RDS Final Project: Justin Chui and Brad Zhang

Background

The purpose of this ADS is to develop a predictive model that accurately classifies risk in life insurance applicants using a more automated approach. The goal is to make the life insurance application process quicker and less labor-intensive for new and existing customers to get a quote while maintaining privacy boundaries. Our project will be focusing on Prudential's dataset that contains over a hundred variables describing attributes of life insurance applicants, including normalized variables related to the product applied for, employment history, insurance history, family history, medical history, age, BMI, weight, and height etc. The target variable will be the “Response” column, which is an ordinal measure of risk with 8 levels. The ultimate objective is to help Prudential better understand the predictive power of the data points in the existing assessment, enabling the company to significantly streamline the process.

Input and Output

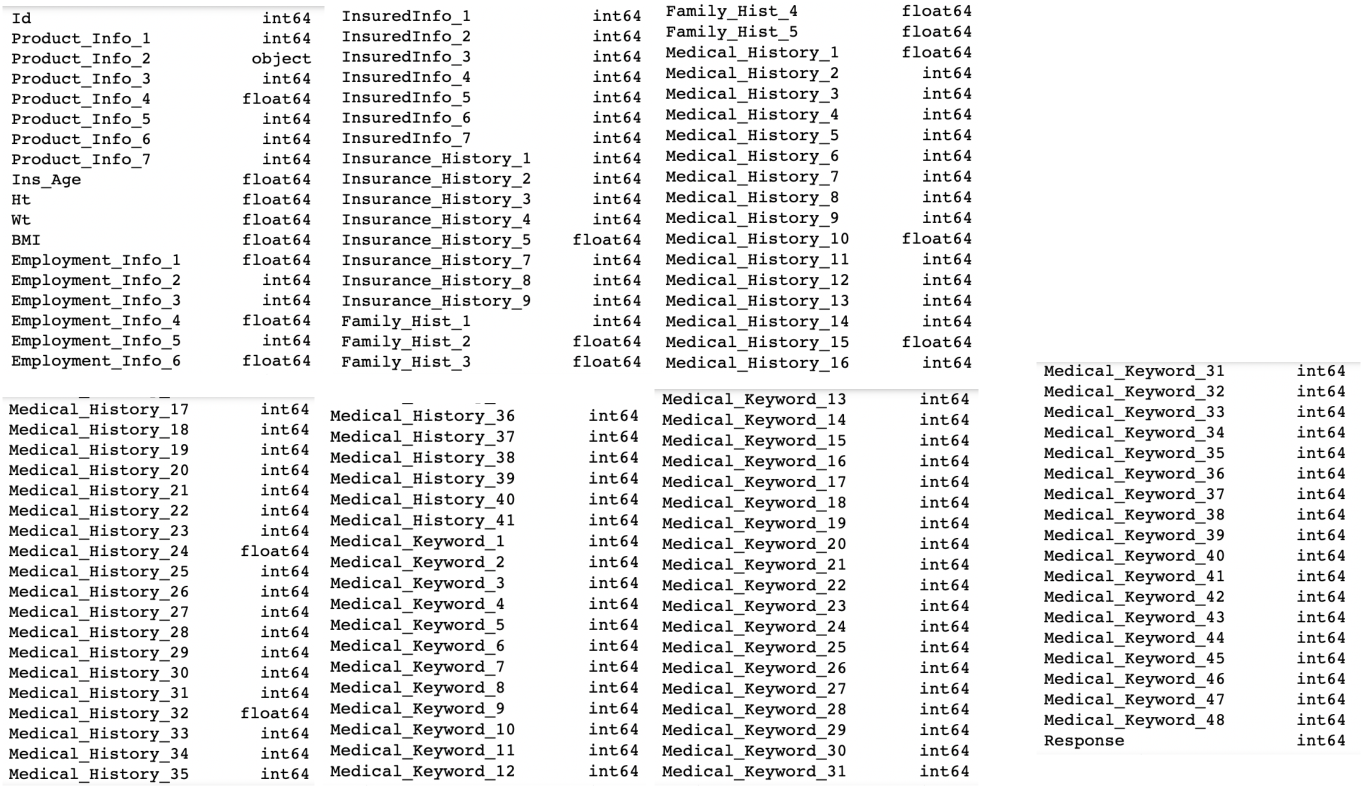
The data, labeled “train.csv”, was uploaded to Kaggle by Prudential.

This dataset is in the Kaggle public domain and is used to train the “test.csv” for the target variable. The goal is to predict the level of risk a patient will have when applying for life insurance. The input data given include:

1. Id: A unique identifier associated with an application.
2. Product_Info_1-7: A set of normalized variables relating to the product applied for.
3. Ins_Age: Normalized age of applicant.
4. Ht: Normalized height of applicant.
5. Wt: Normalized weight of applicant.
6. BMI: Normalized BMI of applicant.
7. Employment_Info_1-6: A set of normalized variables relating to the employment history of the applicant.
8. InsuredInfo_1-6: A set of normalized variables providing information about the applicant.
9. Insurance_History_1-9: A set of normalized variables relating to the insurance history of the applicant.
10. Family_Hist_1-5: A set of normalized variables relating to the family history of the applicant.
11. Medical_History_1-41: A set of normalized variables relating to the medical history of the applicant.
12. Medical_Keyword_1-48: A set of dummy variables relating to the presence of/absence of a medical keyword being associated with the application.

The ADS uses all 128 features in its analysis. Within those 128 features, Product_Info_4, Ins_Age, Ht, Wt, BMI, Employment_Info_1, Employment_Info_4, Employment_Info_6, Insurance_History_5, Family_Hist_2, Family_Hist_3, Family_Hist_4, Family_Hist_5 are all continuous variables, and Medical_History_1, Medical_History_10, Medical_History_15, Medical_History_24, Medical_History_32 are discrete, and all the other ones are categorical.

