CAPSTONE PROJECT 2 REPORT

On

PREDICTING WHAT PHYSICIANS SHOULD BE PAID BASED ON HISTORICAL DATA FROM THE CENTERS FOR MEDICARE AND MEDICAID SERVICES AND THE US FOOD AND DRUG ADMINISTRATION USING GRADIENT BOOSTED TREES AND RANDOM FOREST ALGORITHMS IN PYTHON

By

Andrew Ogah

A report submitted in partial fulfillment of the requirement of the

Springboard Data Science Career Track Fellowship

April 13, 2018

Introduction

In 2013, regulators introduced The Physician Financial Transparency Reports or Sunshine Act mandating that companies manufacturing drugs, devices, and biological agents report individual payments of greater than $10 or $100 in aggregate annually, provided to US physicians. This act was partly to curb any potential conflicts of interest that can inappropriately influence physician practice, because of the incentives derived from prescribing drugs or using devices from such companies. It also supported the need to control altered prescription behavior caused by increased interaction with pharmaceutical representatives.

Prior to this, in 2010, Congress also signed into law the Patient Protection and Affordable Care Act mandating residents in the US to either buy insurance or pay a fine if not covered by an employer-sponsored health plan, Medicaid, Medicare or other public insurance programs. The move provides patients visiting such practitioners the ability to file medical claims for treatments or drugs not within their deductibles to help offset cost and also help prevent unforeseen health cases.

Despite all these efforts made by regulators, chances remain that unethical practices abide. This is due to prior knowledge of the compensation from such companies, which can introduce bias towards some medications and skew prescription data. A good way of judging this is by closely comparing the frequency of prescription of drugs from manufactures amidst potential rewards. The question now is, can we predict the aggregated payments from a drug company to the physician in an effort to help regulators identify cases of conflict of interest? This is the question that would hopefully get answered in this analysis.

Materials and Methods

Data extraction and cleaning

The dataset used in this analysis are from the 2015 Medicare Provider Utilization and Payment Data: 2015 Part D Prescriber data of 24.5M rows and 21 columns, which includes 99.91% of total prescription claims, having prescribers with a valid NPI submitted by the Part D plan sponsors. And, the General Payment Data – Detailed Dataset 2015 Reporting Year of 11.5M records and 65 columns containing physician profile ID, demographic information and amount of payment received. Both of these dataset amongst others are maintained by the Center for Medicare and Medicaid Services on their website.

The variables for the datasets are as shown in Appendix I and details explaining each variables in depth can be found in the Medicare Fee-For Service Provider Utilization & Payment Data Part D Prescriber Public Use File: A Methodological Overview and the Open Payments Public Use Files: Methodology Overview & Data Dictionary on my Github or downloaded from the www.cms.gov website.

