FEDERAL AVIATION ADMINISTRATION LICENSE RENEWAL FORECASTING

DSA 5900

HAIXIAO LU
SPONSOR: OU DISC
ADVISOR: PROFESSOR BEATTIE, DR.NICHOLSON, DR.REINERT

Table of Contents

Federal Aviation Administration License Renewable Forecasting

Introduction

The aim of the project is to provide useful analysis about pilots' licenses renewal based on their medica records from the past one to five years.

The Federal Aviation Administration (FAA) is the largest transportation agency of the U.S. government and regulates all aspects of civil aviation in the country as well as over surrounding international waters. Its powers include air traffic management, certification of personnel and aircraft, setting standards for airports, and protection of U.S. assets during the launch or re-entry of commercial space vehicles.

According to FAA's regulations, to be allowed to independently operate an aircraft, pilots must undergo an aeromedical examination. The purpose of aeromedical examinations is to identify and exclude those who have an unacceptably increased risk of incapacitation during the relevant period of certification after the examination. Episodes of intercurrent illness that may lead to incapacitation are deemed to be self-regulatory. According to International Civil Aviation Organization's standard (ICAO), a pilot requires that he/she will not exercise the privileges of their license if they are aware of any medical condition that might be a flight safety hazard.

A central challenge facing the Federal Aviation Administration (FAA) is how to determine if an airman will be flight ready, such as when an airman applicant presents to the physician, an assessment is performed to determine if the airman, in her/his current state has sufficient capacity to adequately perform safety-critical tasks in the aviation environment; what is the likelihood of a significant state change during the current certification period?

According to Dr. Tvaryanas, the FAA current approaches are precision-based, data-driven to civil aeromedical standards. It uses Aerospace Medicine 1.0 model and Aerospace Medicine 2.0 model.

The Aerospace 1.0 is still based on the model first established by Dr.Louis H.Bauer in 1926. This model is a disease-based, rule focused process. The physician plays an import role to examine an airman applicant to arrive at a list of medical conditions, apply a set of medical certification standards to come to a list of aeromedical disqualifying conditions, and they use a mixture of historical precedent, professional judgment, policy, and guidance to formulate a qualification decision and time limited disposition.

Aerospace Medicine 2.0 uses big data and early AI, as exists with current machine learning models, are being leveraged to improve future forecasting for competitive advantage. In Aerospace medicine 2.0, aeromedical certification decision making is a stage change prediction problem. When an airman applicant presents to the physician, an assessment is performed to determine if the airman, in her/his current state, has sufficient capacity to adequately perform safety-critical tasks in the aviation environment.

The analysis will involve, connecting multiple health data tables from IBM's MarketScan Database to develop forecasts of survival until aeromedical – significant health state changes, the likelihood of individuals transitioning from current state to a different state.

Objective

The bigger picture of the project goal is to analyze those airman applicants' health history from the past one, three and five years, identifying the likelihood of them transitioning from their current health state to an unhealthy state in the next 3 years.

However, with a limited time after getting the data in hand, this project will start small this semester and scale up in the coming semester. This project mainly focuses to demonstrate an approach based on one component of the unhealthy state instead of whole unhealthy state. Particularly, the circulatory system diagnosis category will be analyzed.   We will analyze how likely an individual have diagnosed in the circulatory system category on their current medical history transitioning to a non-circulatory system category.

Data

The datasets come from IBM Truven Health MarketScan database. The Truven Health MarketScan Research Databases capture person-specific clinical utilization, expenditures, and enrollment across inpatient, outpatient, prescription drug, and carve-out services. The data come from a selection of large employers, health plans, and government and public organizations. The Truven Health MarketScan Research Databases are composed of the following individual databases:

- Commercial Claims and Encounters Database
- Medicare supplemental
- Health and Productivity Management Database
- Benefit Plan Design Database
- Medicaid Database
- MarketScan Lab