Describe each feature’s data type:



```
Id                     int64
Product_Info_1         int64
Product_Info_2        object
Product_Info_3         int64
Product_Info_4       float64
Product_Info_5         int64
Product_Info_6         int64
Product_Info_7         int64
Ins_Age              float64
Ht                   float64
Wt                   float64
BMI                  float64
Employment_Info_1    float64
Employment_Info_2      int64
Employment_Info_3      int64
Employment_Info_4    float64
Employment_Info_5      int64
Employment_Info_6    float64
```

```
InsuredInfo_1            int64
InsuredInfo_2            int64
InsuredInfo_3            int64
InsuredInfo_4            int64
InsuredInfo_5            int64
InsuredInfo_6            int64
InsuredInfo_7            int64
Insurance_History_1      int64
Insurance_History_2      int64
Insurance_History_3      int64
Insurance_History_4      int64
Insurance_History_5    float64
Insurance_History_7      int64
Insurance_History_8      int64
Insurance_History_9      int64
Family_Hist_1            int64
Family_Hist_2          float64
Family_Hist_3          float64
```

```
Family_Hist_4          float64
Family_Hist_5          float64
Medical_History_1      float64
Medical_History_2        int64
Medical_History_3        int64
Medical_History_4        int64
Medical_History_5        int64
Medical_History_6        int64
Medical_History_7        int64
Medical_History_8        int64
Medical_History_9        int64
Medical_History_10     float64
Medical_History_11       int64
Medical_History_12       int64
Medical_History_13       int64
Medical_History_14       int64
Medical_History_15     float64
Medical_History_16       int64
```

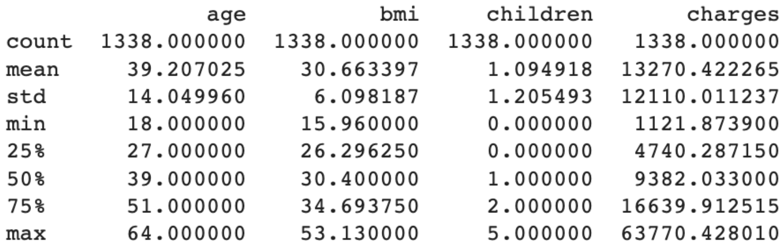
```
Medical_History_17       int64
Medical_History_18       int64
Medical_History_19       int64
Medical_History_20       int64
Medical_History_21       int64
Medical_History_22       int64
Medical_History_23       int64
Medical_History_24     float64
Medical_History_25       int64
Medical_History_26       int64
Medical_History_27       int64
Medical_History_28       int64
Medical_History_29       int64
Medical_History_30       int64
Medical_History_31       int64
Medical_History_32     float64
Medical_History_33       int64
Medical_History_34       int64
Medical_History_35       int64
```

```
Medical_History_36       int64
Medical_History_37       int64
Medical_History_38       int64
Medical_History_39       int64
Medical_History_40       int64
Medical_History_41       int64
Medical_Keyword_1        int64
Medical_Keyword_2        int64
Medical_Keyword_3        int64
Medical_Keyword_4        int64
Medical_Keyword_5        int64
Medical_Keyword_6        int64
Medical_Keyword_7        int64
Medical_Keyword_8        int64
Medical_Keyword_9        int64
Medical_Keyword_10       int64
Medical_Keyword_11       int64
Medical_Keyword_12       int64
```

```
Medical_Keyword_13       int64
Medical_Keyword_14       int64
Medical_Keyword_15       int64
Medical_Keyword_16       int64
Medical_Keyword_17       int64
Medical_Keyword_18       int64
Medical_Keyword_19       int64
Medical_Keyword_20       int64
Medical_Keyword_21       int64
Medical_Keyword_22       int64
Medical_Keyword_23       int64
Medical_Keyword_24       int64
Medical_Keyword_25       int64
Medical_Keyword_26       int64
Medical_Keyword_27       int64
Medical_Keyword_28       int64
Medical_Keyword_29       int64
Medical_Keyword_30       int64
Medical_Keyword_31       int64
```

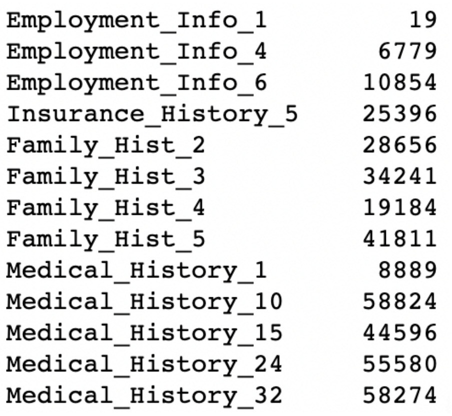
```
Medical_Keyword_31       int64
Medical_Keyword_32       int64
Medical_Keyword_33       int64
Medical_Keyword_34       int64
Medical_Keyword_35       int64
Medical_Keyword_36       int64
Medical_Keyword_37       int64
Medical_Keyword_38       int64
Medical_Keyword_39       int64
Medical_Keyword_40       int64
Medical_Keyword_41       int64
Medical_Keyword_42       int64
Medical_Keyword_43       int64
Medical_Keyword_44       int64
Medical_Keyword_45       int64
Medical_Keyword_46       int64
Medical_Keyword_47       int64
Medical_Keyword_48       int64
Response                 int64
```

Describe the data distribution:



```
               age          bmi     children       charges
count  1338.000000  1338.000000  1338.000000   1338.000000
mean     39.207025    30.663397     1.094918  13270.422265
std      14.049960     6.098187     1.205493  12110.011237
min      18.000000    15.960000     0.000000   1121.873900
25%      27.000000    26.296250     0.000000   4740.287150
50%      39.000000    30.400000     1.000000   9382.033000
75%      51.000000    34.693750     2.000000  16639.912515
max      64.000000    53.130000     5.000000  63770.428010
```

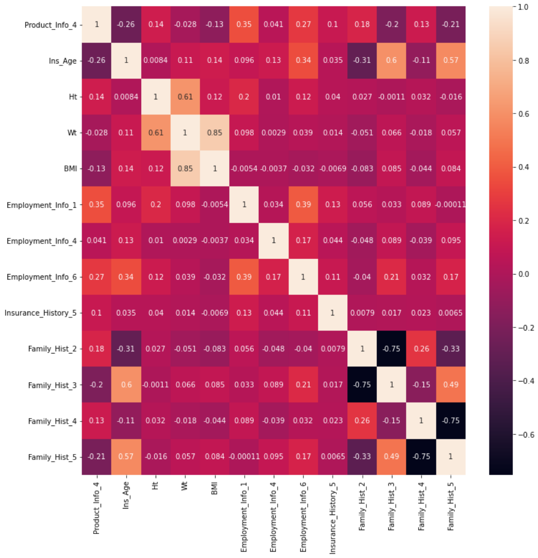
Count the number of missing data for each feature:
The features shown below all have missing data, but we have altered the null values to 0 for computation. While the use of 0 can provide a temporary solution for computational purposes, it is essential to be aware of its potential impact on the distribution, characteristics, and interpretation of the data. It is important to keep in mind that replacing missing values with 0 can distort the distribution and characteristics of the data.



```
Employment_Info_1          19
Employment_Info_4        6779
Employment_Info_6       10854
Insurance_History_5     25396
Family_Hist_2           28656
Family_Hist_3           34241
Family_Hist_4           19184
Family_Hist_5           41811
Medical_History_1        8889
Medical_History_10      58824
Medical_History_15      44596
Medical_History_24      55580
Medical_History_32      58274
```

Since our dataset has both discrete and continuous variables, we have separated them and analyzed each one of them.

