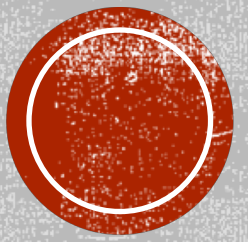
CIBC DATA STUDIO

CIBC MACHINE INTELLIGENCE HACKATHON

Fan(Peter) Chen Xu Jia

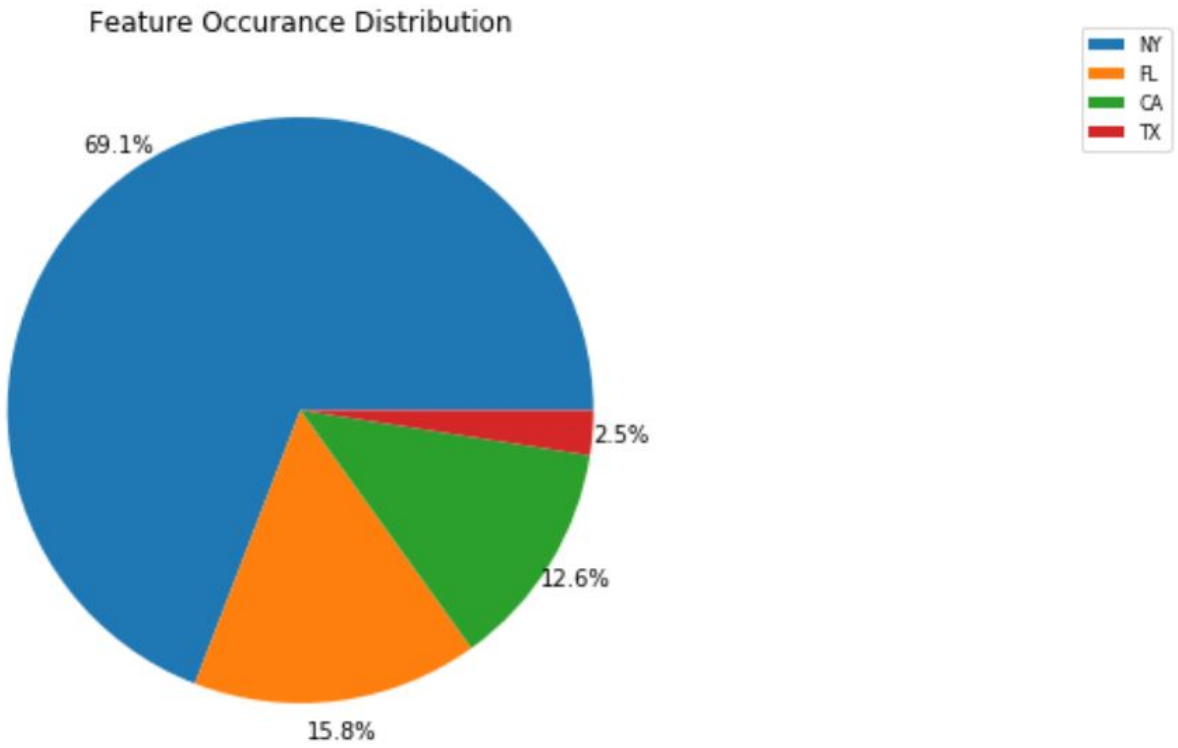


PROBLEM STATEMENT

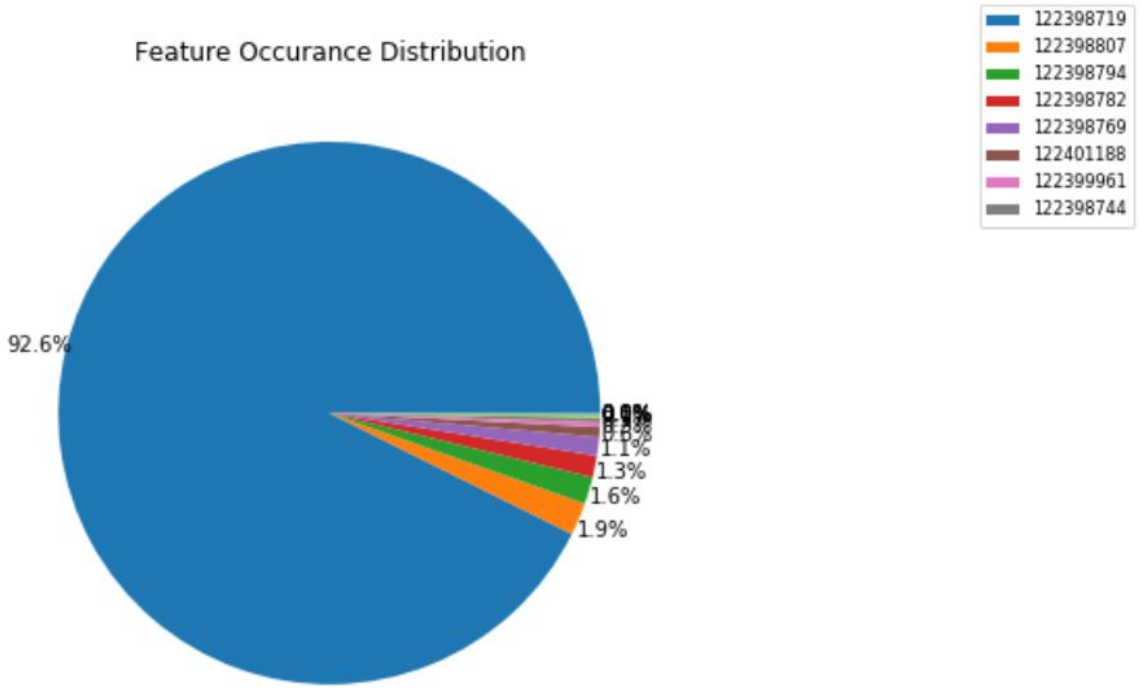
- Teams will be provided an insurance claims data set and will be tasked with finding fraudulent claim records and medical providers. The claims data will contain 1 year of medical claims. The data is based on a sample of actual claims data from a US healthcare company.
- The claims data format is a comma-separated file with the following columns
  1) Patient Family ID
  2) Patient Family Member ID
  3) Provider ID
  4) Provider Type
  5) State Code
  6) Date of Service
  7) Medical Procedure Code
  8) Dollar Amount of Claim

