**Medicare/Medicaid clustering**

Jonathan Shenkman

**Executive Summary**

After helping a patient with Medicare or Medicaid, the service provider submits a claim to the federal government for reimbursement. This claim is then recorded and an amount is determined and awarded to the service provider as reimbursement. These records are an invaluable tool in understanding the extremely high healthcare costs in America.

The goal of this analysis was to identify doctors that are spending more than they should in an effort to lower healthcare costs. In order to do this, we must first create some sort of metric or way to evaluate how much the doctor should be spending. To do this, I grouped together the reimbursement submissions based on their similarities. In theory, very similar submissions should receive very similar payments.

The grouping was done by first isolating the relevant qualities of a submission that would be most predictive of payment amount. After selecting these features, I grouped the submissions accordingly, and then compared the payment amounts of the grouped submissions. I found that many of the grouped submissions had very similar payment amounts, with a few exception, or outliers. These outliers, are potentially a key component to addressing the overspending trends in American healthcare. With the preliminary methods, I was able to identify over 2,000 submissions where doctors spent more than 99.7% of other examples that all had very similar payment amounts.

**Problem Statement**

The US government collects data on doctors and their submissions for Medicare reimbursement. I evaluated approximately 10 million of these records to group similar payment submissions and determine if any doctors received more payment than they should have.

**Assumptions**

* The subtleties of patient treatment are captured in these medical records. The records for reimbursement do not actually record any information about the patients, but simply a large number of data points on the doctor and the type of work they carried out. I assume that these observations hold sufficient information to inform the proper payment the doctor should receive
* I assume that the numeric features are normally distributed. This assumption enables the normalization of data for feature transformations

**Methodology**

The first steps to cluster the medical record is cleaning the data. The first step in data cleaning is identifying the obvious variables to remove. This was done by eliminating variables that are exclusively used to identify the medical provider and provide no bearing on quality of service or potential behavior. These variables include npi, name of provider, and address of provider (distinct from geographic location like city, zip code, or state). In addition to separating irrelevant data, I also separated the potential outputs at this stage. I took out the average payment amount, average standardized payment amount, and average maximum allowed charge – I will explain why I chose these variables in the analysis section.

The next step after identifying potentially useful variables is transforming the existing variables into a usable format. This is done in different ways for the numeric and categorical data. For the numeric data, I normalized all columns of numeric data by centering the distributions around 0 and with a standard deviation of 1. For the categorical data, I used OneHotEncoder to create binary dummy variables for each of the potential values. Some categorical variables had far too many categories, leaving the potential for thousands of dummy variables. I solved this problem in numerous ways. The first method was simple data cleaning. For the provider credentials, there was no standard way of formatting which led to thousands of different values. To reduce this, I parsed through the credentials, standardized the format, and only evaluated the first credential listed. This greatly reduced the dimension of this feature while maintaining a large amount of the originally information. For geographic features such as zip code and city, I felt that these features were sufficiently captured already by the variable for state. This meant I could exclude these features. Another method of category reduction was grouping. For HCPCS codes, there were thousands of potential variables, but they all fell into one of 11 categories that could be used to reduce the number of variables. The final method of category reduction was eliminating outliers. This was particularly useful in lowering the number of relevant provider types. Initially, there were thousands of included provider types, but many of them only included a small number of samples. To focus the analysis on variables that can draw conclusions with high statistical significance, I eliminated all categorical variables that contained under 50000 samples, approximately .5% of the sample size. I used a similar technique for the numeric variables, eliminating all samples that contained numeric results more than 2 standard deviations away from the mean value.

After preprocessing the data thoroughly, I created three versions of the dataset to test using clustering: a base case, the set of features most correlated with payment amount, and a feature set with dimension reduction using pca. I then used minibatchkmeans, an sklearn function in python that efficiently clusters large datasets through smaller samples. I clustered the three different datasets with a large range of cluster numbers, also recording the loss function and average silhouette score. Using the produced graphs and elbow method, I chose the optimal number of clusters and then reclustered the data using this cluster number. I then compared and analyzed the average payment amounts of the records that were clustered together, a feature that was not included in the initial clustering.

**Analysis**

The following graphs are of the loss functions for all numbers of clusters tested and then zoomed in on just the first 10 clusters.

 

There are no clear kinks in the first graph because it is quite zoomed out, however there are still important takeaways. The marginal benefit to the loss function significantly reduces after 20 clusters and the solutions tend to become more unstable at around 40 clusters. I believe the reason for the second observation is that as the number of clusters increase, we must also increase the sample size in the clustering algorithm otherwise the number of observations per cluster becomes too low.

After focusing on the first 10 clusters, a kink becomes more visible. Slight kinks appear at 6 for the base and pca cases and at about 5 for the correlation case. This would indicate that the ideal number of clusters is about 5 or 6. The following graph shows the average silhouette values for each of the cases as the number of clusters increases.



Counter to the initial graphs of the loss function, this graph indicates that solutions continue to improve in cluster quality past 10 clusters. The silhouette values become a bit unstable at higher cluster values as seen in the last graph, likely because the batch size was not properly adjusted for larger number of clusters. Instability aside, the graph shows that the silhouette average for the correlation features hovers around .7, a very high value. Pca would likely yield a similar value, but there was not sufficient time to test both datasets at such a high number of clusters. Another valuable piece of information from this graph is that the datasets must be cleaned before use – the quality of clusters in the base case is quite low and will likely not produce valuable clsuters.

