Fraud Detection in Healthcare Insurance Claims

Ashpak Rakeeb *B00913796*
Shravya Reddy *B00911193*
Sri Ramya *B00900307*
Venkata Vijaya *B00912916*
Meagan Sinclair *B00737317*

***Abstract*—In terms of the value of our healthcare system and in terms of money, fraudulent claims come with a very high price. Healthcare spending in the United States totals at 3.6 trillion dollars in 2018, which equates to billions in insurance claims. [1]. There is no denying that some of these claims are fraudulent. [1]. In a huge set of Medicare data (more than 550,000 claims), we developed a process to predict potential fraud in medical insurance claims where the doctor files the paperwork and submits the claim on the patient's behalf. We experimented with four distinct approaches: Decision Tree Classifier, XGBoost Classifier, Bagging Classifier, and Stacked Classifier. According to our experiments, the XGBoost classifier had the best generalization and a 78% accuracy rate on the reserved test set. The other models' performances were likewise satisfactory, with the lowest-scoring model, the Stacked Classifier, having an accuracy of 72%.**

I. Introduction

Every year, healthcare fraud costs the industry billions of dollars as fraudsters find the healthcare industry an attractive target [1]. Common healthcare frauds include beneficiaries claiming medical conditions for which they were not treated and submitting multiple claims for the same service. Claiming more than the actual monetary cost spent on treatment is also a common fraud technique [2].

If insurance companies don't have procedures in place to identify and stop fraudulent practices, they are at risk to lose significant profit. Further, if fraud detection techniques are unreliable, healthcare providers could be incorrectly subjected to an inquiry that could harm their reputation and revenue, as well as damaging the reputation and wasting resources of the insurance provider. This can raise health insurance premiums for customers. Insurance companies prefer to make payments without conducting investigations since doing so is expensive and time-consuming. Additionally, incorrectly detecting fraudulent claims could delay reimbursement, which is undesirable for honest customers and degrades the public opinion of the insurance provider [3].

To solve this business problem, insurance providers now use predictive analytical methods to prevent false medical bills before healthcare providers receive payments. The goal of this project is to create a model which can predict whether claim is potentially fraudulent or not based on the claim information. This would allow the insurance providers to assess claim details and determine if further fraud investigation is required. This is a supervised classification problem as we have a specific binary target, potential fraud.

II. Literature Review

Several researchers have tackled the problem of fraud detection in healthcare insurance claims. Multiple clustering methods have been employed to identify suspicious claims within databases and provide insight into groups within the data [4]. This is a useful basis for further work toward classification but does not provide conclusive fraud detection. Bayesian co-clustering methods have also been employed [5]. This method is beneficial for the problem as co-clustering allows for modeling relationships between entities and potentially better results. There is little work done on ensemble models or boosted learning algorithms [6].

The accompanying notebook on Kaggle.com utilized feature selection methods and showed testing of various models [7]. Model types trialed included logistic regression, Random Forest classifier, and auto-encoders with good results, around 90% accuracy on test data. The author noted that increasing the database size, vectoring medical codes, and utilizing further ensemble models could lead to better results. The author of the notebook also noted the hyperparameter tuning was not performed on the models, so careful tuning will also be a focus of the project. While this previous work attempted to predict fraud for each provider from the information of all claims belonging to the provider, this project will attempt to predict fraud based on a single claim. This is more useful for the business problem as fraud could be identified immediately after 1 claim is submitted, as opposed to requiring multiple claims information, which is less timely.

This project is valuable as various new model types will be applied to the problem and evaluated. Ensembled and boosted models will also be tested to determine if performance can be further improved. This work extends current knowledge of fraudulent claim detection methods.

III. Methodology

A. Data Exploration

The data set chosen for the project contains relevant information on fraudulent and non-fraudulent healthcare claims [8]. This data is open source and is hosted on Kaggle.com. The raw data is broken down into 4 sections: inpatient data, outpatient data, beneficiary details data, list of possibly fraudulent providers.

Inpatient data contains information about claims made by beneficiaries who were admitted into the hospital. The data describes the corresponding beneficiary (the person who submitted the claim), claim date range, relevant monetary amounts, provider and physician codes, diagnoses information and procedure information. These values are summarized below in Figure 1a. Outpatient data describes claims made by patients who were not admitted. The data contains the same fields as inpatient data, excluding the admission and discharge dates.

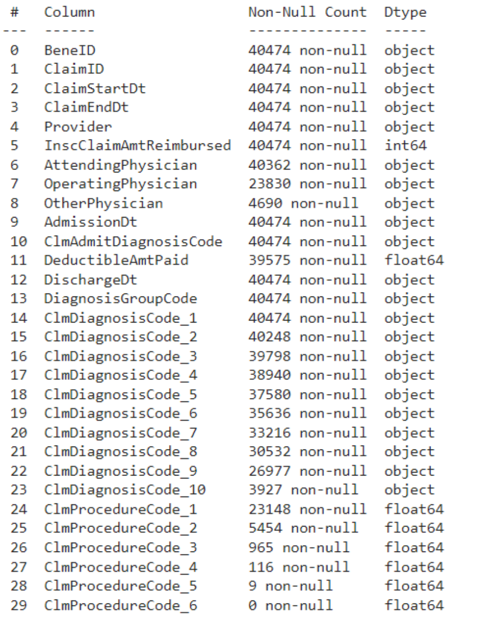
Beneficiary data includes information describing the person who submitted the claim. This includes date of birth and death, gender, race, residential location, the status of various chronic conditions, length of time covered by insurance, and relevant annual monetary values such as deductibles and reimbursements. These values are summarized below in Figure 1b.

Finally, the last data set contains a list of provider codes marked as either potentially fraudulent or not. The domain concepts graph of the entire feature set is shown below in Figure 2. The corresponding Analytic Base Table with all raw features and their descriptions is shown in Figure 3.

Since the data is split in separate subsets, merging the data into one comprehensive set was the first step. The inpatient and outpatient data sets were merged and the corresponding potential fraud indicator based on the provider value was added to each claim. Next, separate categorical and continuous feature reports were created for the combined data set, these are shown in Appendix A and B.