ANALYZE DATA

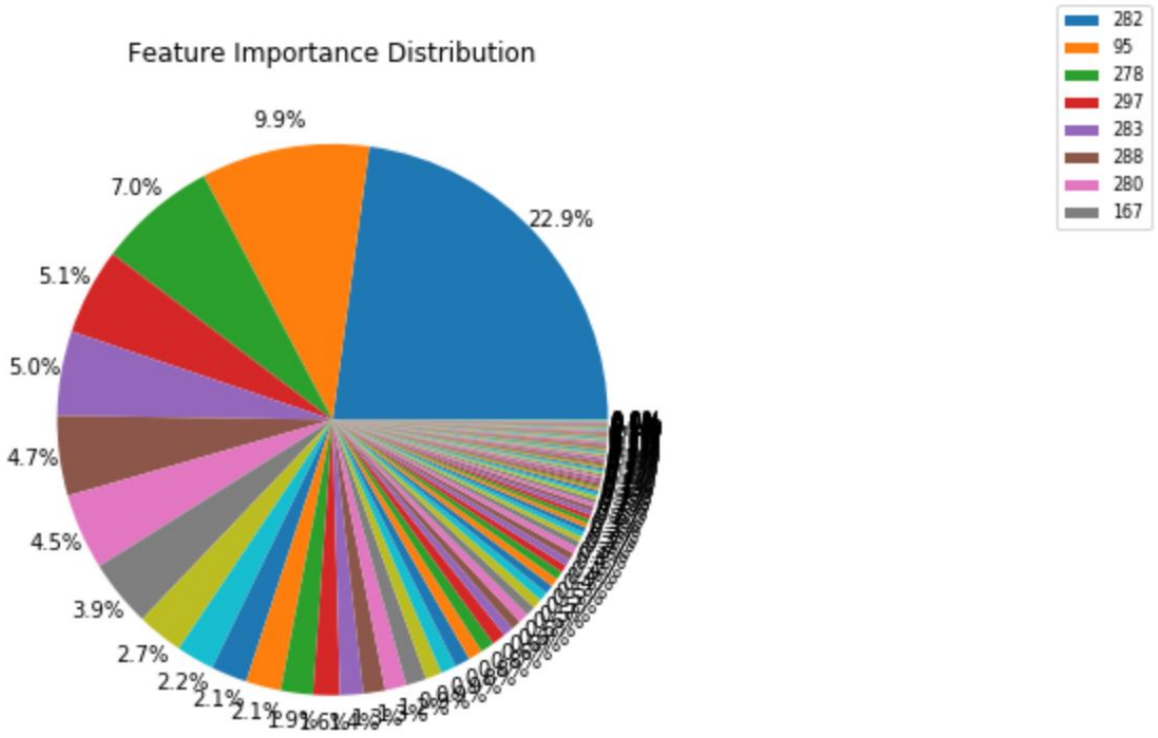
- First read origin data and visualized to see the distribution of every columns to have a basic view of the data.



- State Code



- Provide Type



- Medical Procedure for most Common 8

FEATURE ENGINEERING

- For File1, We first divided features in 4 types:

Timely, Itself, Common medical procedure, Patient units.

For timely features, we have **family numbers per month**, **dollar amount per month** and **visit amounts per month**, and for each timely feature, we can divide into 4 sub features with **minimum**, **maximum**, **average** and **standard deviation** of these features, like showing below:

```
['ProviderID', 'count(month_n)', 'max_month_amount', 'min_month_amount',
 'std_month_amount', 'avg_month_amount', 'max_month_family_num',
 'min_month_family_num', 'std_month_family_num', 'avg_month_family_num',
 'max_month_visit_num', 'min_month_visit_num', 'std_month_visit_num',
```

