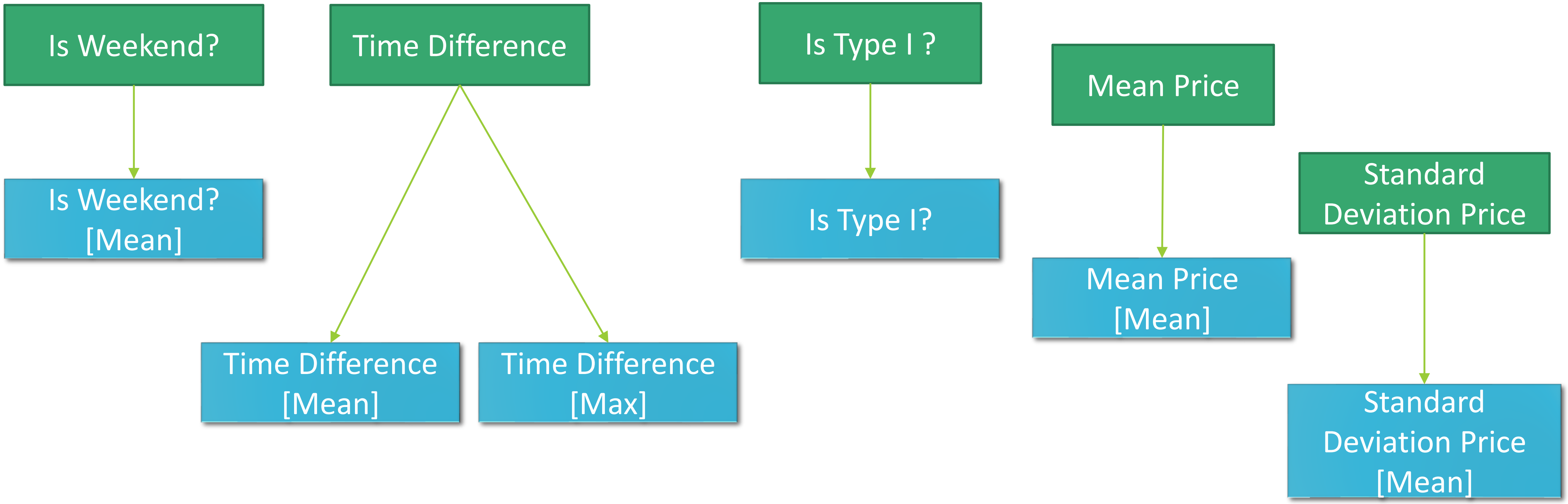
Data Pre-processing :

- The basic idea was to recognize similarity, and numeric data is preferred for any ML based approach.



Data Pre-processing :

- The features were then Grouped by each bus type [117 Unique Bus IDs]



*Mean / Max specifies criteria taken to group the 117 IDs

Data Pre-processing :

- The distributions of features plot [For 117 Unique Bus IDs]

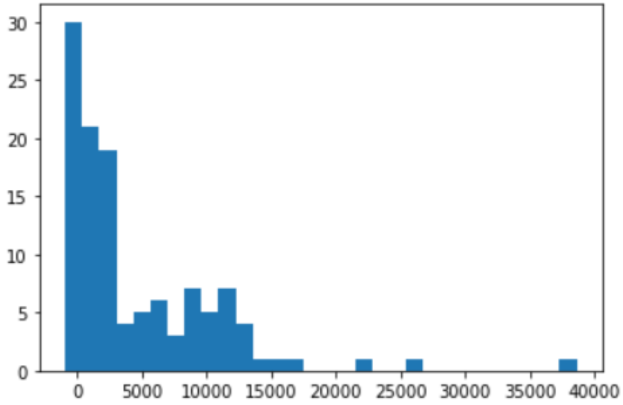
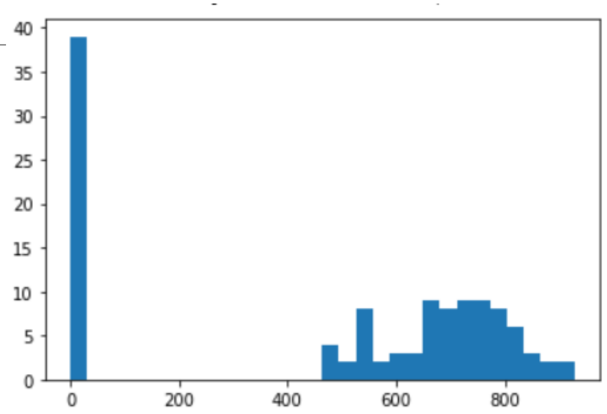
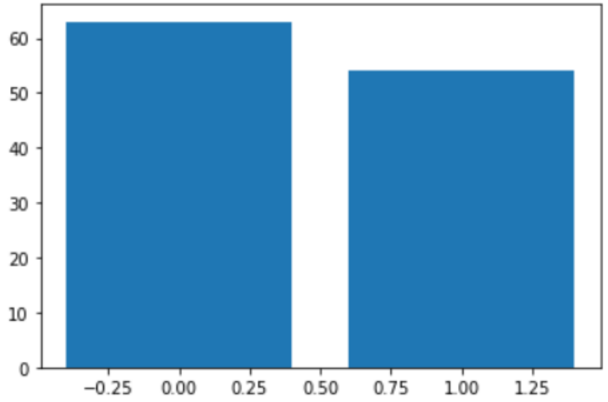