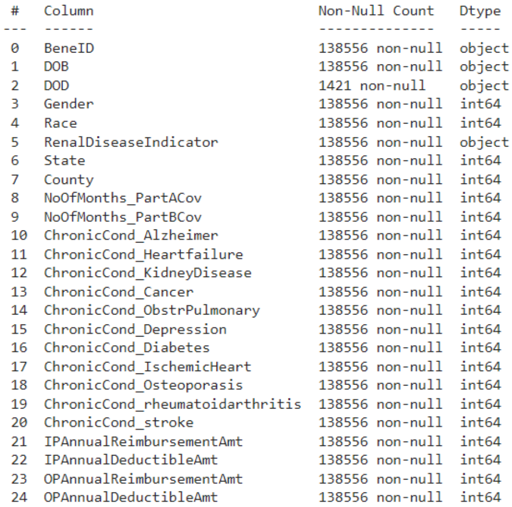
B. Data Pre-Processing

Several steps were required to prepare the data for model training. First, the binary features such as disease indicators, chronic conditions, and potential fraud indicator were transformed into binary values 0 and 1. Since the claim codes contained alphanumeric strings to indicate the diagnostic claim, these were re-encoded into numericals. The procedure codes were strictly numeric so these were left.



```
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   BeneID                  40474 non-null  object
 1   ClaimID                 40474 non-null  object
 2   ClaimStartDt            40474 non-null  object
 3   ClaimEndDt              40474 non-null  object
 4   Provider                40474 non-null  object
 5   InscClaimAmtReimbursed  40474 non-null  int64
 6   AttendingPhysician      40362 non-null  object
 7   OperatingPhysician      23830 non-null  object
 8   OtherPhysician          4690 non-null   object
 9   AdmissionDt             40474 non-null  object
 10  ClmAdmitDiagnosisCode   40474 non-null  object
 11  DeductibleAmtPaid       39575 non-null  float64
 12  DischargeDt             40474 non-null  object
 13  DiagnosisGroupCode      40474 non-null  object
 14  ClmDiagnosisCode_1      40474 non-null  object
 15  ClmDiagnosisCode_2      40248 non-null  object
 16  ClmDiagnosisCode_3      39798 non-null  object
 17  ClmDiagnosisCode_4      38940 non-null  object
 18  ClmDiagnosisCode_5      37580 non-null  object
 19  ClmDiagnosisCode_6      35636 non-null  object
 20  ClmDiagnosisCode_7      33216 non-null  object
 21  ClmDiagnosisCode_8      30532 non-null  object
 22  ClmDiagnosisCode_9      26977 non-null  object
 23  ClmDiagnosisCode_10     3927 non-null   object
 24  ClmProcedureCode_1      23148 non-null  float64
 25  ClmProcedureCode_2      5454 non-null   float64
 26  ClmProcedureCode_3      965 non-null    float64
 27  ClmProcedureCode_4      116 non-null    float64
 28  ClmProcedureCode_5      9 non-null      float64
 29  ClmProcedureCode_6      0 non-null      float64
```

(a) Inpatient Data Summary



```
 #   Column                           Non-Null Count   Dtype
---  ------                           --------------   -----
 0   BeneID                           138556 non-null  object
 1   DOB                              138556 non-null  object
 2   DOD                              1421 non-null    object
 3   Gender                           138556 non-null  int64
 4   Race                             138556 non-null  int64
 5   RenalDiseaseIndicator            138556 non-null  object
 6   State                            138556 non-null  int64
 7   County                           138556 non-null  int64
 8   NoOfMonths_PartACov              138556 non-null  int64
 9   NoOfMonths_PartBCov              138556 non-null  int64
 10  ChronicCond_Alzheimer            138556 non-null  int64
 11  ChronicCond_Heartfailure         138556 non-null  int64
 12  ChronicCond_KidneyDisease        138556 non-null  int64
 13  ChronicCond_Cancer               138556 non-null  int64
 14  ChronicCond_ObstrPulmonary       138556 non-null  int64
 15  ChronicCond_Depression           138556 non-null  int64
 16  ChronicCond_Diabetes             138556 non-null  int64
 17  ChronicCond_IschemicHeart        138556 non-null  int64
 18  ChronicCond_Osteoporasis         138556 non-null  int64
 19  ChronicCond_rheumatoidarthritis  138556 non-null  int64
 20  ChronicCond_stroke               138556 non-null  int64
 21  IPAnnualReimbursementAmt         138556 non-null  int64
 22  IPAnnualDeductibleAmt            138556 non-null  int64
 23  OPAnnualReimbursementAmt         138556 non-null  int64
 24  OPAnnualDeductibleAmt            138556 non-null  int64
```

(b) Beneficiary Data Summary

Fig. 1: Raw Data Summaries

Next, derived features were selected and created. The date of birth and date of death were transformed into an age and a binary feature indicating if the beneficiary was living or not. Time in hospital was derived from the admission and discharge dates and length of claim was derived from the claim start date and claim end date. The features which were used for derivation were dropped.

The features were then assessed for quality. The percent of values missing, the cardinality, and the presence of outliers were determined for each feature. From large portions of missing values as observed in the feature reports, the claim diagnoses codes 8, 9, and 10 were dropped along with all claim procedure codes excluding the first one. While claim procedure code 1 also had a high percentage of missing values (96%) it was decided to retain this feature's information for the time being. The remaining diagnosis codes also had high missing values percentages and we chose the cut-off for retention conservatively at less than 90% missing values. Again, this was to retain more infor-

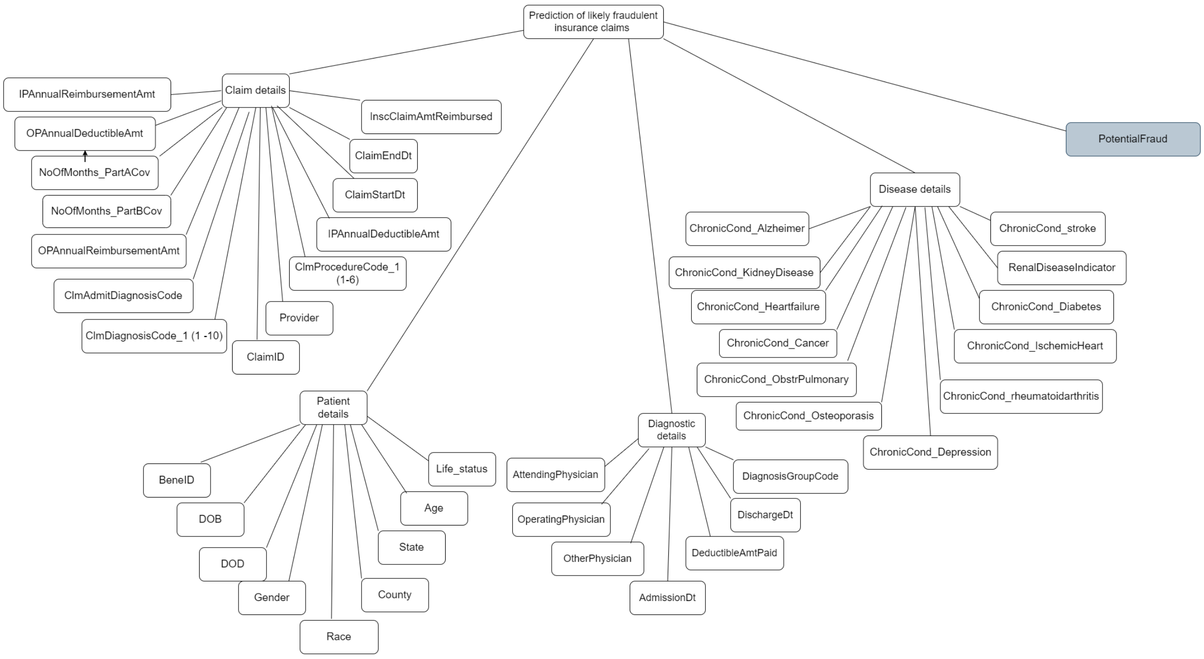
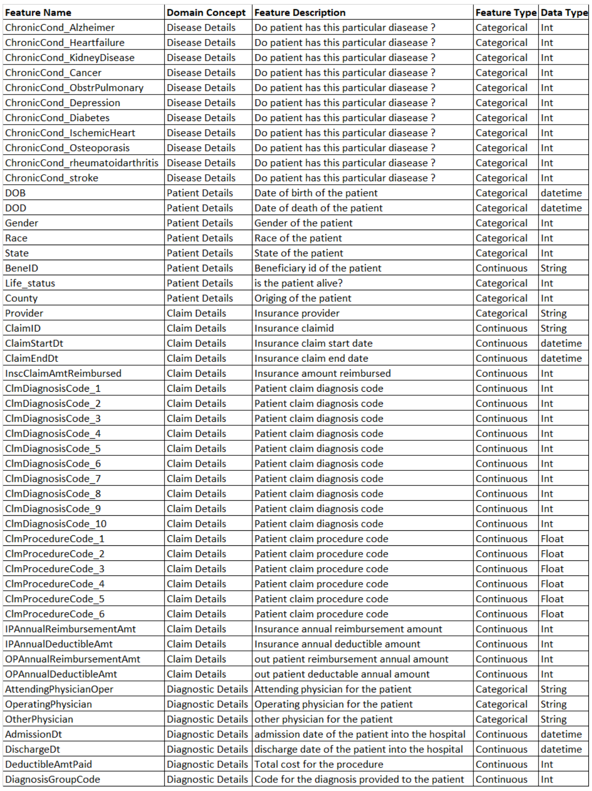