FEATURE ENGINEERING

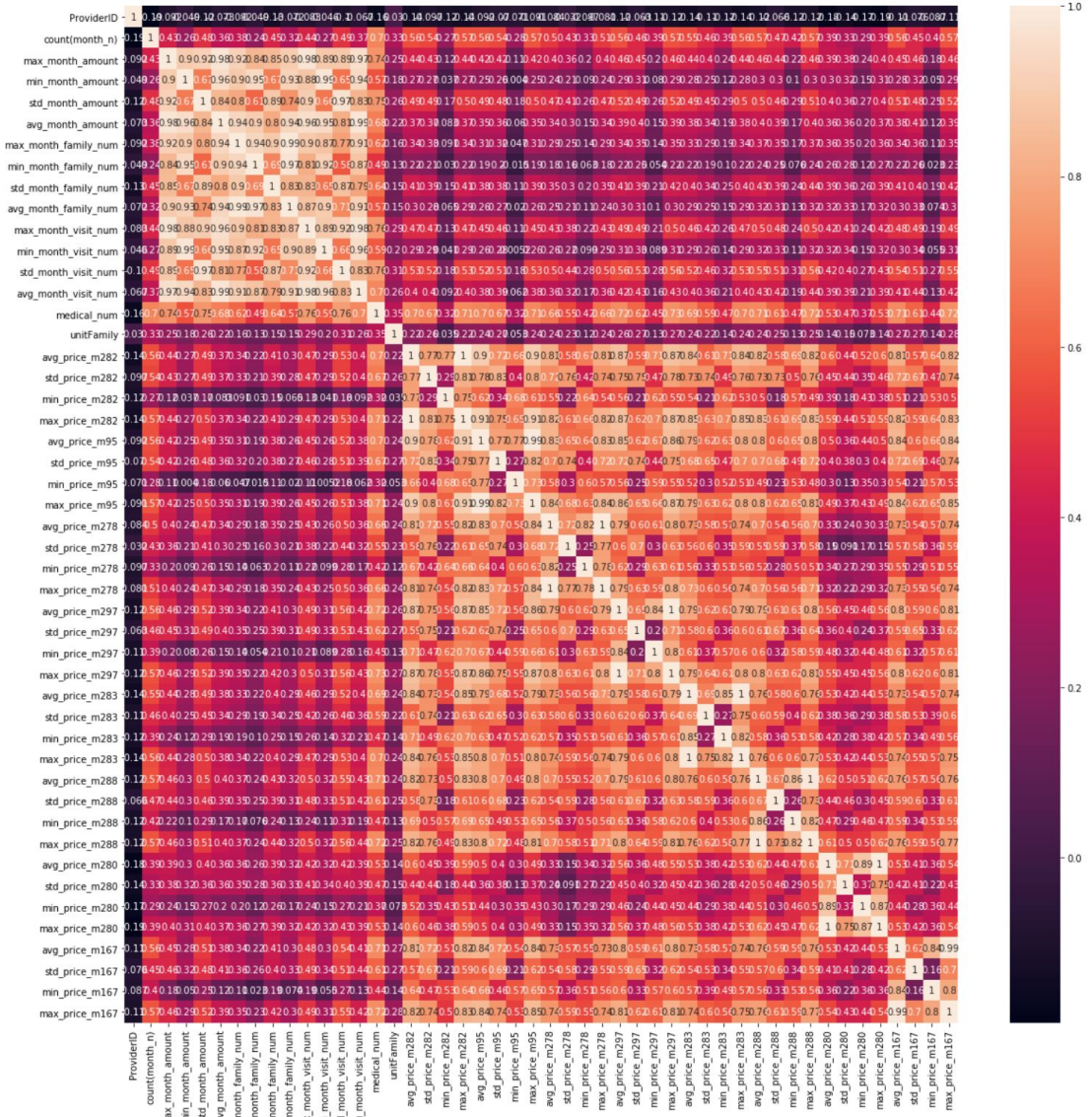
- The Provide Type, state code and medical procedure are features can be added to the feature dataframe directly.
- For the medical procedure types, we selected 8 most common medical procedures by frequency, and we also divided each into 4 sub features as showed below:

```
'std_price_m282', 'min_price_m282', 'max_price_m282', 'avg_price_m95',
'std_price_m95', 'min_price_m95', 'max_price_m95', 'avg_price_m278',
'std_price_m278', 'min_price_m278', 'max_price_m278', 'avg_price_m297',
'std_price_m297', 'min_price_m297', 'max_price_m297', 'avg_price_m283',
'std_price_m283', 'min_price_m283', 'max_price_m283', 'avg_price_m288',
'std_price_m288', 'min_price_m288', 'max_price_m288', 'avg_price_m280',
'std_price_m280', 'min_price_m280', 'max_price_m280', 'avg_price_m167',
```

FEATURE ENGINEERING

- Finally we have Patient unit features that describe how many people in a common family undertook the same medical procedure, which may possibly relate to a fraud in an insurance fraud.

- Then we plot heat map to show the relationships of selected features.



- Heat-map relationship

FEATURE ENGINEERING

- For File2, we did feature engineering mostly like what we do for file1, but we set **familyID**, **family member** number, **provide type** number individual features to fit the requirements.

FEATURE NORMALIZING

- After selecting and adding features manually, we do feature normalizing to make sure every features can equally affect the clustering result.