Pairwise correlations between the continuous features:

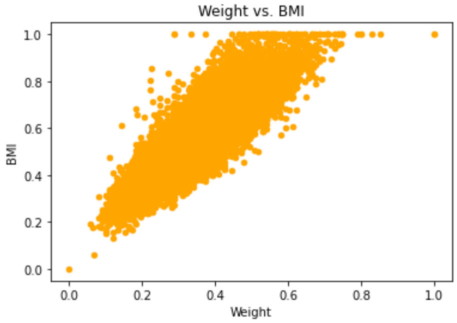


	Product_Info_4	Ins_Age	Ht	Wt	BMI	Employment_Info_1	Employment_Info_4	Employment_Info_6	Insurance_History_5	Family_Hist_2	Family_Hist_3	Family_Hist_4	Family_Hist_5
Product_Info_4	1	-0.26	0.14	-0.028	-0.13	0.35	0.041	0.27	0.1	0.18	-0.2	0.13	-0.21
Ins_Age	-0.26	1	0.0084	0.11	0.14	0.096	0.13	0.34	0.035	-0.31	0.6	-0.11	0.57
Ht	0.14	0.0084	1	0.61	0.12	0.2	0.01	0.12	0.04	0.027	-0.0011	0.032	-0.016
Wt	-0.028	0.11	0.61	1	0.85	0.098	0.0029	0.039	0.014	-0.051	0.066	-0.018	0.057
BMI	-0.13	0.14	0.12	0.85	1	-0.0054	-0.0037	-0.032	-0.0069	-0.083	0.085	-0.044	0.084
Employment_Info_1	0.35	0.096	0.2	0.098	-0.0054	1	0.034	0.39	0.13	0.056	0.033	0.089	-0.00011
Employment_Info_4	0.041	0.13	0.01	0.0029	-0.0037	0.034	1	0.17	0.044	-0.048	0.089	-0.039	0.095
Employment_Info_6	0.27	0.34	0.12	0.039	-0.032	0.39	0.17	1	0.11	-0.04	0.21	0.032	0.17
Insurance_History_5	0.1	0.035	0.04	0.014	-0.0069	0.13	0.044	0.11	1	0.0079	0.017	0.023	0.0065
Family_Hist_2	0.18	-0.31	0.027	-0.051	-0.083	0.056	-0.048	-0.04	0.0079	1	-0.75	0.26	-0.33
Family_Hist_3	-0.2	0.6	-0.0011	0.066	0.085	0.033	0.089	0.21	0.017	-0.75	1	-0.15	0.49
Family_Hist_4	0.13	-0.11	0.032	-0.018	-0.044	0.089	-0.039	0.032	0.023	0.26	-0.15	1	-0.75
Family_Hist_5	-0.21	0.57	-0.016	0.057	0.084	-0.00011	0.095	0.17	0.0065	-0.33	0.49	-0.75	1

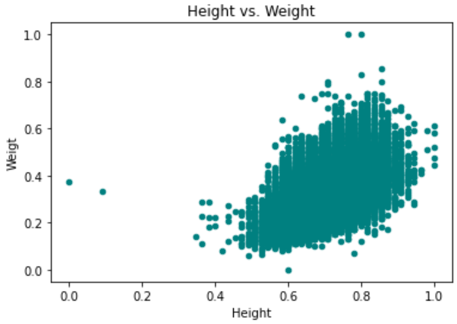
From the correlation heatmap above, we see that there are some pairs of stronger correlations.

- Weight VS Height
- BMI VS Weight
- Product Info 4 VS Employment Info 1
- Age VS Family History 3
- Family History 3 VS Family History 4
- Family History 2 VS Family History 3

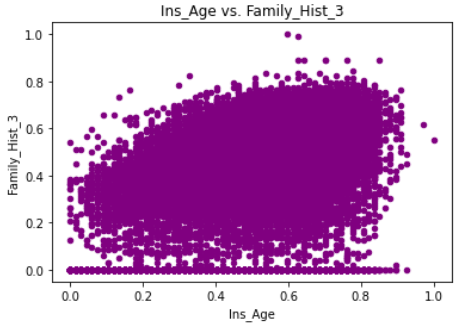
Out of these higher correlation pairs, we believe that only the highlighted ones are meaningful.



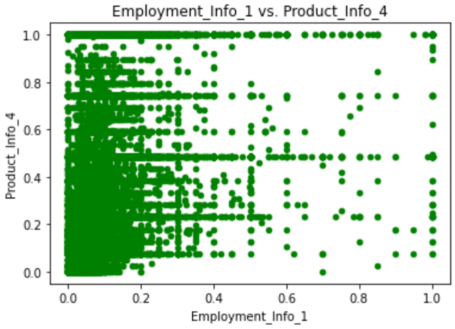
Since BMI is calculated by weight/height^2, we definitely expect to see a positive correlation between BMI and weight. However, it's also worth noting that BMI doesn't take into account body composition like muscle and fat, so it may not be the best indicator or feature of health for everyone.



This is natural because taller people would generally weigh more than shorter people. As a result, we would expect to see a positive correlation.

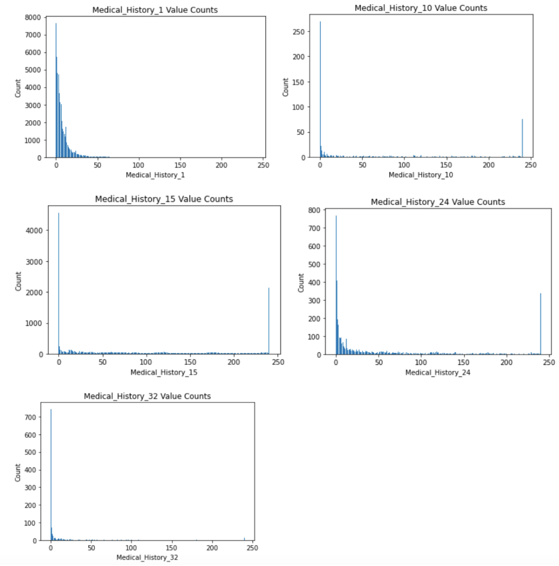


Family History 3 represents the data of certain family historical medical conditions such as diabetes, heart disease, and stroke. It's possible that age and family history of cancer are correlated, as severe medical conditions are more common among older people.



Product 4 Info represents the type of insurance product that a customer purchased, and Employment Info 1 represents the level of employment of the customer. It could be possible that there is a correlation between the type of insurance product and the level of employment of the customer. For example, people with higher incomes might purchase as people with higher incomes may be more likely to purchase more expensive insurance products.

After analyzing the continuous variables, we have decided to take a dive into the discrete variables.