Fig. 2: Domain Concepts



Feature Name	Domain Concept	Feature Description	Feature Type	Data Type
ChronicCond_Alzheimer	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_Heartfailure	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_KidneyDisease	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_Cancer	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_ObstrPulmonary	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_Depression	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_Diabetes	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_IschemicHeart	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_Osteoporasis	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_rheumatoidarthritis	Disease Details	Do patient has this particular diasease ?	Categorical	Int
ChronicCond_stroke	Disease Details	Do patient has this particular diasease ?	Categorical	Int
DOB	Patient Details	Date of birth of the patient	Categorical	datetime
DOD	Patient Details	Date of death of the patient	Categorical	datetime
Gender	Patient Details	Gender of the patient	Categorical	Int
Race	Patient Details	Race of the patient	Categorical	Int
State	Patient Details	State of the patient	Categorical	Int
BeneID	Patient Details	Beneficiary id of the patient	Continuous	String
Life_status	Patient Details	is the patient alive?	Categorical	Int
County	Patient Details	Origing of the patient	Categorical	Int
Provider	Claim Details	Insurance provider	Categorical	String
ClaimID	Claim Details	Insurance claimid	Continuous	String
ClaimStartDt	Claim Details	Insurance claim start date	Continuous	datetime
ClaimEndDt	Claim Details	Insurance claim end date	Continuous	datetime
InscClaimAmtReimbursed	Claim Details	Insurance amount reimbursed	Continuous	Int
ClmDiagnosisCode_1	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_2	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_3	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_4	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_5	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_6	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_7	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_8	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_9	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmDiagnosisCode_10	Claim Details	Patient claim diagnosis code	Continuous	Int
ClmProcedureCode_1	Claim Details	Patient claim procedure code	Continuous	Float
ClmProcedureCode_2	Claim Details	Patient claim procedure code	Continuous	Float
ClmProcedureCode_3	Claim Details	Patient claim procedure code	Continuous	Float
ClmProcedureCode_4	Claim Details	Patient claim procedure code	Continuous	Float
ClmProcedureCode_5	Claim Details	Patient claim procedure code	Continuous	Float
ClmProcedureCode_6	Claim Details	Patient claim procedure code	Continuous	Float
IPAnnualReimbursementAmt	Claim Details	Insurance annual reimbursement amount	Continuous	Int
IPAnnualDeductibleAmt	Claim Details	Insurance annual deductible amount	Continuous	Int
OPAnnualReimbursementAmt	Claim Details	out patient reimbursement annual amount	Continuous	Int
OPAnnualDeductibleAmt	Claim Details	out patient deductable annual amount	Continuous	Int
AttendingPhysicianOper	Diagnostic Details	Attending physician for the patient	Categorical	String
OperatingPhysician	Diagnostic Details	Operating physician for the patient	Categorical	String
OtherPhysician	Diagnostic Details	other physician for the patient	Categorical	String
AdmissionDt	Diagnostic Details	admission date of the patient into the hospital	Continuous	datetime
DischargeDt	Diagnostic Details	discharge date of the patient into the hospital	Continuous	datetime
DeductibleAmtPaid	Diagnostic Details	Total cost for the procedure	Continuous	Int
DiagnosisGroupCode	Diagnostic Details	Code for the diagnosis provided to the patient	Continuous	Int

Fig. 3: Analytical Base Table

mation at this stage. The claim admission diagnosis code, diagnosis group code, operating physician, and other physicians were also dropped for missing values. Attending physician was also dropped as the cardinality was too high to be useful for a categorical feature (~82,000). No issues with outliers were detected. The claim and beneficiary IDs were dropped next as they contain identification codes and not useful data.

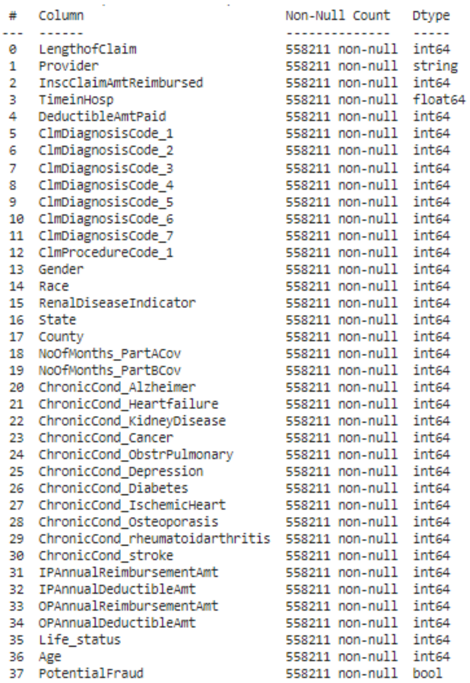
Finally, the provider code was also dropped. Since the original data set tagged potential fraud based on provider code, but the desired model would predict fraud based on one specific claim, this would potentially cause the model to simply learn which providers are fraudulent over the claims. This would not allow the model to predict on new data. The final feature set is summarized below in Figure 4. The final number of instances is 558211.

C. Feature Selection

The sklearn library tools SelectKBest and mutual_info_classif was utilized to determine the most useful 15 features to use for model training [9], [10]. mutual_info_classif estimates the mutual information to score each feature for dependency concerning the target feature. It functions based on entropy calculations using k-nearest neighbors' distance. SelectKBest selects the top K features based on the scores from mutual_info_classif. The top 15 features and the corresponding mutual information score are shown below in Figure 5.

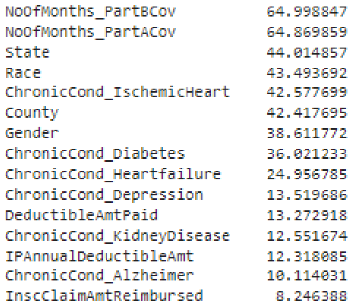
D. Model Building, Parameter Tuning, and Training