- To do this, we use Z-score method to ensure all features follow the Normal distribution.

- feature_scale = new_feature.apply(lambda x: (x - np.mean(x)) / (np.std(x)))

PCA ANALYZE

- Because we already do the normalizing to fit the data to a scale of normal distribution, now we can apply PCA algorithm to find out which dimension can enlarge the deviation of features most.

```
In [15]: from sklearn.decomposition import PCA

         pca = PCA(n_components=12)
         pca.fit(feature_scale)

Out[15]: PCA(copy=True, iterated_power='auto', n_components=12, random_state=None,
           svd_solver='auto', tol=0.0, whiten=False)
```

```
In [16]: print(pca.explained_variance_ratio_)

         [0.51286947 0.16362745 0.05832216 0.04435079 0.02167841 0.01874498
          0.01802328 0.0167939  0.01645183 0.01465396 0.01310866 0.01218797]
```

```
In [17]: pca.explained_variance_ratio_[0:12].sum()

Out[17]: 0.91081286534189
```

```
In [18]: reduced_data = pca.transform(feature_scale)
```

DIMENSIONALITY REDUCTION

- When using PCA analysis, one of the major goals is to reduce the data dimension to reduce task complexity.

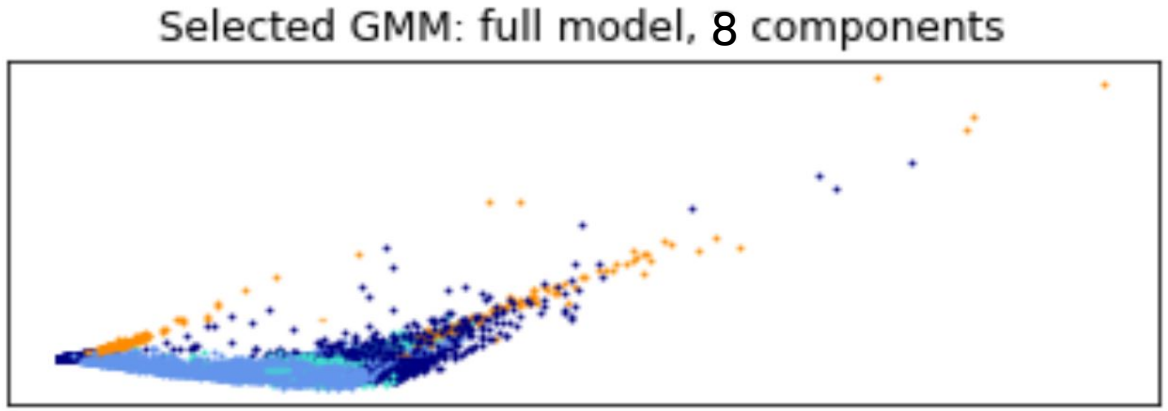
CLUSTERING

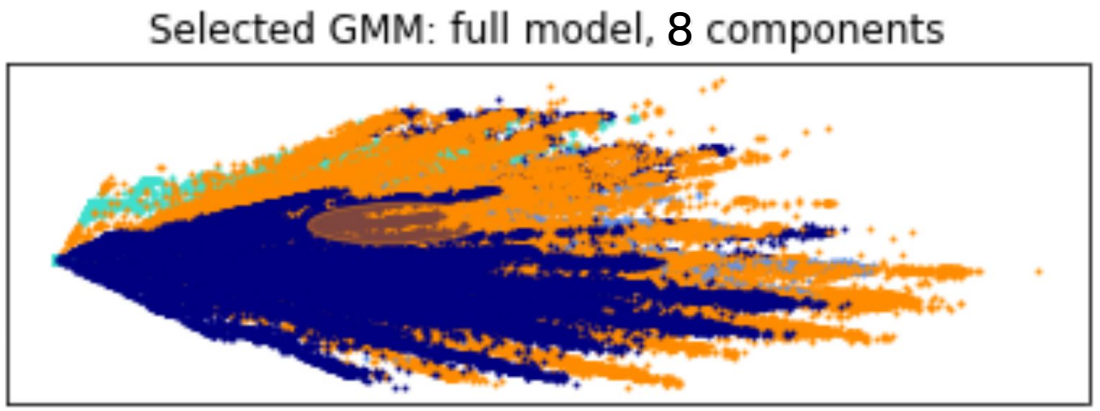
- We tried both Gauss mixture clustering algorithm and K-means clustering, by calculating the profile factor, we work out the best number of clusters , and finally decide on Gauss mixture clustering algorithm.