For Medical_History_1, the values range from 0 to 178, and the most common value is 1.0 with a count of 10,111. The count then decreases as the value increases.

For Medical_History_10, the values range from 0 to 240, and the most common value is 240.0 with a count of 363. The count for values less than 10 is relatively low.

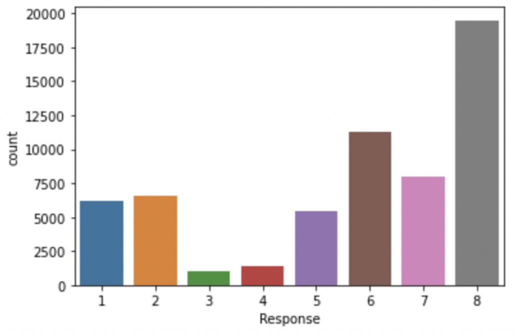
For Medical_History_15, the values range from 0 to 240, and the most common value is 240.0 with a count of 6,140. The count for values less than 10 is relatively low.

For Medical_History_24, the values range from 0 to 240, and the most common value is 0.0 with a count of 1,010. The count for values greater than 100 is relatively low.

For Medical_History_32, the values range from 0 to 240, and the most common value is 0.0 with a count of 979. The count for values greater than 10 is relatively low.

Since Prudential did not give the specific details of what each medical history category represents, it's hard to provide specific insights. However, we believe that the higher the count for a certain category, the more common that medical condition is in the dataset.

At last, we look at the target variable - Response.



As we can see here, there is a class imbalance where Response 8 has highest values and 3 has the least. This class imbalance can be a problem for some machine learning models that the ADS does as they may become biased towards the majority class and struggle to accurately predict the minority class.

Hence, the user first adjusts the class imbalance to continue to train the models. To do this, they set it up as, if the value of Response is between 0 and 7 (inclusive), map it to 0. If the value of Response is 8, map it to 1. By doing this, the author is combining the original 8 categories of Response into 2 categories: 0, 1. This helps to address the class imbalance issue, as the new categories are more balanced than the original ones.

Output of the System

The ADS uses both unsupervised and supervised machine learning tools, including Random Forest, XGBoost, Logistic Regression, and Gradient Boosts.

Random Forest: Class label of risk score (1-8, but author modified it to 1-7 as 0, and 8 as 1)
XGBoost: Class label of risk score (1-8, but author modified it to 1-7 as 0, and 8 as 1)
Logistic Regression: Class label of risk score (1-8, but author modified it to 1-7 as 0, and 8 as 1)
Gradient Boosts: Class label of risk score (1-8, but author modified it to 1-7 as 0, and 8 as 1)

Implementation and Validation

The implementation first starts with data cleaning and preprocessing steps. The dataset was first checked for duplicate rows, and removed; then checked for the percentage of missing values in each column. It was found that more than 40% missing values were dropped, and the remaining missing values were filled with the mean of the column. After dropping null values and duplicate data, a correlation analysis was performed to check the relationship between features. Highly correlated features were identified but were not removed or transformed as the model to be used is a tree-based model which is not affected much by correlation.

After running a correlation analysis, a feature selection was conducted where it was found that “Product_Info_2” contains no important information, hence this feature was also dropped. Next, to set up the data for machine learning algorithms, the dataset was split into training and testing sets with a 75:25 ratio, and no scaling was needed to be performed as the model is a tree-based model that doesn’t require feature scaling.

Information about the implementation of the system

The system utilizes supervised machine learning models including logistic regression, XGBoost, gradient boosting, and random forest to predict the Response category. The ADS also conducts checks to evaluate whether the model is performing well or not, with metrics including train and train ROC (Receiver Operator Characteristic), test and train accuracy, test and train log loss, F-Score, precision, and recall.

Logistic regression: The model learns a linear relationship between the various attributes of the applicant and the probability of each risk level, and uses a logistic function to convert the linear combination of attributes into probability scores for each risk level. The model first performs a parameter grid and grid search in order to find the best model configuration based on the provided evaluation metric. Then, the code evaluates the logistic regression model with the best parameters on the training and test datasets. It calculates the performance metrics, and also provides feature importance values trained from logistic regression models. It ranks the features based on their respective coefficients. Positive coefficients indicate features that positively influence the risk level prediction, while negative coefficients indicate features that negatively influence the prediction. The top feature importance scores predicted by logistic regression are Ht (height), Medical_Keyword_41, Family_Hist_3, Medical_History_4, and Medical_History_20.

Random forest: The model first defines a grid of hyperparameters for the random forest classifier. Hyperparameters include the number of trees, maximum depth of trees, minimum number of samples required to split a node, and also the minimum number of samples required at each leaf node. Then, grid search is performed to find the best model based on the specified evaluation metric. After finding the optimal model, it compares the metrics for both training and testing sets, and calculates the feature importance.

It was predicted that BMI, weight (Wt), Medical_History_23, Medical_History_4, and Medical_Keyword_15. These features have higher importance scores compared to other features in the dataset.

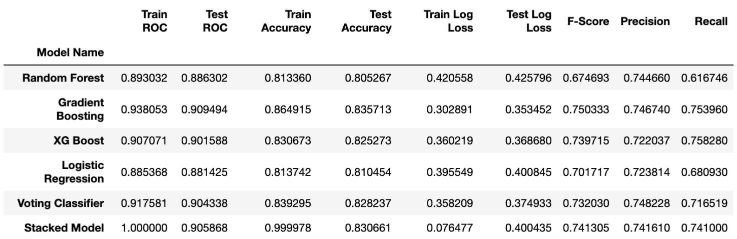
Gradient boosting: The model first starts with a grid search to tune the hyper parameter to optimize the performance of the algorithm. Then evaluate the trained model using the test set and calculate relevant metrics like accuracy and precision. After the model has been trained and evaluated with the desired performance criteria, it was found that BMI, weight, Medical_History_23, Medical_History_4 and Medical_Keyword_15 seem to be the most important 5 features according to gradient boosting.

XGBoost: Similar to gradient boosting, XGBoost is a more regularized form. The model first performs a grid search to find all possible combinations of hyperparameters that optimizes the model. Then, the model evaluates its performance using various metrics like accuracy and F1 score, and lastly determines the feature importance score, with the top features being "Medical_History_23," "BMI," and "Medical_History_4.

Stacked model: This model combines the predictions of multiple base models (Logistic Regression, XGBoost, and Gradient Boosting) using a meta-model (Random Forest). The base models are trained on the training data, and their predictions on the validation data are used as input features for the meta-model. The purpose of the meta-model is to learn how to best combine the predictions of the base models to make the final prediction. The stacked model achieved an accuracy of 0.999978 on the training dataset and an accuracy of 0.830661 on the test dataset, which indicates that that the model is performing very well on the training data, but there might be some overfitting because the test accuracy is slightly lower.