This section describes the models chosen for testing, the justification of choice, and the hyperparameter tuning performed on each before the final training. Section 4 will describe the chosen evaluation plan and the results of training and evaluation will be shown in Section 5. Hyperparameter tuning was performed with 5 cross-fold shuffle split validation with 20% test size



```
 #   Column                              Non-Null Count   Dtype
---  ------                              --------------   -----
 0   LengthofClaim                       558211 non-null  int64
 1   Provider                            558211 non-null  string
 2   InscClaimAmtReimbursed              558211 non-null  int64
 3   TimeinHosp                          558211 non-null  float64
 4   DeductibleAmtPaid                   558211 non-null  int64
 5   ClmDiagnosisCode_1                  558211 non-null  int64
 6   ClmDiagnosisCode_2                  558211 non-null  int64
 7   ClmDiagnosisCode_3                  558211 non-null  int64
 8   ClmDiagnosisCode_4                  558211 non-null  int64
 9   ClmDiagnosisCode_5                  558211 non-null  int64
 10  ClmDiagnosisCode_6                  558211 non-null  int64
 11  ClmDiagnosisCode_7                  558211 non-null  int64
 12  ClmProcedureCode_1                  558211 non-null  int64
 13  Gender                              558211 non-null  int64
 14  Race                                558211 non-null  int64
 15  RenalDiseaseIndicator               558211 non-null  int64
 16  State                               558211 non-null  int64
 17  County                              558211 non-null  int64
 18  NoOfMonths_PartACov                 558211 non-null  int64
 19  NoOfMonths_PartBCov                 558211 non-null  int64
 20  ChronicCond_Alzheimer               558211 non-null  int64
 21  ChronicCond_Heartfailure            558211 non-null  int64
 22  ChronicCond_KidneyDisease           558211 non-null  int64
 23  ChronicCond_Cancer                  558211 non-null  int64
 24  ChronicCond_ObstrPulmonary          558211 non-null  int64
 25  ChronicCond_Depression              558211 non-null  int64
 26  ChronicCond_Diabetes                558211 non-null  int64
 27  ChronicCond_IschemicHeart           558211 non-null  int64
 28  ChronicCond_Osteoporasis            558211 non-null  int64
 29  ChronicCond_rheumatoidarthritis     558211 non-null  int64
 30  ChronicCond_stroke                  558211 non-null  int64
 31  IPAnnualReimbursementAmt            558211 non-null  int64
 32  IPAnnualDeductibleAmt               558211 non-null  int64
 33  OPAnnualReimbursementAmt            558211 non-null  int64
 34  OPAnnualDeductibleAmt               558211 non-null  int64
 35  Life_status                         558211 non-null  int64
 36  Age                                 558211 non-null  int64
 37  PotentialFraud                      558211 non-null  bool
```

Fig. 4: Final Data Set



```
NoOfMonths_PartBCov          64.998847
NoOfMonths_PartACov          64.869859
State                        44.014857
Race                         43.493692
ChronicCond_IschemicHeart    42.577699
County                       42.417695
Gender                       38.611772
ChronicCond_Diabetes         36.021233
ChronicCond_Heartfailure     24.956785
ChronicCond_Depression       13.519686
DeductibleAmtPaid            13.272918
ChronicCond_KidneyDisease    12.551674
IPAnnualDeductibleAmt        12.318085
ChronicCond_Alzheimer        10.114031
InscClaimAmtReimbursed        8.246388
```

Fig. 5: Feature Selection Results

using sklearn's GridSearchCV [11]. This parameter selection function was used as it performs an exhaustive search for the given parameter ranges for the ideal combination. All other model input parameters which were not tuned were left as the default values. At this stage, randomly sampled 33% of the data was reserved for final testing.

1) Decision Tree Classifier: The decision tree classifier was chosen as the baseline model as this is a powerful but simple model to implement, and the logic within the model can be investigated. The implementation of DecisionTreeClassifier from sklearn was used [12]. The parameters max_depth and max_features were tuned with the discussed method. The max_depth parameter was selected as 18 and the max_features selected as 15.

2) XGBoost Classifier: Next, XGBClassifier was used from the implementation provided by the XGBoost Python library [13]. This classifier uses boosting on multiple decision tree classifiers. This model was chosen because testing ensemble models was a focus of this project and boosting can potentially reduce bias in the model results. The max_depth of the model was tuned to be 15.

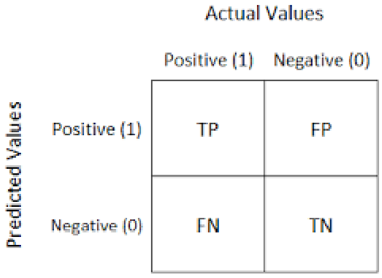
3) Bagging Classifier: The BaggingClassifier implemented in sklearn's ensemble package was used next [14]. A bootstrap aggregating (boosting) model based on decision tree classifiers is a good option as boosting can reduce variance and avoid over fitting our model. The number of estimators, n_estimators, parameter was tuned to be 10.

4) Stacked Classifier: Finally, a stacked classifier was tested using StackingClassifier as implemented by sklearn [15]. The stacked model consisted of a Logistic Regression model, K-Neighbours Classifier, Decision Tree Classifier, and Gaussian Naive Bayes, with the final estimator set to an addition Logistic Regression model. These base models are all implemented by sklearn [16]. This model type was chosen as using multiple types of models and combining the results could give more diverse analysis of the data and utilize the strengths of each. Hyperparameter tuning is not applicable for this model type.

E. Evaluation

We employed a variety of evaluation criteria to compare and analyze the output quality from our models, including the following:

1) Confusion Matrix: Confusion matrices give you the ability to see the various predictions, errors you might make. By comparing the characteristics of correctly and incorrectly classified data, you can learn more about how to apply machine learning more effectively. The confusion matrix format is shown below in Figure 6 and defines True Positive, False Positive, True Negative, and False Negative values.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig. 6: Confusion matrix

2) Accuracy: One parameter for assessing classification models is accuracy. The percentage of predictions that our model correctly predicted is known as accuracy.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

Accuracy can also be determined in terms of positives and negatives for binary classification, as seen below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

3) Precision: Precision speaks about a model's ability to make a correct estimate. When calculating precision, one divides the total number of positive predictions by the proportion of actual positives.

$$Precision = \frac{TP}{TP + FP}$$

4) Recall: The recall is determined as the proportion of Positive samples that were correctly identified as Positive to all Positive samples. The recall measures how well the model can identify positive samples. The more positive samples that are identified, the higher the recall.

$$Recall = \frac{TP}{TP + FN}$$

5) F1 score: Instead of evaluating a model's overall performance like accuracy does, F1score focuses on how well it performs in each class to determine how predictive it is. The precision and recall scores of a model are combined into one statistic called the F1 score.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

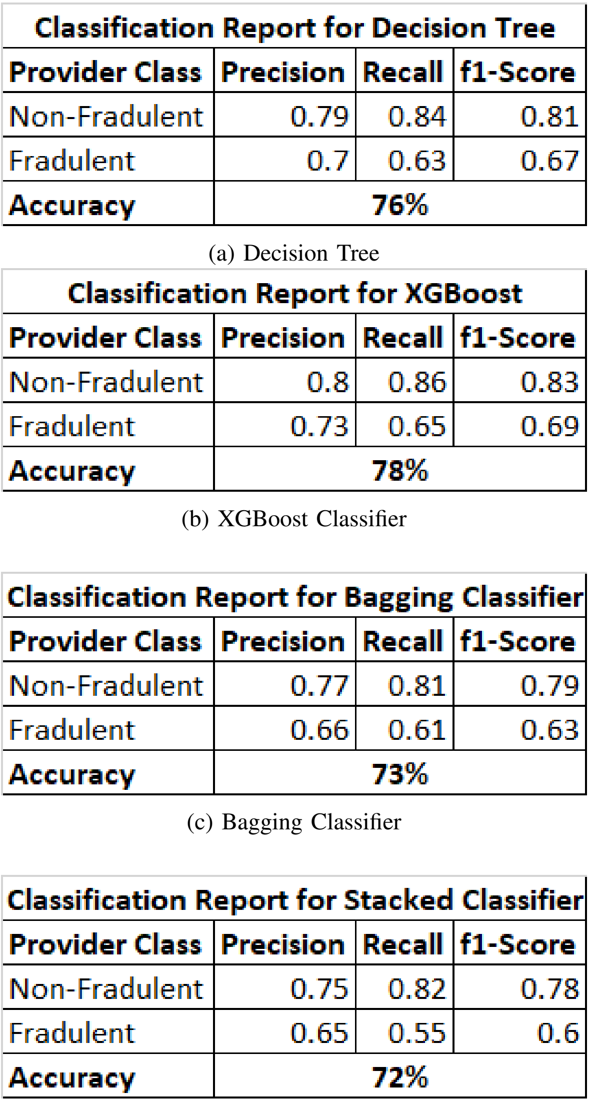
6) Learning Curve: An improvement in a process over time as a result of learning and growing skill is represented visually by a learning curve. According to the learning curve idea, tasks will demand less time and resources as they are carried out more frequently due to the proficiency that is acquired when the procedure is learned. Training was performed with 5-fold cross validation.