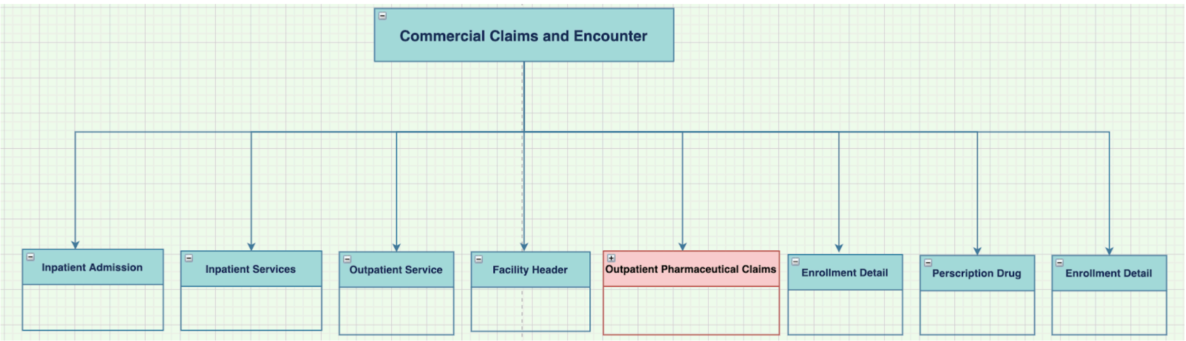


Fig 1. Commercial Claims and Encounters Database

Since we are narrowing down the project scope, we will only use Commercial Claims and Encounters Database. It's composed of the following individual tables:

- Inpatient Admissions Table
- Facility Header Table
- Inpatient Service Table
- Outpatient Services Table
- Outpatient Pharmaceutical Claims Table
- Enrollment Table
- Prescription Drug Table
- Red book Table

The original plan is connecting Inpatient Admission, Inpatient and Outpatient Services, Outpatient Pharmaceutical Claims, Prescription Drug and Red Book tables. Then use this combine the dataset to identify individuals have diagnosed or have claims in these three years. But the Outpatient Pharmaceuticla table (red color in the above figure) is not available currently. There are no meaningful results after combine all the tables because it cannot track individuals across all three years, so this project only uses the most relevant tables which shows in the following to explore the analysis:

- Inpatient Admission
- Prescription Drug
- Red Book

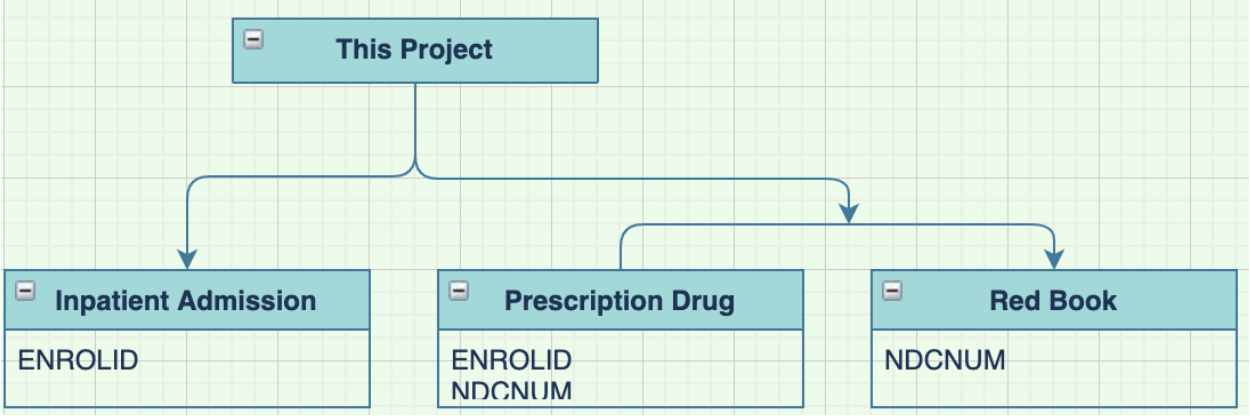


Fig 2. Data tables used in this project

After combining all the tables together, the new dataset has 1574047 instances and 38 features.

Data Preparation

Data preparation has two parts. The first part mainly focusses to combine different tables into one table and impute all the missing values. This part really is trying to get a sense of the data.