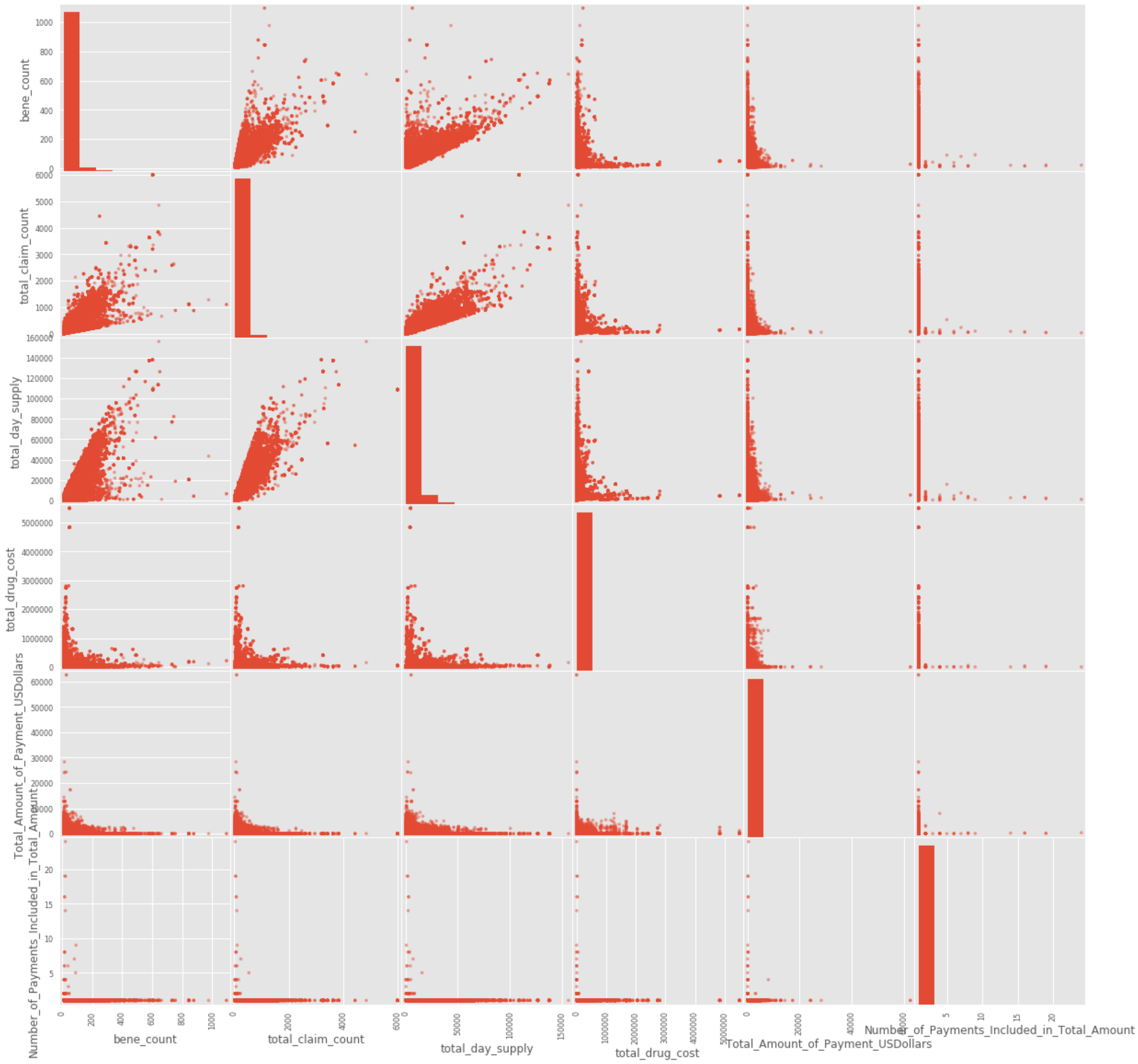
Analyzing the features of both datasets, I noticed that the IDs are not right keys for merging both in into one, because not all doctors registered as a national provider in the Medicare program received payments for a particular prescription from a drug company. So, I engineered a new feature by combining the first, and last names of the physician. This new feature, in addition to drug name, city and state where a medical practitioner is located now acted as this mapping key to merge both datasets. I also created a new feature for the state based on five situated regions, because the original variable had high cardinality issue with uneven classes. At the time of writing this report, the drug names and medical specialties had similar problems. I contemplate using a fuzzy matching algorithm with variable edit distances to re-engineer the drug name feature in future improvement of the model.

Considering that the dataset is massive, I decided to select a random sample of 20% from each dataset before the merge. Further cleaning the dataset, I removed noisy columns like the first and last name, 'NPI' - national provider identifier, and 'physician profile ID,' because they are assigned values and do not represent a random measure on each observation. They introduce high dimensionality and cause unnecessary overfitting of the data. I also dropped the drug name for now because of the issue mentioned earlier, but should be re-added after feature engineering. The 'name of associated covered drug or biological1' feature was redundant since it is a variant of the drug name feature, so dropped that too. Because the record was still too extensive for my machines resources, I decided to remove all unknown values. This act is not a good practice because you may have some information that could be of value in some discarded data. I will revisit those when fine-tuning the model.

At this point, it is advisable to create an out of sample test set data. Doing this now usually prevents what is called an information snooping bias. The premise is that the human brain is known to be a fantastic pattern detection system, which means that it is highly prone to overfitting. So, we set the test set data aside that we will not touch at all, and continue with the training set data. The test data set is used as an out of sample dataset to validate the model.

Exploratory Data Analysis

Exploring the dataset a little bit and looking closely at the numerical features using some plots, we see little or no trends of the explanatory variables with the target. Most of the variables are either slightly left or right skewed, but nothing severe since the spread of the data points is not overly high. Skewness translates to the presence of some outliers in the distribution, which sometimes may not bode well for linear models, and may need normalization to provide better results. My decision was to leave those in these for now, to see how the models perform with the option of revisiting if models don't meet expected outcomes.