IV. Experiments and Results

The evaluation was performed on the 33% reserved data. Here we have the confusion matrix, and classification report with accuracy, precision, recall, and f1-score. We also have the learning curve, scalability, and performance of each model. The confusion matrix is giving the number of TP, TN, FP an FN for each model. The classification report gives the values of precision, recall, f1-score, support, and accuracy. Overall, our models have shown acceptable accuracy. Although all the models have shown almost similar results, the highest recorded was for XGBoost at 78% and the lowest for Stacked Classifier at 72%. Close to XGBoost, we have the Decision Tree classification where the accuracy has been recorded at 76%.

From our learning curves of all our models, as the training set increases, the training score curve, and the cross-validation curve are about to converge at some point. In the Decision tree classification and the XGBoost classification, as the instances are increasing, there is a significant change in the error which explains that the model has a proper learning rate.

The confusion matrix for each model are shown below in Figure 8. The classification report of each model are shown below in Figure 7. The learning curves for each classifier are shown in Figures 9 - 12. The means of the 5-fold cross validation are plotted with the standard deviation shaded.



Classification Report for Decision Tree

Provider Class	Precision	Recall	f1-Score
Non-Fradulent	0.79	0.84	0.81
Fradulent	0.7	0.63	0.67
Accuracy	76%		

(a) Decision Tree

Classification Report for XGBoost

Provider Class	Precision	Recall	f1-Score
Non-Fradulent	0.8	0.86	0.83
Fradulent	0.73	0.65	0.69
Accuracy	78%		

(b) XGBoost Classifier

Classification Report for Bagging Classifier

Provider Class	Precision	Recall	f1-Score
Non-Fradulent	0.77	0.81	0.79
Fradulent	0.66	0.61	0.63
Accuracy	73%		

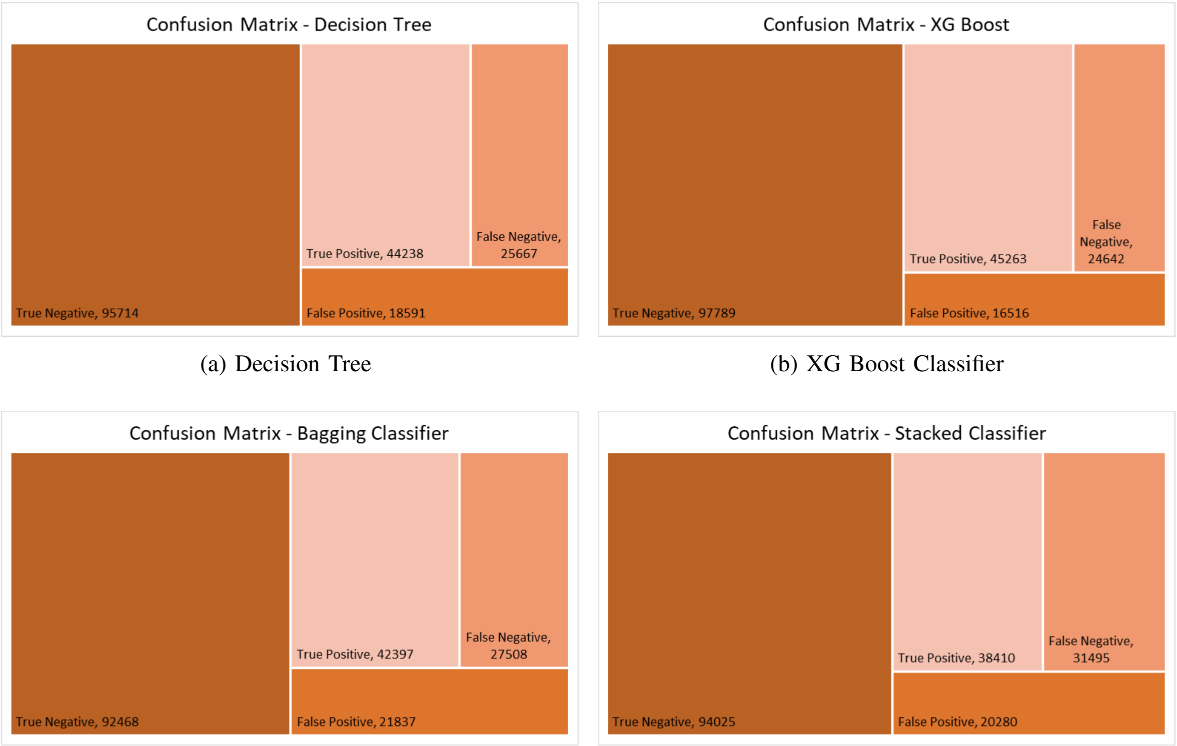
(c) Bagging Classifier

Classification Report for Stacked Classifier

Provider Class	Precision	Recall	f1-Score
Non-Fradulent	0.75	0.82	0.78
Fradulent	0.65	0.55	0.6
Accuracy	72%		