- The first part of the data preparation processes shows on the figure below:

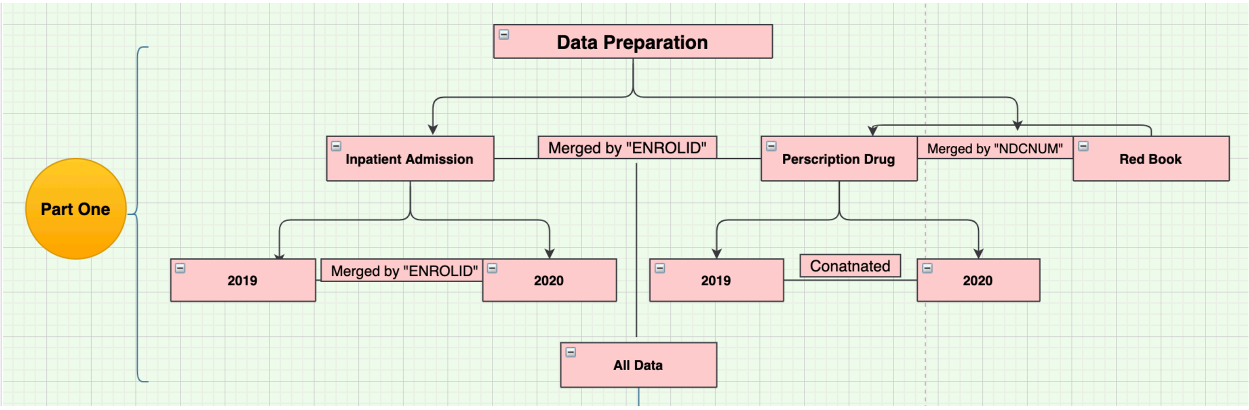


Fig 3. Data preparation process part one

The Inpatient Admission Table has three separate csv files from 2018 – 2020, but it is only merged to one file based on "ENROLID" from 2019 to 2020. The Prescription Drug table also has three different separate csv files in three different years. It is concatenated to one file. The Prescription Drug table and Red Book are merged based on "NDCNUM" first. Then, Inpatient Admission table and Prescription Drug table are merged.

The 38 features are selected from each table that can contribute to the machine learning model the most. It includes Time Variables, Patient Variables, Clinical Variables, Demographic Variables and Drug Variables. Different features are selected from each variable.

The features are:

- Time Variable
  - YEAR (Date Year Incurred)
- Patient Variable
  - ENROLID (Enrollee ID)
- Clinical Variable
  - DX1 – DX15 (Diagnosis Code)
- Demographic Variables
  - AGE (Age of patient)
  - SEX (Gender of patient)
- Drug Variables
  - NDCNUM (National Drug Code)
- Red Book
  - PRODNME (Drug Name)

Some of these features' type are Float and Object. To make it run faster when run the program, all Float type converted to Integer

All the missing values occurred in the Diagnosis Code features which from Diagnosis Code 2 to Diagnosis Code 15. All these Diagnosis are standard International Classification Diseases Code (ICD-10-CM). It structures in three different ways:

- Character 1 – 3
    - Indicate the category of the diagnosis
- Character 4 – 6
    - Indicated anatomic site, severity, or other clinical detail
- Character 7
    - Indicated extension of the code