Correlation matrix showing relationship between each predictors and the target – Total Amount of Payments

Statistical Inference

With the knowledge that most predictors have a very weak association with our target variable, the question now is: are these predictors significant enough in explaining variability in the Total Amount of Payment - the target variable? To answer this question, we usually fit a multivariate linear regression model to the dataset. And before doing that, we should ensure that the data, most especially our predictors meet the assumptions for the hypothesis of a linear model. Which are:

- sample must be reasonably random,
- data must be from a normal distribution or large sample (need to check n ≥ 30),

- the observations are independent.
- the true relationship is linear. Check that the scatter plot is roughly linear and that the residual plot has no pattern.
- the standard deviation of the response y about the true line is the same everywhere. Look at the residual plot and check that the residuals have roughly the same spread across all the x-values.
- for any fixed value of x, the response y varies normally about the true line. Check a histogram or stemplot of the residuals.

Since I believe it is better spending more time on prediction and analyzing the data than tweaking the abnormalities in the data, I have chosen to go the way of ensemble models that are robust against linearity, skewness, and collinearity. They also provide features that can identify which models are essential in explaining or predicting the target variables.

Building the machine learning algorithm.

My approach to this was comparing different algorithms to see how they performed using the mean squared error (MSE) as a yard stick. The MSE of an estimator measures the average of the squares of the erros or deviations- that is, the difference between the estimator and what is being estimated. Since our dataset is significantly imbalanced, care was taken to balance the training dataset before training the model because most models especially the linear ones do not perform well such. The Random Forest model was selected because it performed better than the other two models (Linear Regression and Gradient Boosted Trees) used. Carrying out the prediction on the imbalanced out of sample data we created at the start of the analysis, it shows that our model generalizes well to a dataset has not been seen before.

The model was also cross-validated to ensure we are not overfitting the training dataset using the KFold validation techniques that splits the training set into 10 distinct subsets called folds. The selected algorithm was then trained and evaluated 10 times on the model, picking a different fold for evaluation every time and training on the other 9 folds. The Scikit-Learn cross-validation features was used, which expects a utility function (greater is better) rather than a cost function (lower is better) of the MSE.

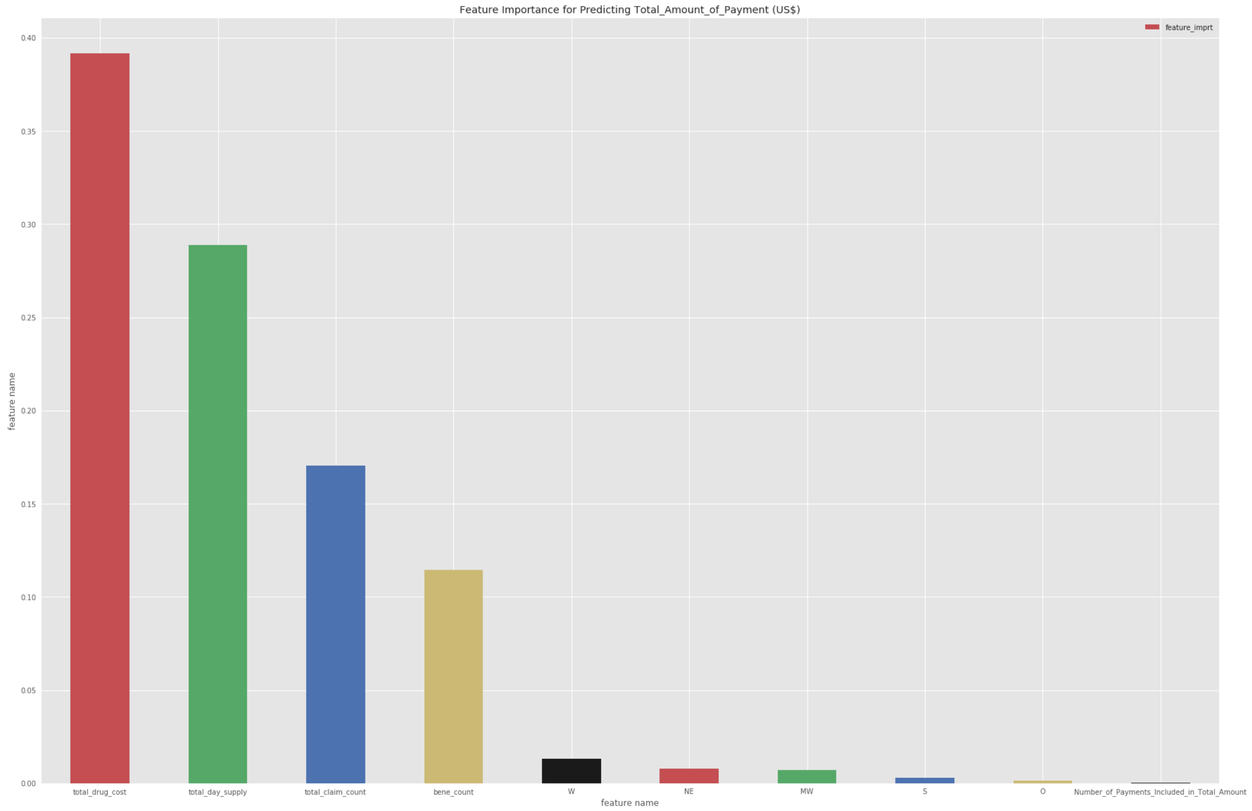
In addition to checking for accuracy by comparing the MSE amongst three algorithms, I also tried to fine-tune the selected model by fiddling with the hyperparameters to find the great combination of hyperparameter values. The Scikit- Learn’s GridSearchCV feature was used to automate this process. All that was necessary was to specify which hyperparameters are needed to experiment with, and what values to try out, and it evaluated all possible combinations of hyperparameter values using cross-validation.