As aforementioned, the ADS is validated by conducting performance checks with each machine learning algorithm. These metrics were calculated for both the training and testing sets, to ensure that the models were not overfitting to the training data. The ADS also utilized a stacked model approach that combines the predictions of multiple base models to improve the overall performance. The stacked model achieved high accuracy on the training set but slightly lower accuracy on the testing set, indicating some degree of overfitting.

The ADS met its stated goals of accurately classifying the risk of life insurance applicants and streamlining the application process. The models developed were able to accurately predict the risk level of applicants based on their data points, with high performance on both the training and testing sets. Additionally, the ADS was able to identify the most important features in the dataset, providing insights into the factors that contribute to risk assessment. Overall, the ADS provides an automated approach to risk assessment that can save time and resources for both the insurance company and the applicants.

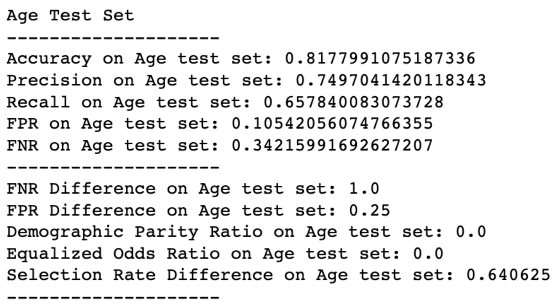


	Train ROC	Test ROC	Train Accuracy	Test Accuracy	Train Log Loss	Test Log Loss	F-Score	Precision	Recall
Model Name									
Random Forest	0.893032	0.886302	0.813360	0.805267	0.420558	0.425796	0.674693	0.744660	0.616746
Gradient Boosting	0.938053	0.909494	0.864915	0.835713	0.302891	0.353452	0.750333	0.746740	0.753960
XG Boost	0.907071	0.901588	0.830673	0.825273	0.360219	0.368680	0.739715	0.722037	0.758280
Logistic Regression	0.885368	0.881425	0.813742	0.810454	0.395549	0.400845	0.701717	0.723814	0.680930
Voting Classifier	0.917581	0.904338	0.839295	0.828237	0.358209	0.374933	0.732030	0.748228	0.716519
Stacked Model	1.000000	0.905868	0.999978	0.830661	0.076477	0.400435	0.741305	0.741610	0.741000

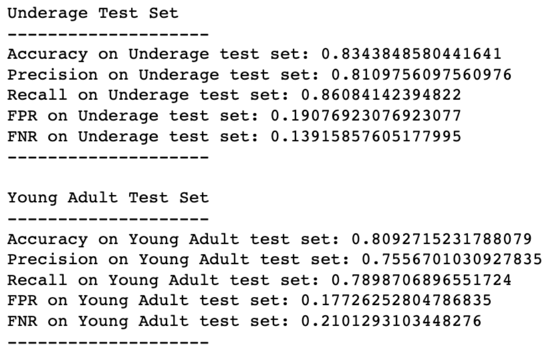
Outcomes

Although a variety of models were used, we want to determine whether using any of these ADS's would result in not only accuracy, but also fairness for different subpopulations. Thus, in order to put this to the test, we determined accuracy as well as fairness by training a baseline Random Forest model for the entire dataset and proceeded to analyze its fairness metrics such as Demographic Parity Ratio, Equalized Odds Ratio, Selection Rate Difference, and others. In addition, we trained baseline Random Forest models for subsets of the dataset based on two different sensitive features: Age and BMI.

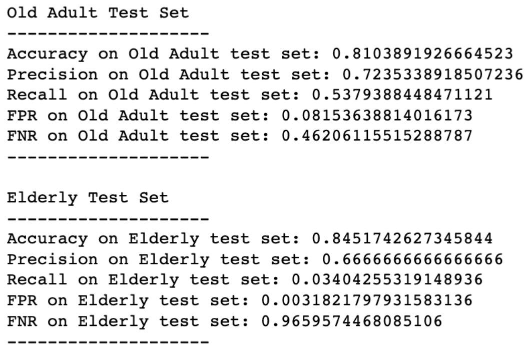
For conducting the model on the sensitive feature age, we set 4 thresholds in order to extract the subpopulations. These categories are Underage (0-17), Young Adult (18-29), Old Adult (30-64), and Elderly (65+). After setting these thresholds, we extracted their respective data and trained a random forest classifier for each group as well as the whole dataset. For the whole dataset, we calculated accuracy and fairness metrics consisting of precision, recall, FNR, FPR, FNRD, FPRD, DPR, EOR, and SRD. For the subgroup models, we calculated precision, recall, FNR, and FPR. Our results are presented below:



```
Age Test Set
--------------------
Accuracy on Age test set: 0.8177991075187336
Precision on Age test set: 0.7497041420118343
Recall on Age test set: 0.657840083073728
FPR on Age test set: 0.10542056074766355
FNR on Age test set: 0.34215991692627207
--------------------
FNR Difference on Age test set: 1.0
FPR Difference on Age test set: 0.25
Demographic Parity Ratio on Age test set: 0.0
Equalized Odds Ratio on Age test set: 0.0
Selection Rate Difference on Age test set: 0.640625
--------------------
```



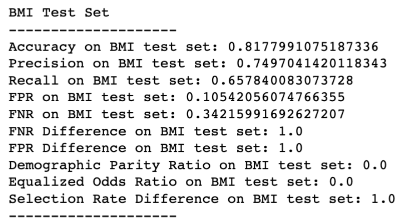
```
Underage Test Set
--------------------
Accuracy on Underage test set: 0.8343848580441641
Precision on Underage test set: 0.8109756097560976
Recall on Underage test set: 0.86084142394822
FPR on Underage test set: 0.19076923076923077
FNR on Underage test set: 0.13915857605177995
--------------------

Young Adult Test Set
--------------------
Accuracy on Young Adult test set: 0.8092715231788079
Precision on Young Adult test set: 0.7556701030927835
Recall on Young Adult test set: 0.7898706896551724
FPR on Young Adult test set: 0.17726252804786835
FNR on Young Adult test set: 0.2101293103448276
--------------------
```