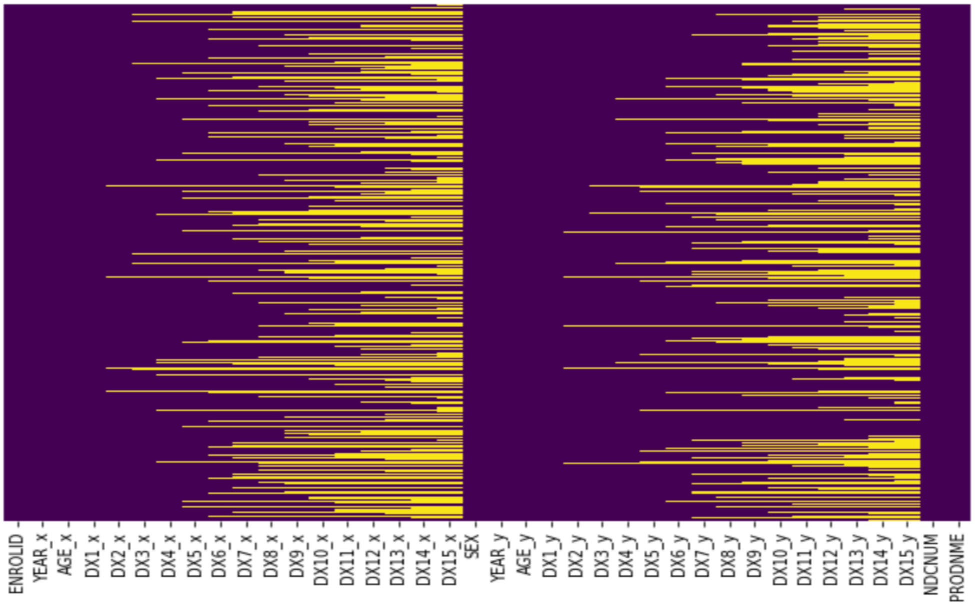


Fig 4. Missing Values

From the above figure, the yellow color represents all the missing values. As it appears, all the missing values from DX2_x to DX15_x and DX2_y – DX15_y. To compute all these missing values in a meaningful way is not much an option since the code is combination of letter and numbers. However, the missing value are computed to 999 as string type instead of all 0s to make more convenience of encoding the diagnosis code later.

The second part of the data preparation focusses on Data Cleaning and Feature Engineering.

- The second part of the data preparation steps show below:

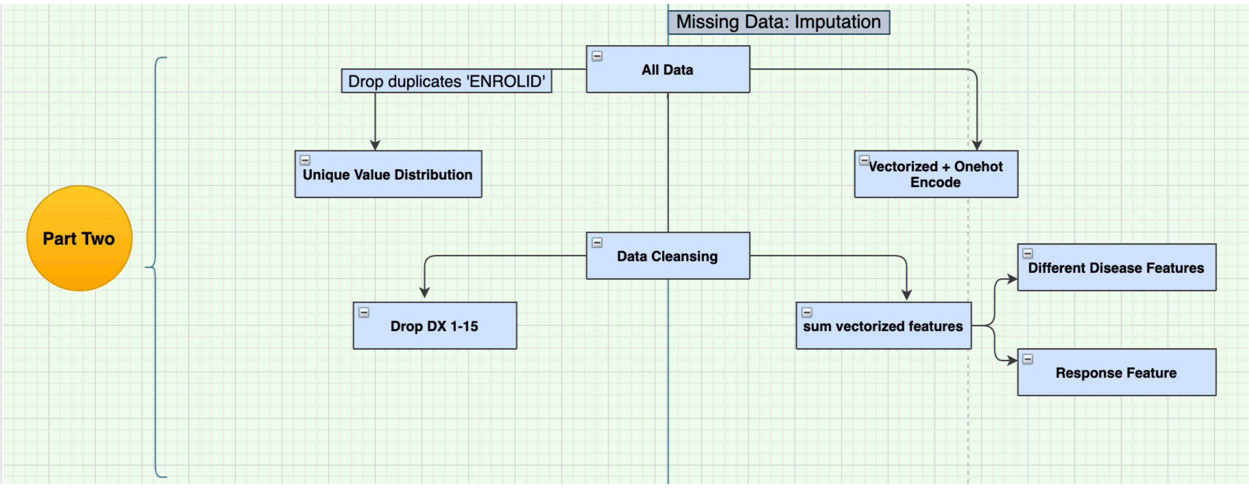
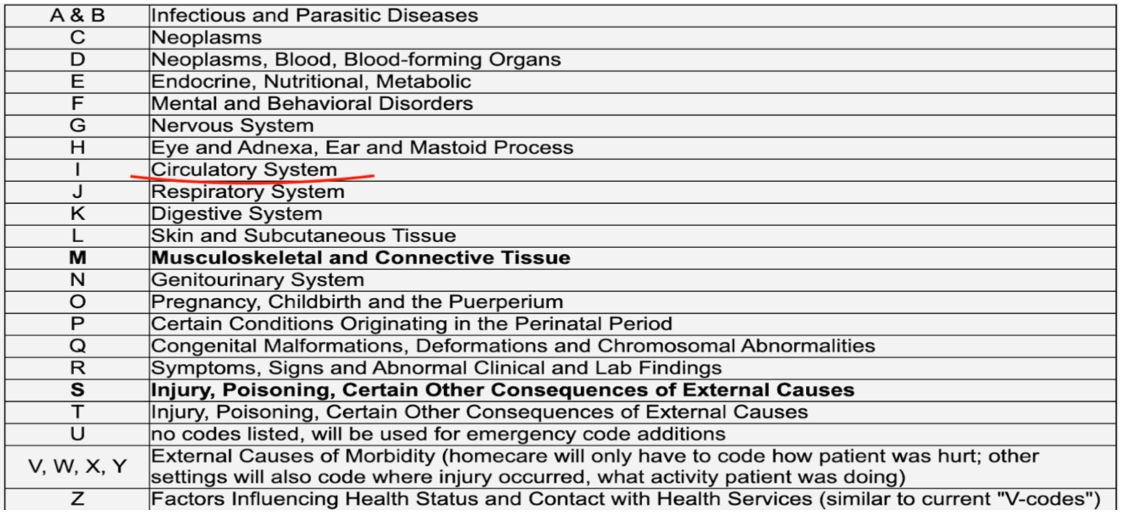