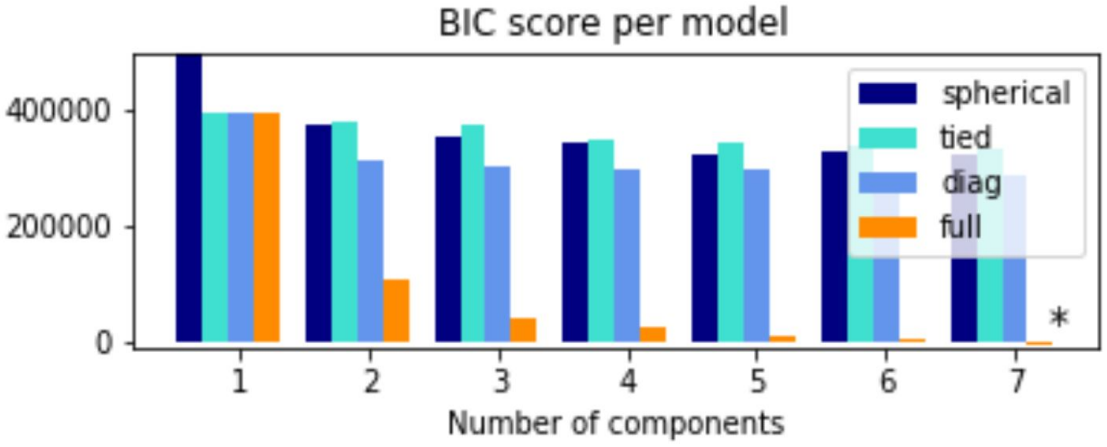
- **Pros and Cons**:
- **Kmeans**: Easy to apply, quick in calculation, assume all data distribute in the cluster equally.

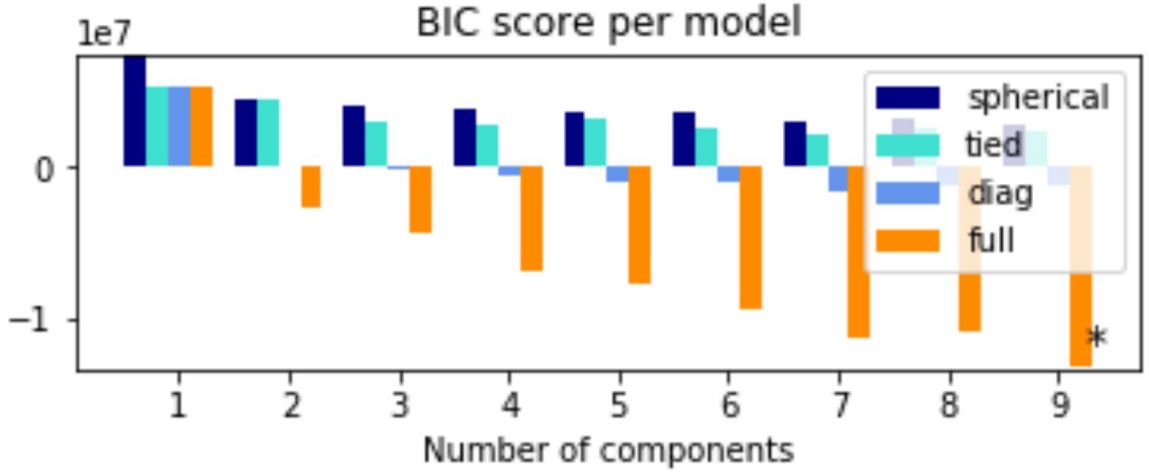
- **GMM**: Assume every feature weights different, assume data distribute in the cluster randomly, adequate to all continuous functions theoretically.