(d) Stacked Classifier

Fig. 7: Classification Reports



(a) Decision Tree

(b) XG Boost Classifier

(c) Bagging Classifier

(d) Stacked Classifier

Fig. 8: Confusion Matrices

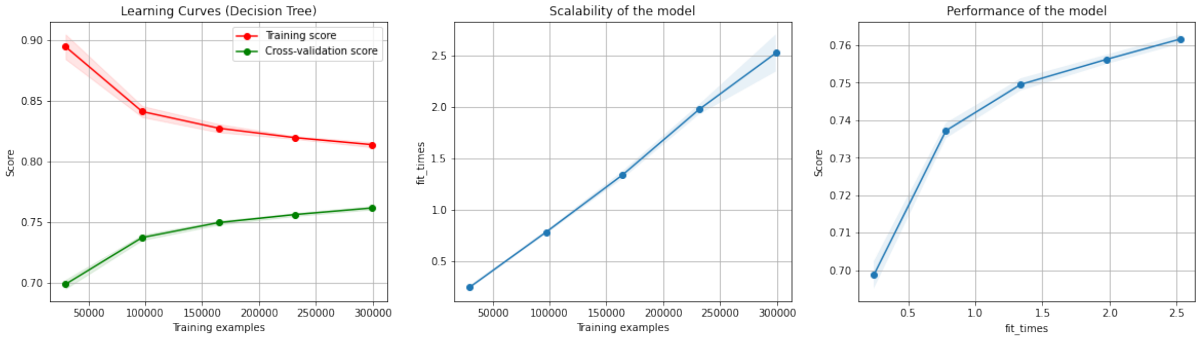


Fig. 9: Decision Tree Classifier Learning Curves

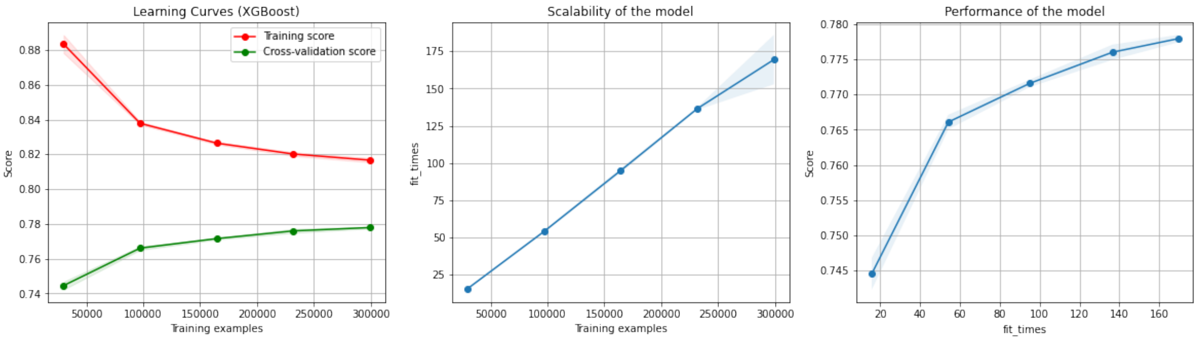


Fig. 10: XGBoost Classifier Learning Curves

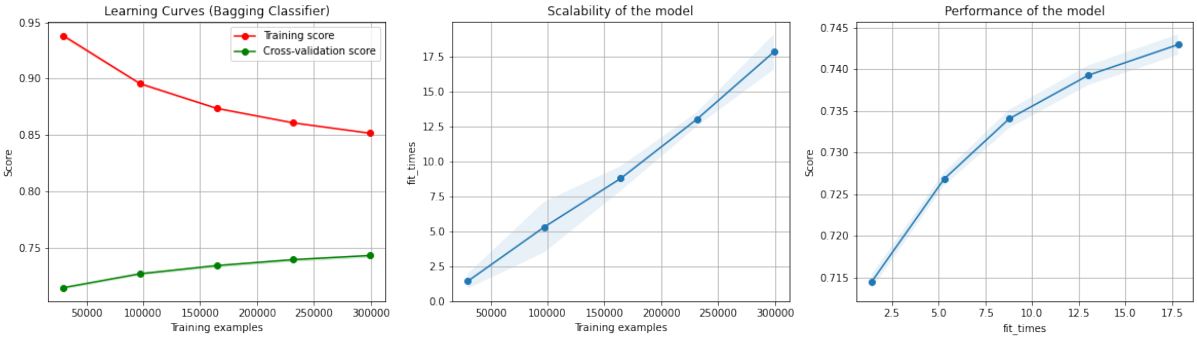


Fig. 11: Bagging Classifier Learning Curves

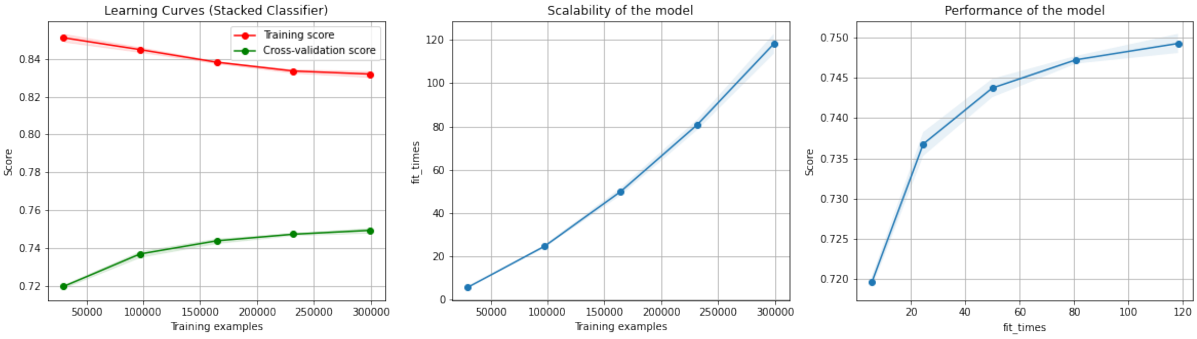


Fig. 12: Stacked Classifier Learning Curves

Our results show that fraudulent claims can predicted with moderate accuracy. While improvements to accuracy would likely be needed before final deployment, the model output solves the business problem. In deployment, the submitted claim details would be automatically input to the fraud detection model and the output potential fraud prediction would flag suspicious claims for further review by the insurance company.

V. Conclusion

From a vast data set of both potentially fraudulent and non-fraudulent claims, we identified the potentially fraudulent claims in this paper, employing the Decision Tree Classifier, XGBoost Classifier, Bagging Classifier, and Stacked Classifier supervised methods. In order to approach our solution, we used the CRISP-DM paradigm, beginning with business understanding, where we attempted to comprehend fraudulent claims and identify the problem statement. We derived new features and identified key features that are useful in spotting possibly fraudulent providers' actions. Our experiments revealed that each algorithm can produce predictions with a good degree of accuracy and usability. Despite the similar accuracies of all models, the stacking classifier model under performed than other models, with a performance of 72% accuracy. According to our investigation, XGBoost was the best-performing model, with an accuracy of 78%, while Decision Tree Classifier and Bagging Classifier were at 76% and 73% respectively.

In future work, we aim to investigate how well unsupervised models perform with the data we have. We would also analyze the K-nearest neighbor algorithm in combination with a genetic algorithm in order to find the ideal weighting of the features. These insights could assist with increasing performance of the prediction models. Gathering more data for training would also be attempted to improve model results.

References

[1] T. L. Leap, "The major health care fraud laws," in *Phantom Billing, Fake Prescriptions, and the High Cost of Medicine: Health Care Fraud and What to Do about It*, T. L. Leap, Ed. Cornell University Press, p. 0. [Online]. Available: https://doi.org/10.7591/cornell/9780801449796.003.0002

[2] https://revcycleintelligence.com/features/how-providers-can-detect-prevent-healthcare-fraud-and-abuse, accessed: 2022-12-20.

[3] G. Brooks, M. Button, and J. Gee, "The scale of health-care fraud: A global evaluation," vol. 25, no. 1, pp. 76–87. [Online]. Available: https://doi.org/10.1057/sj.2011.7

[4] Y. Peng, G. Kou, A. Sabatka, Z. Chen, D. Khazanchi, and Y. Shi, "Application of clustering methods to health insurance fraud detection," in *2006 International Conference on Service Systems and Service Management*, vol. 1, pp. 116–120, ISSN: 2161-1904.

[5] T. Ekin, "Application of bayesian methods in detection of healthcare fraud," vol. 33.