Results and Conclusion

The first result I got from comparison of the MSE amongst all three models was questionable because all models gave same result of approximately $292.37. However, after KFold cross-validation with ten-fold, Random Forest gave the lowest score mean score of $272.29 with a tighter standard deviation of $31.12. Remember, with MSE or RMSE, lower is better because we are trying to reduce the difference between the estimate or predicted value and the real value.

With the best model selected, GridSearchCV was used for fine tuning to search for the most important hyperparameters. And, that gave the result of maximum best feature of eight (8) predictors and ten (10) estimators. Using the result of these estimators, the important features and their ranks were computed and plotted as shown below:

Important features	Ranking
total_drug_cost	0.39149840381059342
total_day_supply	0.28871781108074251
total_claim_count	0.17050574957727693
bene_count	0.11463922485071887
W	0.013153519455514646
NE	0.007739768175049247
MW	0.007241369450545816
S	0.0029457455014294904
O	0.0015673829352763548
Number_of_Payments_Included_in_Total_Amount	0.00015465677240344316



Plot showing the importance features needed to predict the Total_Amount_of_Payment_USDollars

A refined model was now created with these important features and hyperparamters and the result evaluated on the out of sample test set created at the start of the analysis. The final result was $288. 54. This is still better than what we had in the Gradient Boosted Trees and the Linear Regression. This performance is slightly worse than what we measured using cross-validation because the system fine-tuned lots of hyperparameters to perform well on the validation data, but may likely not perform as well on unknown dataset like we just demonstrated. So, one should resist the temptation of unnecessarily tuning hyperparameters as that may not translate to new unseen data. What we just need to see is having predicted result within range of what we had for the cross-validation.

These estimates are good considering we still have the options of adding more features like the re-worked drug name to improve it. Other things that need looked at are:

- concentrating more on research on domain knowledge to help improve some features like the ‘drug names’ and ‘specialty types’
- looking closely at outliers and multicollinearity issues. Questions have to be asked if removing some data points make sense or not.

APPENDIX 1

Table 1: Medicare Provider Utilization and Payment Data: 2015 Part D Prescriber

Column Name	Description	Data Type
npi	National Provider Identifier	Text
nppes_provider_last_org_name	Last Name/Organization Name of the Provider	Text
nppes_provider_first_name	First Name of the Provider	Text
nppes_provider_city	City of the Provider	Text
nppes_provider_state	State Code of the Provider	Text
specialty_description	Provider Specialty Type	Text
description_flag	Source of Provider Specialty	Text
drug_name	Brand Name	Text
generic_name	USAN Generic Name - Short Version	Text
bene_count	Number of Medicare Beneficiaries	Number
total_claim_count	Number of Medicare Part D Claims, Including Refills	Number
total_30_day_fill_count	Number of Standardized 30-Day Fills, Including Refills	Number
total_day_supply	Number of Day's Supply for All Claims	Number
total_drug_cost	Aggregate Cost Paid for All Claims	Money
bene_count_ge65	Number of Medicare Beneficiaries Age 65+	Number
bene_count_ge65_suppress_flag	Reason for Suppression of Bene_Count_Ge65	Text
total_claim_count_ge65	Number of Claims, Including Refills, for Beneficiaries Age 65+	Number
ge65_suppress_flag	Reason for Suppression of Total_Claim_Count_Ge65, Total_30_Day_Fill_Count_Ge65, Total_Day_Supply_Ge65, and Total_Drug_Cost_Ge65	Text
total_30_day_fill_count_ge65	Number of Standardized 30-Day Fills, Including Refills, for Beneficiaries Age 65+	Number
total_day_supply_ge65	Number of Day's Supply for All Claims for Beneficiaries Age 65+	Number
total_drug_cost_ge65	Aggregate Cost Paid for All Claims for Beneficiaries Age 65+	Money