Fig 5. Data preparation process part two

The above figure with light color shows all the processes in the second part of the data preparation. For Data Cleaning, it is a little bit trick to figure out how to make all the values are unique. ENROLID is a unique identifier to track individuals across different years, but duplicated ENROLID might represent same individuals have visited doctor multiples time or diagnosis many times in this combined dataset in a given year. The NDCNUM is unique National Drug Code for trace different drugs, but a patient can have multiple prescription drugs with different diagnosis.
To reduce the complexity of the project, the duplicates are dropped based on "NDCNUM" first. After Feature Engineering, the second part of the duplicates are dropped based on "ENROLID".

For Feature Engineering, encoding all the Diagnosis Code into vectors are mainly focused. Since all the Diagnosis Code are standard ICD-10-CM code. The first letter of the code represents different diagnosis category.

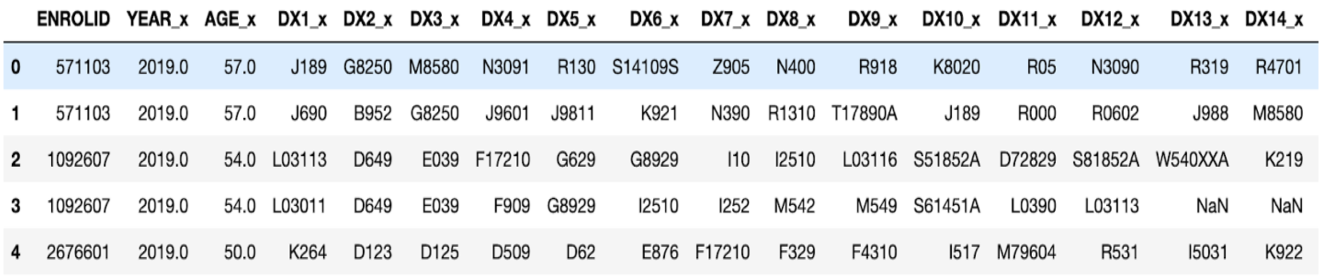


A & B	Infectious and Parasitic Diseases
C	Neoplasms
D	Neoplasms, Blood, Blood-forming Organs
E	Endocrine, Nutritional, Metabolic
F	Mental and Behavioral Disorders
G	Nervous System
H	Eye and Adnexa, Ear and Mastoid Process
I	Circulatory System
J	Respiratory System
K	Digestive System
L	Skin and Subcutaneous Tissue
M	**Musculoskeletal and Connective Tissue**
N	Genitourinary System
O	Pregnancy, Childbirth and the Puerperium
P	Certain Conditions Originating in the Perinatal Period
Q	Congenital Malformations, Deformations and Chromosomal Abnormalities
R	Symptoms, Signs and Abnormal Clinical and Lab Findings
S	**Injury, Poisoning, Certain Other Consequences of External Causes**
T	Injury, Poisoning, Certain Other Consequences of External Causes
U	no codes listed, will be used for emergency code additions
V, W, X, Y	External Causes of Morbidity (homecare will only have to code how patient was hurt; other settings will also code where injury occurred, what activity patient was doing)
Z	Factors Influencing Health Status and Contact with Health Services (similar to current "V-codes")