[6] R. Bauder, T. M. Khoshgoftaar, and N. Seliya, "A survey on the state of healthcare upcoding fraud analysis and detection," vol. 17, no. 1, pp. 31–55. [Online]. Available: https://doi.org/10.1007/s10742-016-0154-8

[7] R. Anand Gupta. Medical provider fraud detection. [Online]. Available: https://kaggle.com/code/rohitrox/medical-provider-fraud-detection

[8] ——. Healthcare provider fraud detection analysis. [Online]. Available: https://www.kaggle.com/datasets/rohitrox/healthcare-provider-fraud-detection-analysis

[9] sklearn.feature_selection.SelectKBest. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.feature_selection.SelectKBest.html

[10] sklearn.feature_selection.mutual_info_classif. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.feature_selection.mutual_info_classif.html

[11] sklearn.model_selection.GridSearchCV. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

[12] sklearn.tree.DecisionTreeClassifier. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html

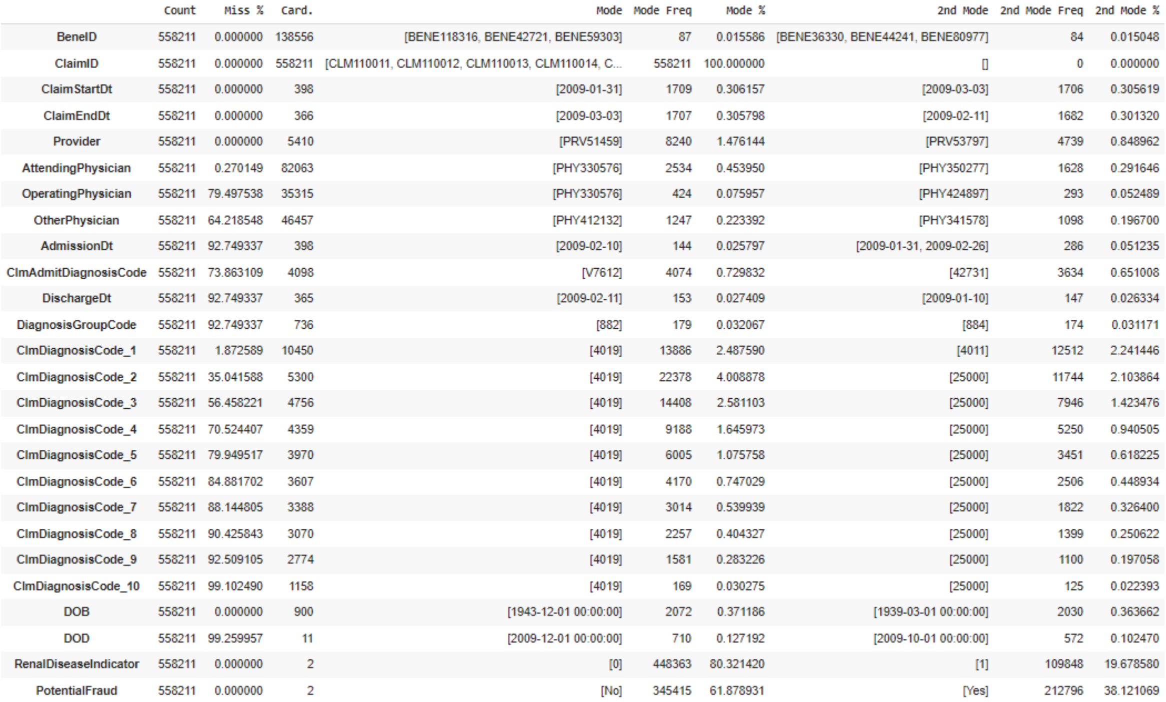
[13] Python API reference — xgboost 1.7.2 documentation. [Online]. Available: https://xgboost.readthedocs.io/en/stable/python/python_api.html

[14] sklearn.ensemble.BaggingClassifier. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html

[15] sklearn.ensemble.StackingClassifier. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.ensemble.StackingClassifier.html

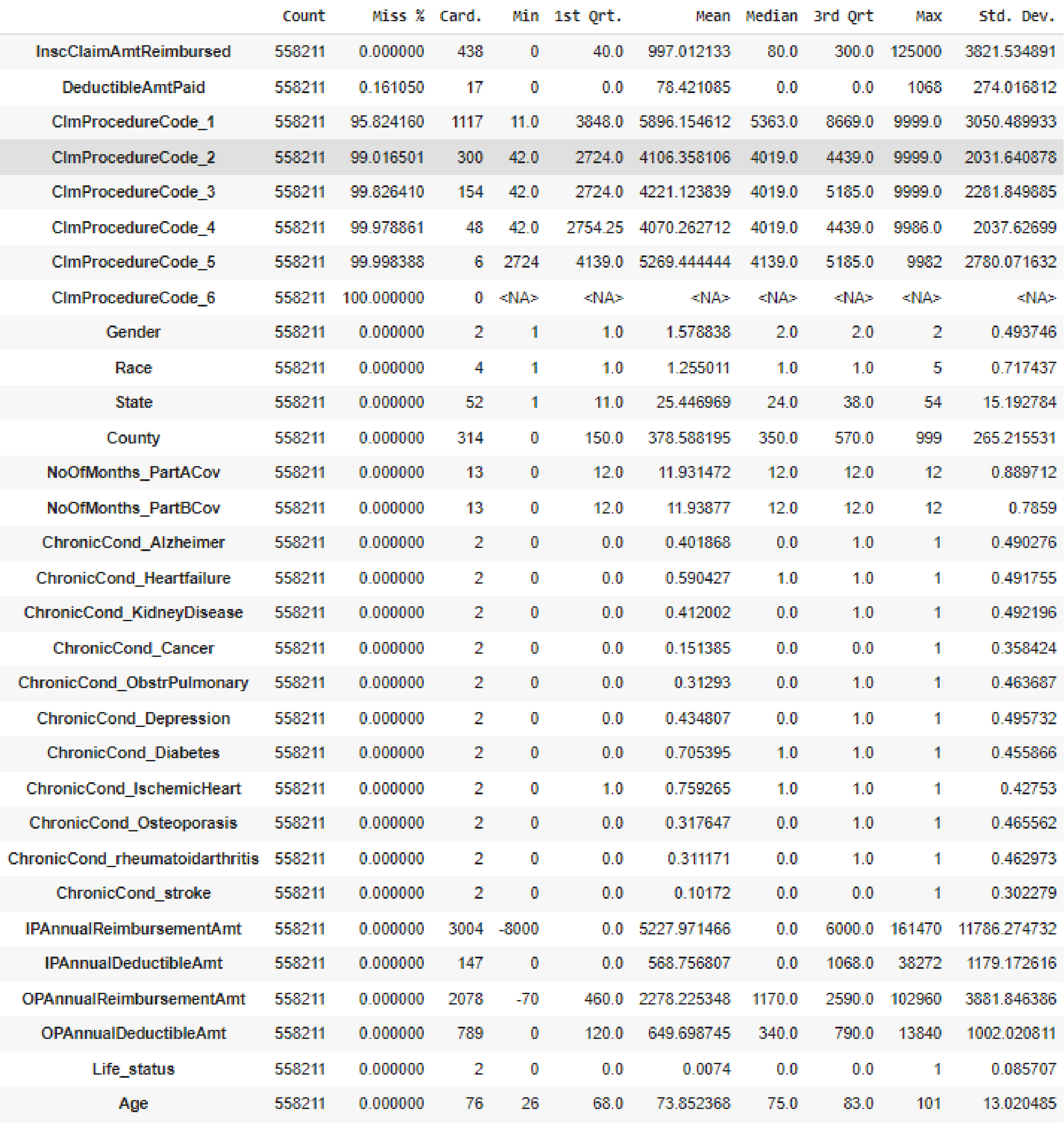
[16] scikit-learn: machine learning in python — scikit-learn 1.2.0 documentation. [Online]. Available: https://scikit-learn.org/stable/