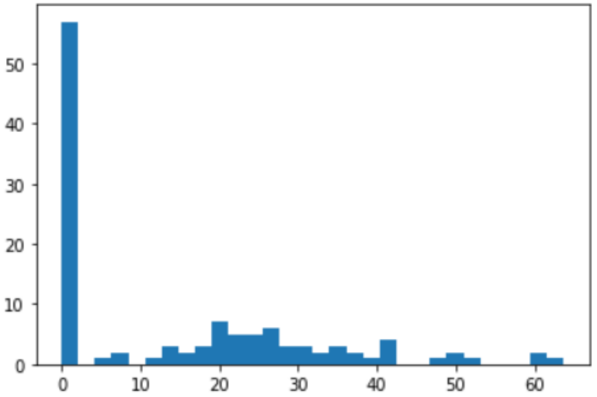
Fig (a): Absolute Time Difference[mean]



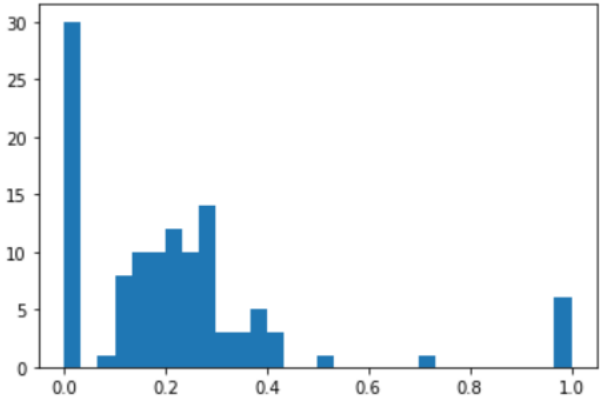
Fig(b): Mean Fare [mean]



Fig(c): Is Type1



Fig(d): Fare Deviation[mean]



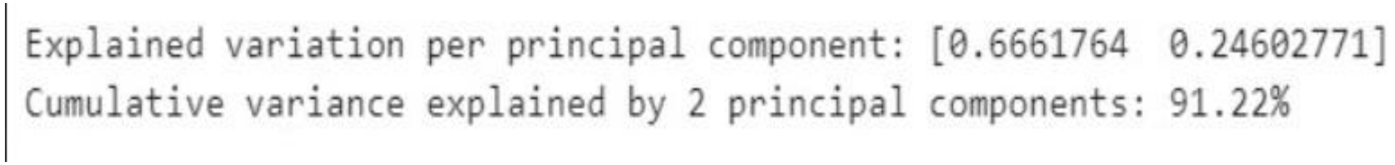
Fig(e): Is weekend[mean]

*Data was normalized

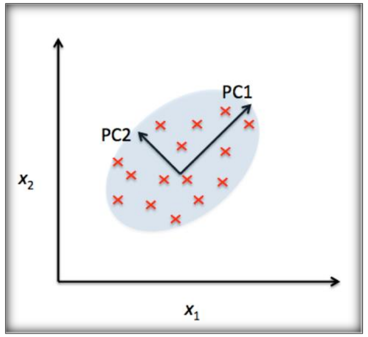
Curse of Dimensionality :

- The clustering algorithms suffer when trying to cluster high dimensional data. To overcome this issue Principal Component Analysis (PCA)

The 6 dimensional data was reduced down to 2 to obtain optimal performance.



```
Explained variation per principal component: [0.6661764  0.24602771]
Cumulative variance explained by 2 principal components: 91.22%
```



The 2 dimensions effectively capture 91.22% variance.