CLUSTERING VISUALIZED

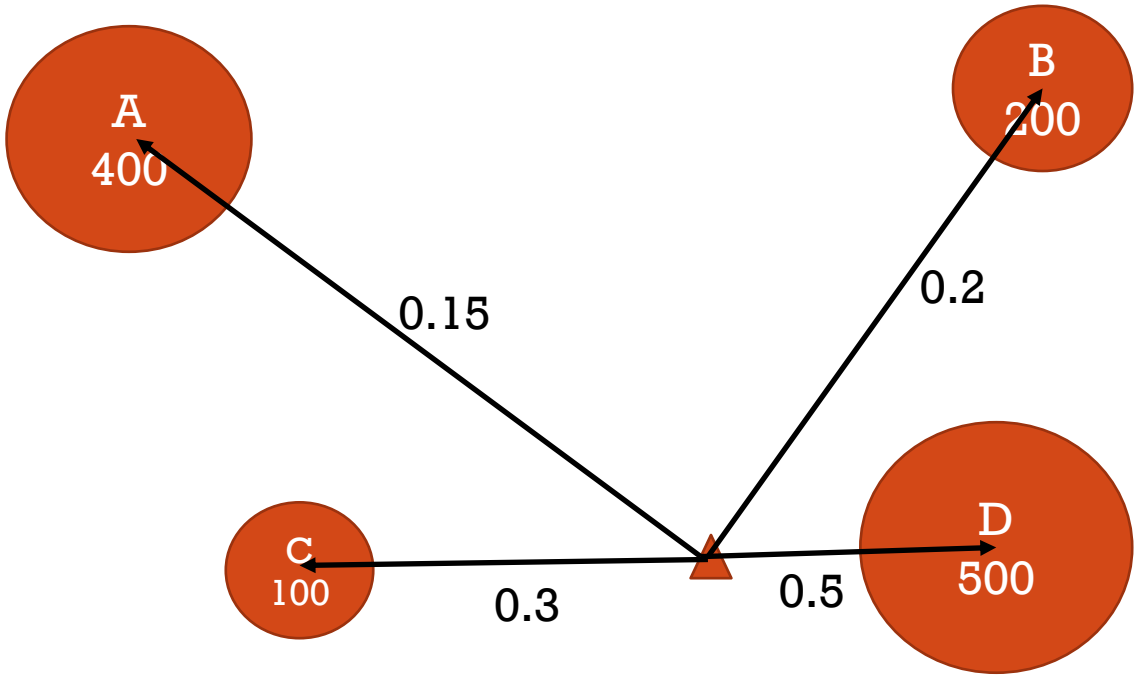








ALGORITHM



- Based on the distance, this customer 50% belongs to Cluster D, 30% belongs cluster C, 20% belongs to Cluster B, 15% belongs to Cluster A.
- The more people in the cluster, the less risky the cluster is.
- The credit score is calculated by the number of people in the clusters and the probability matrix. The higher the score is, the more innocent the instance will be.

Credit score = 0.15/400+0.2/200+0.3/100+0.5/500

ASSESSMENT FOR RESULTS

- This model applied unsupervised learning GMM to meet the requirements and divided providers into 6 clusters which are well distinct.

- All features in the model are created with careful consideration to make sure these features can describe pattern distribution of the data.

- The model generate two credit scores – medical provider and patients. The two scores can be applied as two features in claims evaluation. For the medical providers and patients who have a low credit scores, investigation team should pay more attention on claim evaluation and background checking.