Appendix A: Categorical Feature Report



	Count	Miss %	Card.	Mode	Mode Freq	Mode %	2nd Mode	2nd Mode Freq	2nd Mode %
BeneID	558211	0.000000	138556	[BENE118316, BENE42721, BENE59303]	87	0.015586	[BENE36330, BENE44241, BENE80977]	84	0.015048
ClaimID	558211	0.000000	558211	[CLM110011, CLM110012, CLM110013, CLM110014, C...	558211	100.000000	[]	0	0.000000
ClaimStartDt	558211	0.000000	398	[2009-01-31]	1709	0.306157	[2009-03-03]	1706	0.305619
ClaimEndDt	558211	0.000000	366	[2009-03-03]	1707	0.305798	[2009-02-11]	1682	0.301320
Provider	558211	0.000000	5410	[PRV51459]	8240	1.476144	[PRV53797]	4739	0.848962
AttendingPhysician	558211	0.270149	82063	[PHY330576]	2534	0.453950	[PHY350277]	1628	0.291646
OperatingPhysician	558211	79.497538	35315	[PHY330576]	424	0.075957	[PHY424897]	293	0.052489
OtherPhysician	558211	64.218548	46457	[PHY412132]	1247	0.223392	[PHY341578]	1098	0.196700
AdmissionDt	558211	92.749337	398	[2009-02-10]	144	0.025797	[2009-01-31, 2009-02-26]	286	0.051235
ClmAdmitDiagnosisCode	558211	73.863109	4098	[V7612]	4074	0.729832	[42731]	3634	0.651008
DischargeDt	558211	92.749337	365	[2009-02-11]	153	0.027409	[2009-01-10]	147	0.026334
DiagnosisGroupCode	558211	92.749337	736	[882]	179	0.032067	[884]	174	0.031171
ClmDiagnosisCode_1	558211	1.872589	10450	[4019]	13886	2.487590	[4011]	12512	2.241446
ClmDiagnosisCode_2	558211	35.041588	5300	[4019]	22378	4.008878	[25000]	11744	2.103864
ClmDiagnosisCode_3	558211	56.458221	4756	[4019]	14408	2.581103	[25000]	7946	1.423476
ClmDiagnosisCode_4	558211	70.524407	4359	[4019]	9188	1.645973	[25000]	5250	0.940505
ClmDiagnosisCode_5	558211	79.949517	3970	[4019]	6005	1.075758	[25000]	3451	0.618225
ClmDiagnosisCode_6	558211	84.881702	3607	[4019]	4170	0.747029	[25000]	2506	0.448934
ClmDiagnosisCode_7	558211	88.144805	3388	[4019]	3014	0.539939	[25000]	1822	0.326400
ClmDiagnosisCode_8	558211	90.425843	3070	[4019]	2257	0.404327	[25000]	1399	0.250622
ClmDiagnosisCode_9	558211	92.509105	2774	[4019]	1581	0.283226	[25000]	1100	0.197058
ClmDiagnosisCode_10	558211	99.102490	1158	[4019]	169	0.030275	[25000]	125	0.022393
DOB	558211	0.000000	900	[1943-12-01 00:00:00]	2072	0.371186	[1939-03-01 00:00:00]	2030	0.363662
DOD	558211	99.259957	11	[2009-12-01 00:00:00]	710	0.127192	[2009-10-01 00:00:00]	572	0.102470
RenalDiseaseIndicator	558211	0.000000	2	[0]	448363	80.321420	[1]	109848	19.678580
PotentialFraud	558211	0.000000	2	[No]	345415	61.878931	[Yes]	212796	38.121069

APPENDIX B: CONTINUOUS FEATURE REPORT



	Count	Miss %	Card.	Min	1st Qrt.	Mean	Median	3rd Qrt	Max	Std. Dev.
InscClaimAmtReimbursed	558211	0.000000	438	0	40.0	997.012133	80.0	300.0	125000	3821.534891
DeductibleAmtPaid	558211	0.161050	17	0	0.0	78.421085	0.0	0.0	1068	274.016812
ClmProcedureCode_1	558211	95.824160	1117	11.0	3848.0	5896.154612	5363.0	8669.0	9999.0	3050.489933
ClmProcedureCode_2	558211	99.016501	300	42.0	2724.0	4106.358106	4019.0	4439.0	9999.0	2031.640878
ClmProcedureCode_3	558211	99.826410	154	42.0	2724.0	4221.123839	4019.0	5185.0	9999.0	2281.849885
ClmProcedureCode_4	558211	99.978861	48	42.0	2754.25	4070.262712	4019.0	4439.0	9986.0	2037.62699
ClmProcedureCode_5	558211	99.998388	6	2724	4139.0	5269.444444	4139.0	5185.0	9982	2780.071632
ClmProcedureCode_6	558211	100.000000	0	<NA>	<NA>	<NA>	<NA>	<NA>	<NA>	<NA>
Gender	558211	0.000000	2	1	1.0	1.578838	2.0	2.0	2	0.493746
Race	558211	0.000000	4	1	1.0	1.255011	1.0	1.0	5	0.717437
State	558211	0.000000	52	1	11.0	25.446969	24.0	38.0	54	15.192784
County	558211	0.000000	314	0	150.0	378.588195	350.0	570.0	999	265.215531
NoOfMonths_PartACov	558211	0.000000	13	0	12.0	11.931472	12.0	12.0	12	0.889712
NoOfMonths_PartBCov	558211	0.000000	13	0	12.0	11.93877	12.0	12.0	12	0.7859
ChronicCond_Alzheimer	558211	0.000000	2	0	0.0	0.401868	0.0	1.0	1	0.490276
ChronicCond_Heartfailure	558211	0.000000	2	0	0.0	0.590427	1.0	1.0	1	0.491755
ChronicCond_KidneyDisease	558211	0.000000	2	0	0.0	0.412002	0.0	1.0	1	0.492196
ChronicCond_Cancer	558211	0.000000	2	0	0.0	0.151385	0.0	0.0	1	0.358424
ChronicCond_ObstrPulmonary	558211	0.000000	2	0	0.0	0.31293	0.0	1.0	1	0.463687
ChronicCond_Depression	558211	0.000000	2	0	0.0	0.434807	0.0	1.0	1	0.495732
ChronicCond_Diabetes	558211	0.000000	2	0	0.0	0.705395	1.0	1.0	1	0.455866
ChronicCond_IschemicHeart	558211	0.000000	2	0	1.0	0.759265	1.0	1.0	1	0.42753
ChronicCond_Osteoporasis	558211	0.000000	2	0	0.0	0.317647	0.0	1.0	1	0.465562
ChronicCond_rheumatoidarthritis	558211	0.000000	2	0	0.0	0.311171	0.0	1.0	1	0.462973
ChronicCond_stroke	558211	0.000000	2	0	0.0	0.10172	0.0	0.0	1	0.302279
IPAnnualReimbursementAmt	558211	0.000000	3004	-8000	0.0	5227.971466	0.0	6000.0	161470	11786.274732
IPAnnualDeductibleAmt	558211	0.000000	147	0	0.0	568.756807	0.0	1068.0	38272	1179.172616
OPAnnualReimbursementAmt	558211	0.000000	2078	-70	460.0	2278.225348	1170.0	2590.0	102960	3881.846386
OPAnnualDeductibleAmt	558211	0.000000	789	0	120.0	649.698745	340.0	790.0	13840	1002.020811
Life_status	558211	0.000000	2	0	0.0	0.0074	0.0	0.0	1	0.085707
Age	558211	0.000000	76	26	68.0	73.852368	75.0	83.0	101	13.020485