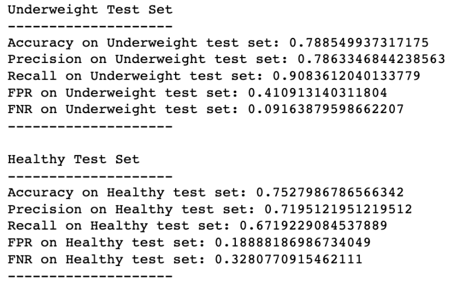
```
Old Adult Test Set
--------------------
Accuracy on Old Adult test set: 0.8103891926664523
Precision on Old Adult test set: 0.7235338918507236
Recall on Old Adult test set: 0.5379388448471121
FPR on Old Adult test set: 0.08153638814016173
FNR on Old Adult test set: 0.46206115515288787
--------------------

Elderly Test Set
--------------------
Accuracy on Elderly test set: 0.8451742627345844
Precision on Elderly test set: 0.6666666666666666
Recall on Elderly test set: 0.03404255319148936
FPR on Elderly test set: 0.0031821797931583136
FNR on Elderly test set: 0.9659574468085106
--------------------
```

From the above results, we can see that accuracy and precision is relatively high and similar among the entire age dataset, underage, young adult, old adult, and elderly datasets with all having accuracy values above 0.80 and almost all precisions above 0.7 except for elderly. Notably, the underage subpopulation achieved the highest accuracy and precision with the highest recall as well, having values of 0.8344, 0.8110, and 0.8608, respectively. Looking at the recall on other subpopulations, we can see that Young Adults possessed the second highest with 0.7899 and Elderly had the lowest with 0.034. Overall, the entire dataset showed promising recall with a value of 0.6578. The FPR and FNR rates are also relatively low amongst all the datasets with Old Adults having the lowest FPR of 0.0815 and Underage having the lowest FNR of 0.1392. The entire dataset also possessed low values for FPR and FNR which is good. If we take a look at the fairness metrics on the entire dataset, we can see that the FNRD and FPRD have values 1.0 and 0.25, respectively. Lastly, the DPR, EOR, and SRD produced values of 0.0, 0.0, and 0.640625, respectively.

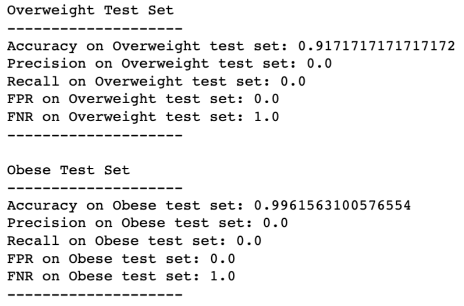
Next, we decided to repeat this process but on the sensitive feature BMI (Body Mass Index). For this feature, we once again set 4 thresholds to extract their respective subpopulations. These categories are Underweight (<18.5), Healthy (>=18.5 and <25), Overweight (>=25 and <30), and Obese (>=30). Once we finished classifying the thresholds, we extracted the respective datasets for the subpopulations and trained a random forest classifier on each group as well as the entire dataset. Like previously, we calculated the accuracy and fairness metrics for the entire dataset and only accuracy metrics for subgroup models. The results are presented below:



```
BMI Test Set
--------------------
Accuracy on BMI test set: 0.8177991075187336
Precision on BMI test set: 0.7497041420118343
Recall on BMI test set: 0.657840083073728
FPR on BMI test set: 0.10542056074766355
FNR on BMI test set: 0.34215991692627207
FNR Difference on BMI test set: 1.0
FPR Difference on BMI test set: 1.0
Demographic Parity Ratio on BMI test set: 0.0
Equalized Odds Ratio on BMI test set: 0.0
Selection Rate Difference on BMI test set: 1.0
--------------------
```



```
Underweight Test Set
--------------------
Accuracy on Underweight test set: 0.788549937317175
Precision on Underweight test set: 0.7863346844238563
Recall on Underweight test set: 0.9083612040133779
FPR on Underweight test set: 0.410913140311804
FNR on Underweight test set: 0.09163879598662207
--------------------

Healthy Test Set
--------------------
Accuracy on Healthy test set: 0.7527986786566342
Precision on Healthy test set: 0.7195121951219512
Recall on Healthy test set: 0.6719229084537889
FPR on Healthy test set: 0.18888186986734049
FNR on Healthy test set: 0.3280770915462111
--------------------
```



```
Overweight Test Set
--------------------
Accuracy on Overweight test set: 0.9171717171717172
Precision on Overweight test set: 0.0
Recall on Overweight test set: 0.0
FPR on Overweight test set: 0.0
FNR on Overweight test set: 1.0
--------------------

Obese Test Set
--------------------
Accuracy on Obese test set: 0.9961563100576554
Precision on Obese test set: 0.0
Recall on Obese test set: 0.0
FPR on Obese test set: 0.0
FNR on Obese test set: 1.0
--------------------
```

From the above results, we can see that accuracy was significantly high in Overweight and Obese test sets, however, if we note their precisions we can see that these subgroups produced 0.0 values. Now if we look at the other subgroups, we have Underweight and Healthy producing accuracies of 0.7886 and 0.7528 respectively and their precisions are 0.7863 and 0.7195 respectively. The entire test set produced an accuracy of 0.8188 and a precision of 0.7497. Overall, these accuracies are relatively high and their precisions are also relatively high with the false positive rates and false negative rates being relatively low as well. If we take a look at the fairness metrics for the overall BMI dataset, we can see that the FNRD and FPRD are 1.0 and 1.0, respectively and DPR, EOR, and SRD being 0.0, 0.0, and 1.0.

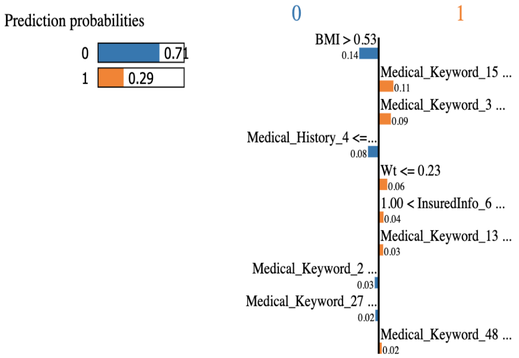
To summarize the results from above, it can be inferred that the data present for body mass index as well as age are consistent and fair across their subpopulations and both of these sensitive features presented relatively high accuracy, precision, and recall values. Thus, the model is able to successfully and accurately predict the risk of an applicant when applying for insurance. To go over the definitions of precision, we know that precision represents the proportion of true positives identified by our model out of all the instances that our model classified as positive. A high precision score will correlate to a model which can accurately identify positive predictions with low probability of false positives. To go over recall, it is the proportion of true positives identified by our model out of all instances that are truly positive. Thus, a high recall suggests that our model can identify a majority of the positive instances in the set. With this, our results showed that accuracy, precision, and recall were all relatively high which indicates that the model performs well.

Next, if we look over the false negative rates and false positive rates for both Age and BMI, we can see that overall, the results were relatively low. The false negative rate is the proportion of actual positive instances that are incorrectly classified as negative by the model and the false positive rate is the proportion of actual negative instances that are incorrectly classified by the model as positive. Since our results showed a low value for both, it indicates that the model is effective in identifying positive instances as well as identifying negative instances.