Algorithm :

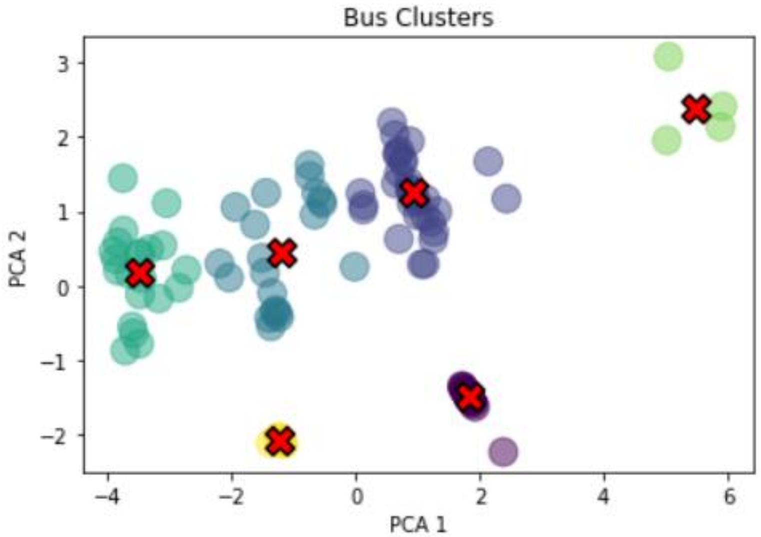
- The main algorithm used is K-Means Clustering : In 2 stages

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters.

The k-means clustering algorithm mainly performs two tasks:
(i) Determines the best value for K center points or centroids by an iterative process.
(ii) Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

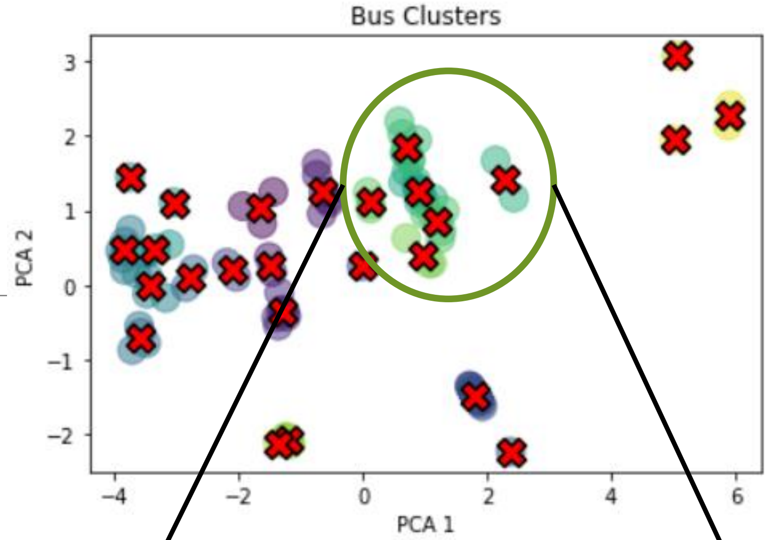
To obtain a good cluster number : K , we use silhouette analysis.

Silhouette analysis can be used to study the separation distance between the resulting clusters. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually

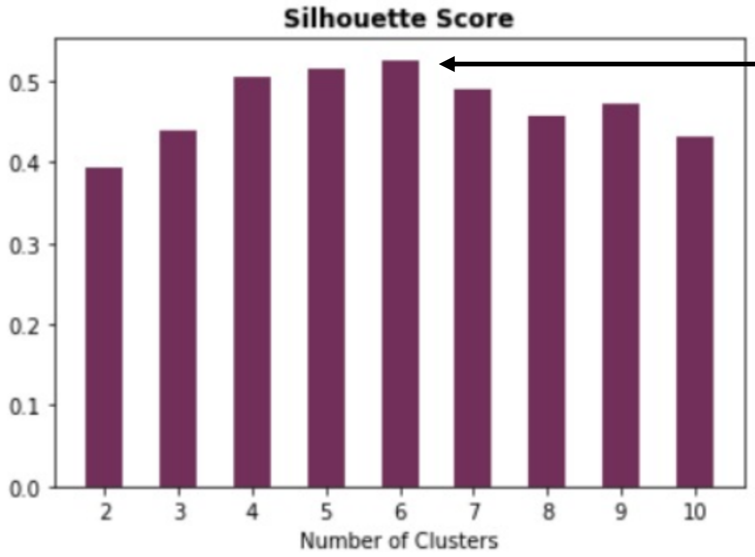


Fig(a) : Primary Clusters

Secondary Clustering

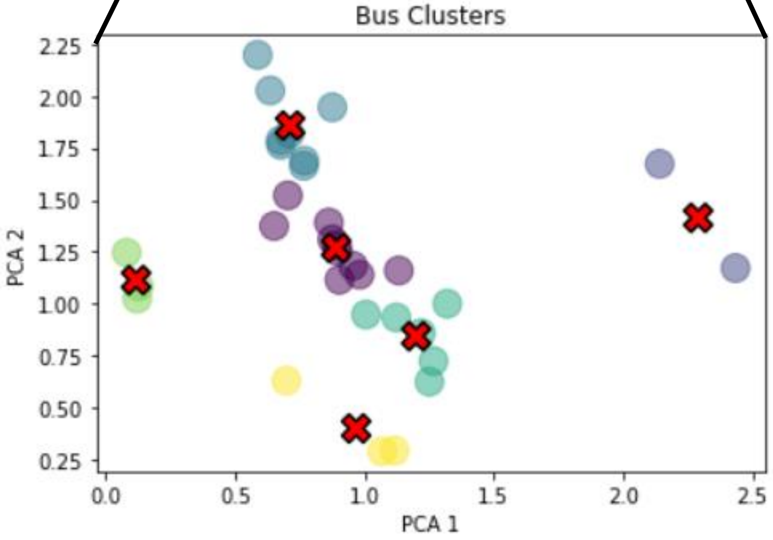


Fig(b): Secondary clusters



Best Choice : 6

Fig(c): Silhouette Score plot for a Secondary Cluster



Fig(d): Cluster plot inside a Primary cluster

Confidence Score & Leader-Follower :

- The confidence score was determined for each sub-cluster (after secondary clustering), and a simple ranking algorithm was followed.

The formula used to generate the confidence metric is shown.

$$\frac{\frac{1}{||x-c_i||}}{\frac{1}{||x-c_i||}+\sum_j \frac{1}{||x-c_j||}}$$

The intuition is to obtain a higher score for points close to the centroid.

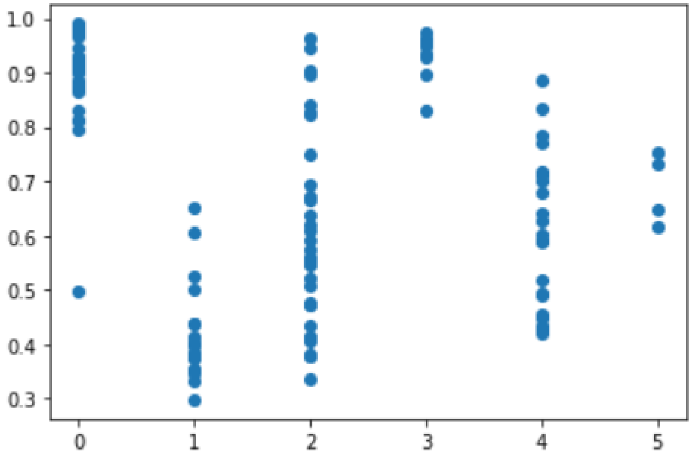


Fig : Confidence scores within a cluster

The ranking algorithm is devised on the fundamental logic that points close are the most similar. The point with the highest confidence metrics are the leaders of the cluster. While the remaining scores are sorted in a non-increasing manner, and the points adjacent to each other are designated as “follows” and “is followed by”. The confidence metrics are renormalized point by point basis.

Intuition & Generalization :

- The problem required a method to find similarity and rank instances. An unsupervised approach was required.
- Feature engineering was done so as to produce numeric features which were thought to be relevant to dynamic pricing.
- A two staged clustering was used to find tight clusters. The clustering based approach requires a low dimensional data.
- A confidence metric and a simple ranking algorithm was to be determined.

The above basic steps are common for any problem which requires a similarity based classification, particularly on unlabelled data. The feature engineering is completely dependant on the problem at hand. The ranking algorithm is kept simple, with scope for further improvements in this part of the approach.

Thank You !