Table 2. General Payment Data – Detailed Dataset 2015 Reporting Year.

Column Name	Data Type
Change_Type	Text
Covered_Recipient_Type	Text
Teaching_Hospital_CCN	Text
Teaching_Hospital_ID	Text
Teaching_Hospital_Name	Text
Physician_Profile_ID	Text
Physician_First_Name	Text
Physician_Middle_Name	Text
Physician_Last_Name	Text
Physician_Name_Suffix	Text

Recipient_Primary_Business_Street_Address_Line1	**Text**
Recipient_Primary_Business_Street_Address_Line2	**Text**
Recipient_City	**Text**
Recipient_State	**Text**
Recipient_Zip_Code	**Text**
Recipient_Country	**Text**
Recipient_Province	**Text**
Recipient_Postal_Code	**Text**
Physician_Primary_Type	**Text**
Physician_Specialty	**Text**
Physician_License_State_code1	**Text**
Physician_License_State_code2	**Text**
Physician_License_State_code3	**Text**
Physician_License_State_code4	**Text**
Physician_License_State_code5	**Text**
Submitting_Applicable_Manufacturer_or_Applicable_GPO_Name	**Text**
Applicable_Manufacturer_or_Applicable_GPO_Making_Payment_ID	**Text**
Applicable_Manufacturer_or_Applicable_GPO_Making_Payment_Name	**Text**
Applicable_Manufacturer_or_Applicable_GPO_Making_Payment_State	**Text**
Applicable_Manufacturer_or_Applicable_GPO_Making_Payment_Country	**Text**
Total_Amount_of_Payment_USDollars	**Number**
Date_of_Payment	**Floating Timestamp**
Number_of_Payments_Included_in_Total_Amount	**Number**
Form_of_Payment_or_Transfer_of_Value	**Text**
Nature_of_Payment_or_Transfer_of_Value	**Text**
City_of_Travel	**Text**
State_of_Travel	**Text**
Country_of_Travel	**Text**
Physician_Ownership_Indicator	**Text**
Third_Party_Payment_Recipient_Indicator	**Text**
Name_of_Third_Party_Entity_Receiving_Payment_or_Transfer_of_Value	**Text**
Charity_Indicator	**Text**
Third_Party_Equals_Covered_Recipient_Indicator	**Text**
Contextual_Information	**Text**
Delay_in_Publication_Indicator	**Text**
Record_ID	**Text**
Dispute_Status_for_Publication	**Text**
Product_Indicator	**Text**
Name_of_Associated_Covered_Drug_or_Biological1	**Text**
Name_of_Associated_Covered_Drug_or_Biological2	**Text**
Name_of_Associated_Covered_Drug_or_Biological3	**Text**
Name_of_Associated_Covered_Drug_or_Biological4	**Text**
Name_of_Associated_Covered_Drug_or_Biological5	**Text**
NDC_of_Associated_Covered_Drug_or_Biological1	**Text**
NDC_of_Associated_Covered_Drug_or_Biological2	**Text**
NDC_of_Associated_Covered_Drug_or_Biological3	**Text**
NDC_of_Associated_Covered_Drug_or_Biological4	**Text**
NDC_of_Associated_Covered_Drug_or_Biological5	**Text**
Name_of_Associated_Covered_Device_or_Medical_Supply1	**Text**
Name_of_Associated_Covered_Device_or_Medical_Supply2	**Text**

Name_of_Associated_Covered_Device_or_Medical_Supply3	**Text**
Name_of_Associated_Covered_Device_or_Medical_Supply4	**Text**
Name_of_Associated_Covered_Device_or_Medical_Supply5	**Text**
Program_Year	**Number**
Payment_Publication_Date	**Floating Timestamp**