There are two potential reasons that the pca and correlation datasets may produce higher silhouette averages without increasing actual cluster quality. The first concern is that by reducing the number of variables and artificially reducing the variance in the data, this would skew the dataset. Specifically, since there are a large number of categorical variables and few numeric, I worried that the clusters may simply group the categorical variables which would create a Euclidian distance of close to 0.

The other and primary concern with the higher performance of the pca and correlation feature sets is that they produce a higher silhouette value partially by clustering medical records submitted by the same doctor. The dataset samples are submissions for reimbursement by medical workers, meaning there are many submissions made by each doctor. Many of the features used categorize the type of work done by the doctor, which is likely very similar across the different submissions. If the number of clusters grows very large, this would allow clusters to form by individual doctors because these datapoints are artificially similar.

With this in mind, I explored the outputs of creating clusters with each of the cases with 5 clusters and then 30 clusters for the correlation dataset. After clustering the samples, I evaluated the similarity of the average payment amount to assess the quality of the clusters. I created a similar value to a partial f statistic by taking the overall payment variance and dividing by the payment variance within a cluster. The larger this value is, the better the cluster performed in grouping samples that should have similar payment amounts.

This partial f statistic however, does not account for the bias created by the potential clustering of samples by individual doctor. To address this, I created a baseline of probability that a sample is unique in a cluster by dividing the number of unique medical providers by the total number of samples. This baseline probability is about .09. Then for each cluster, I recalculated this statistic to determine if the cluster is biased by grouping together the samples by doctor. If the probability of being unique is above the baseline, I consider the sample mostly unbiased (by this cause). Below are the partial f vs. probability unique graphs for the 5 cluster trials of the base, correlated, and pca datasets in that order.

BASE:



CORRELATED:



PCA:



Though these are around the values where a kink is produced in the objective function graph, the results do not seem particularly substantive. For the base and pca cases, there is one cluster created with a reasonably high partial f statistic and probability of being unique, but these methods only produce one useful graph. Since the silhouette average graph showed a peak at around 30 clusters, I tested this value as well for both the correlation and pca datasets. The resulting graphs are below for the correlation dataset and then the pca dataset. There are two graphs for the correlation dataset because the first contains a data point with an extremely high partial f value that distorts the scale.

CORRELATION:





PCA:



The results from the 30 cluster plots above show there are a substantive number of clusters that did a good job clustering submissions in a way that predicts their expected payment amount. Further, many of these clusters do not succumb to potential flaws such as clustering by individual doctor.

**Conclusions**

By reducing the dimensions of the dataset through correlation and/or pca, it is possible to identify reimbursement submissions where the doctor spent more than they should have. I defined a very well formed cluster as one that reduced average payment variation by a factor of 20 and maintains a probability of unique doctor over .09.

Under this definition, **the correlation dataset was able to identify 15,505 charge submissions that are over 3 standard deviations away from the expected amount and the pca method identified 21,354 submissions**. These overspending samples may have been a function of unnecessary additional medical procedures or opting for more expensive methods over less expensive- both are examples in which behavior can be changed. Since these overspending examples may have been a function of extenuating circumstances, it is not fair to expect them to be completely eliminated. However, on just these well formed clusters, if these doctors edited their behavior to be just 2 standard deviations above the mean, vast amounts of money could be saved. **The correlation dataset finds that this would save over $700,000 and the pca dataset finds savings of over $1.5 million**. To put this into context, the overall average payment amount summed to about $600 million.

**Next steps**

* normalize onehotencoder. In the current model, onehotencoder outputs values of either 0 or 1, but this creates a uniform and artificial value when calculating the Euclidian distances
* try different objectives. I only experimented with evaluating the average payment amounts by doctors, but I have created a framework that would work well for exploring other objectives as well. For example, it may be beneficial to explore the different number of patients served by doctors to identify particularly unproductive doctors. On a similar note, exploring operations/actions per patient may detect doctors that conduct an unnecessary number of treatments for patients.
* try with larger batching size when the number of clusters increases. This prevents the unstable results seen in the loss function and average silhouette graphs.
* Test different variance levels for PCA dimension reduction. I kept the largest influencing components that could explain 50% of the data’s variance. This number was arbitrarily chosen to reduce the dimensions of the dataset by a large amount, so there may be other variance values that would produce better clusters
* Similar to the variance value, I arbitrarily chose .05 correlation as the cutoff for the correlation dataset. Other values may produce better clusters if given enough time to test them
* PCA is primarily used for numerical variables, so I think it may be beneficial to look into other component analysis better suited for categorical variables
* If given more time, I would like to evaluate the payment amount outliers further to discover potential reasons why they are spending so much money and their similarity to other records in the cluster. This is a critical step before any changes can be made.
* Inclusion of outside macrodata may improve cluster quality as well. For example, we did not include categorical features for city and zip code because there were too many categories, but there may be important numerical features we could use to proxy these categories. Information such as GDP per capita or employment in a region may have large impact on health spending outcomes.