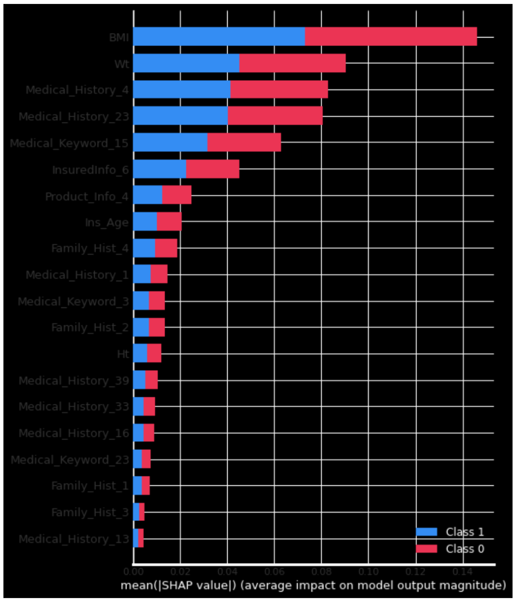
Lastly, if we go over the fairness metrics for Ages and BMI, the false negative rate difference is 1.0 for both features which tells us that the model may miss positive instances for a given demographic group compared to another demographic group which would lead to unfairness and potential bias in our model. The false positive rate is 1.0 for BMI and 0.25 for Ages, which suggests that for BMI, our model is more likely to classify negative instances as positive for one demographic group compared to another which suggests potential bias. However, for Ages, this value is relatively low which suggests the model is relatively fair. In addition for both Ages and BMI, our demographic parity ratio and equalized odds ratio produced values of 0.0 which would mean that our model achieves perfect fairness and sounds too good to be true and thus needs to be further analyzed to be confident in our model’s fairness. Lastly, the selection rate difference is 1.0 for BMI and 0.64 for Ages this means that for BMI, our model is more likely to classify instances as positive for one demographic group compared to another which may introduce bias, however, for Ages, our model does a sufficient job of controlling this bias.

The next step in our analytics was to utilize Locally Interpretable Model-Agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP) to predict from our models which features are most influential and have the heaviest weight in the prediction of an applicant's risk. We utilized different methods for different models that were used. Our results are presented below:

ADS 1 - Random Forest

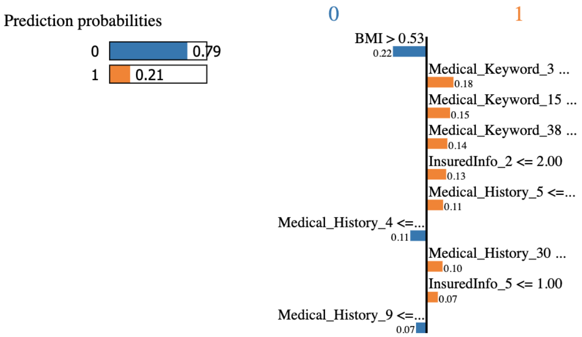


Feature	Value
BMI	0.53
Medical_Keyword_15	0.00
Medical_Keyword_3	0.00
Medical_History_4	1.00
Wt	0.23
InsuredInfo_6	2.00
Medical_Keyword_13	0.00
Medical_Keyword_2	0.00
Medical_Keyword_27	0.00
Medical_Keyword_48	0.00

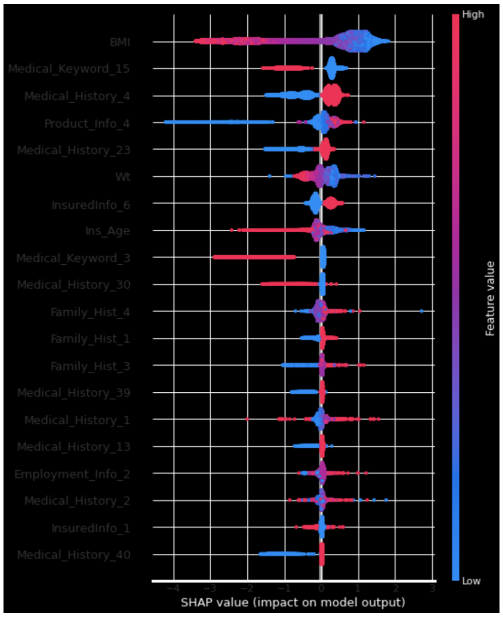


For the Random Forest ADS, we utilized both LIME (left) and SHAP (right) for our analysis. Our results showed that BMI has the highest importance, followed by Medical_Keyword_15, Medical_Keyword_3, Medical_Keyword_4, and Weight.

ADS 2 - Gradient Boosting

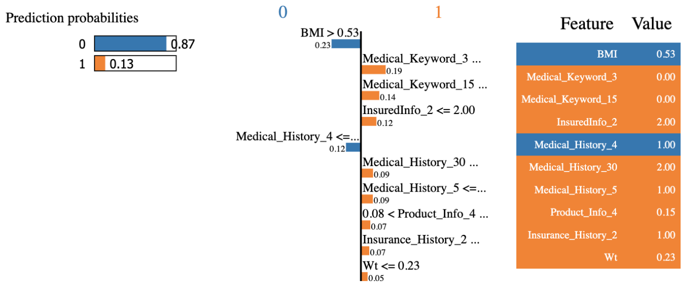


Feature	Value
BMI	0.53
Medical_Keyword_3	0.00
Medical_Keyword_15	0.00
Medical_Keyword_38	0.00
InsuredInfo_2	2.00
Medical_History_5	1.00
Medical_History_4	1.00
Medical_History_30	2.00
InsuredInfo_5	1.00
Medical_History_9	1.00

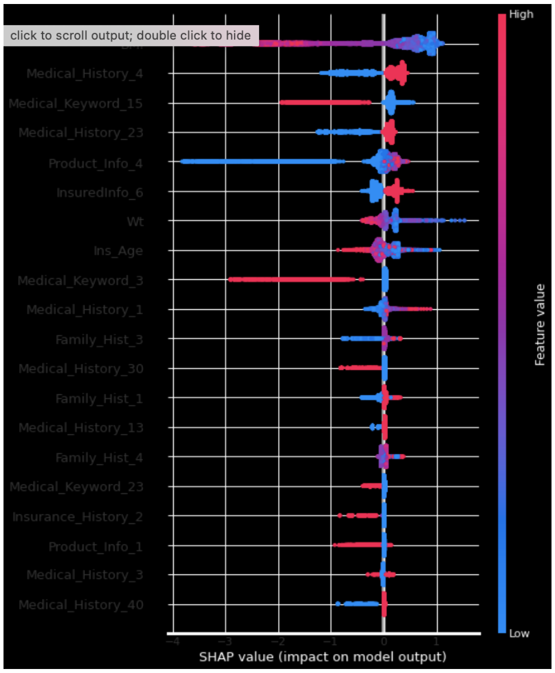


For the Gradient Boosting ADS, we once again utilized both LIME (left) and SHAP (right) for our analysis. Our results showed that BMI had the highest importance again, but this time it was followed up with Medical_Keyword_3, Medical_Keyword_15, Medical_Keyword_38, and lastly, InsuredInfo_2.

ADS 3 - XGBoost



Feature	Value
BMI	0.53
Medical_Keyword_3	0.00
Medical_Keyword_15	0.00
InsuredInfo_2	2.00
Medical_History_4	1.00
Medical_History_30	2.00
Medical_History_5	1.00
Product_Info_4	0.15
Insurance_History_2	1.00
Wt	0.23



Lastly, for the XGBoost ADS, we used both LIME (left) and SHAP (right) for our analysis. This time, we BMI showed up for a third time as the highest important feature and it was followed by Medical_Keyword_3, Medical_Keyword_15, InsuredInfo_2, and Medical_History_4. Overall, we can see the BMI definitely has the highest feature importance in terms of determining risk in an applicant's insurance application. In addition, Medical_Keyword_15, Medical_Keyword_3, and InsuredInfo_2 showed up more than twice in our analytics and thus could be potentially important in determining risk as well.

*The dependence plots for the top 5 features for all three of the ADS analytics are present in the Jupyter Notebook.

Summary

Appropriateness of data for ADS

The presence of null values in the dataset can pose challenges during the modeling process. It was found that there is a high amount of null values in Product_Info columns, which can affect the accuracy and reliability of the predictions if not appropriately handled. In addition, because the dataset contains a large number (128) of features, it could possibly lead to overfitting and reduced performance of the model, hence the user should employ dimensionality reduction such as PCA first, in order to mitigate these issues and improve the ADS' efficiency. Moreover, because demographic information is not present in the data, this might lead to potential biases, such as underrepresentation or overrepresentation of certain demographics, which could result in biased predictions and unfair outcomes.

At the same time, the dataset contains a wide range of variables that are directly related to life insurance applicants and their risk assessment. These variables include product attributes, employment history, insurance history, family history, medical history, age, BMI, weight, and height, and we believe that the availability of such comprehensive information aligns with the objective of developing a predictive model for risk classification in life insurance applicants. Moreover, our target variable “Response” represents an ordinal measure of risk with 8 levels. Having a well-defined target variable enables the development of a supervised learning model, where the goal is to predict the risk category based on the given features. This aligns with the objective of the ADS to classify risk accurately.

Most importantly, the dataset is designed to maintain privacy, as all patient information is encrypted, and the medical histories and keywords are all masked in numbers. This is important in the context of life insurance where customer data privacy is crucial, and by using anonymized and data synthesizing techniques, we conclude that the data is appropriate for the ADS to provide accurate risk assessments while respecting the user's privacy.

Robustness, Accuracy, and Fairness

All in all, we believe that the implementation is robust, accurate, and fair to a certain degree. From our analytics, we have concluded that the model performs relatively well and can produce reliable results even in the presence of noise. The implementation achieved relatively high scores of accuracy in all circumstances and the majority of fairness metrics returned satisfactory results as well. Despite this, we believe it is accurate to a certain degree as more analytics and work can be done in order to ensure an even better model with a more secure evaluation in fairness.

Stakeholder relations with measures

We believe that there are 3 main groups that may find these measures appropriate, including Prudential Company itself, insurance underwriters, and insurance applications. Fairness is important for the company to ensure that the risk assessment process is unbiased and equitable for all applicants, which would help in avoiding any discriminatory practices and ensure that applicants are evaluated based on their actual risk profiles.

Insurance underwriters or actuaries would find fairness and accuracy important as it helps them ensure that all applicants are treated equally and are not subjected to any bias. Accuracy would also be important to underwrites as it would influence their ability to evaluate and price insurance products and policies accurately, as well as assessing likelihood of claims.

As for insurance applicants, they have a significant stake in the fairness and accuracy of the ADS model. Fairness is important for applicants as it ensures that their risk assessment is based on relevant and non-discriminatory factors. They expect a fair evaluation that takes into account their individual characteristics and risk profiles, rather than being unfairly influenced by sensitive attributes or demographic factors. Moreover, accurate risk predictions enable applicants to receive insurance policies that align with their actual risk levels, providing them with appropriate coverage and fair pricing.

Deployment of ADS in public sector

In terms of deployment, we feel like it would be safe to deploy in a private setting such as within the firm or for educational purposes. However, with the idea that more fairness evaluation and changes can be executed in order to make this model even better and a possibility to deploy into the public sector and/or industry. Despite this, we believe that the implementation has a good foundation and could be used in the public sector but there could be backlash and we would recommend that more changes could be made privately to ensure a smooth deployment.

Improvements on data collection, processing, and analysis methodology

In terms of data collection, although the dataset already includes a wide variety of patient information for insurance risk assessment, we believe it would be beneficial to include family/personal income and also the severity of medical histories. Including information about the applicant's family income can provide insights into their financial stability and ability to meet insurance premium obligations. It can be an important factor in determining the risk level, as individuals with higher incomes may have a better ability to pay premiums and may be considered lower risk. Conversely, applicants with lower incomes may be at a higher risk due to potential financial challenges. In terms of the severity of medical histories, assessing that can offer a more accurate evaluation of their health condition and associated risk. This information can help differentiate between applicants with similar medical histories but varying levels of illness severity. This can be obtained through medical indicators, and can be modeled as discrete variables to further analysis.

As aforementioned, because the dataset contains a large number of features, applying dimensionality reduction techniques like Principal Component Analysis. Analysis would be beneficial. PCA captures the most important information while reducing the dimensions of the dataset, which improves efficiency of machine learning models.

During the data manipulation process, the creator handled class imbalance by modifying the target variable, where 0-7 are mapped to 0 and 8 is mapped to 1, the specific reasoning behind this modification is not clear, however even after manipulation there was still imbalance (as seen in the graphic in the above parts). We would suggest modifying the categories to 0-6 as 0 and 7-8 as 1 to reduce class imbalance to prevent the model from being biased towards the majority class.

Lastly, using additional validation techniques such as cross-validation or bootstrapping can help assess the robustness and generalizability of the ADS. These techniques provide a more holistic evaluation of the model's performance by re-testing it on multiple subsets of the data. Employing a robust feature selection technique like LIME and SHAP can also help to identify the most important features for training the model. This ensures that the model focuses on the most relevant and informative variables, leading to improved accuracy and interpretability.