Fig 6. First letter corresponding its Disease Categories

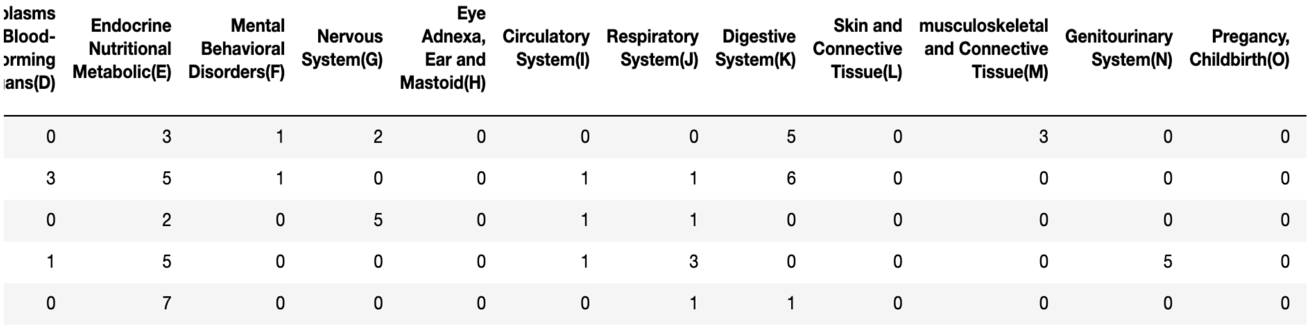
To achieve the goal of the project, codes start with “I” are the ones we are cared about. First, we encoded the Diagnosis Code into one vector. There are 26 alphanumeric categories that point towards different biological systems. Each of these categories will represent one kind of disease category, so we decoded all these diagnosis codes into 26 vectors. Second, the letter “I” represent Circulatory System (shows on the above figure), so we will make Circulatory System (I) to our response variable. We will predict how many people will diagnosed with and without circulatory system in 2020.

- Before Feature Engineering:



	ENROLID	YEAR_x	AGE_x	DX1_x	DX2_x	DX3_x	DX4_x	DX5_x	DX6_x	DX7_x	DX8_x	DX9_x	DX10_x	DX11_x	DX12_x	DX13_x	DX14_x
0	571103	2019.0	57.0	J189	G8250	M8580	N3091	R130	S14109S	Z905	N400	R918	K8020	R05	N3090	R319	R4701
1	571103	2019.0	57.0	J690	B952	G8250	J9601	J9811	K921	N390	R1310	T17890A	J189	R000	R0602	J988	M8580
2	1092607	2019.0	54.0	L03113	D649	E039	F17210	G629	G8929	I10	I2510	L03116	S51852A	D72829	S81852A	W540XXA	K219
3	1092607	2019.0	54.0	L03011	D649	E039	F909	G8929	I2510	I252	M542	M549	S61451A	L0390	L03113	NaN	NaN
4	2676601	2019.0	50.0	K264	D123	D125	D509	D62	E876	F17210	F329	F4310	I517	M79604	R531	I5031	K922

- After Feature Engineering:



lasms Blood- rming ans(D)	Endocrine Nutritional Metabolic(E)	Mental Behavioral Disorders(F)	Nervous System(G)	Eye Adnexa, Ear and Mastoid(H)	Circulatory System(I)	Respiratory System(J)	Digestive System(K)	Skin and Connective Tissue(L)	musculoskeletal and Connective Tissue(M)	Genitourinary System(N)	Pregancy, Childbirth(O)
0	3	1	2	0	0	0	5	0	3	0	0
3	5	1	0	0	1	1	6	0	0	0	0
0	2	0	5	0	1	1	0	0	0	0	0
1	5	0	0	0	1	3	0	0	0	5	0
0	7	0	0	0	0	1	1	0	0	0	0

Fig 7. Feature Engineering Comparison

Exploratory Data Analysis

- The distribution of diagnosis codes relates to Circulatory System

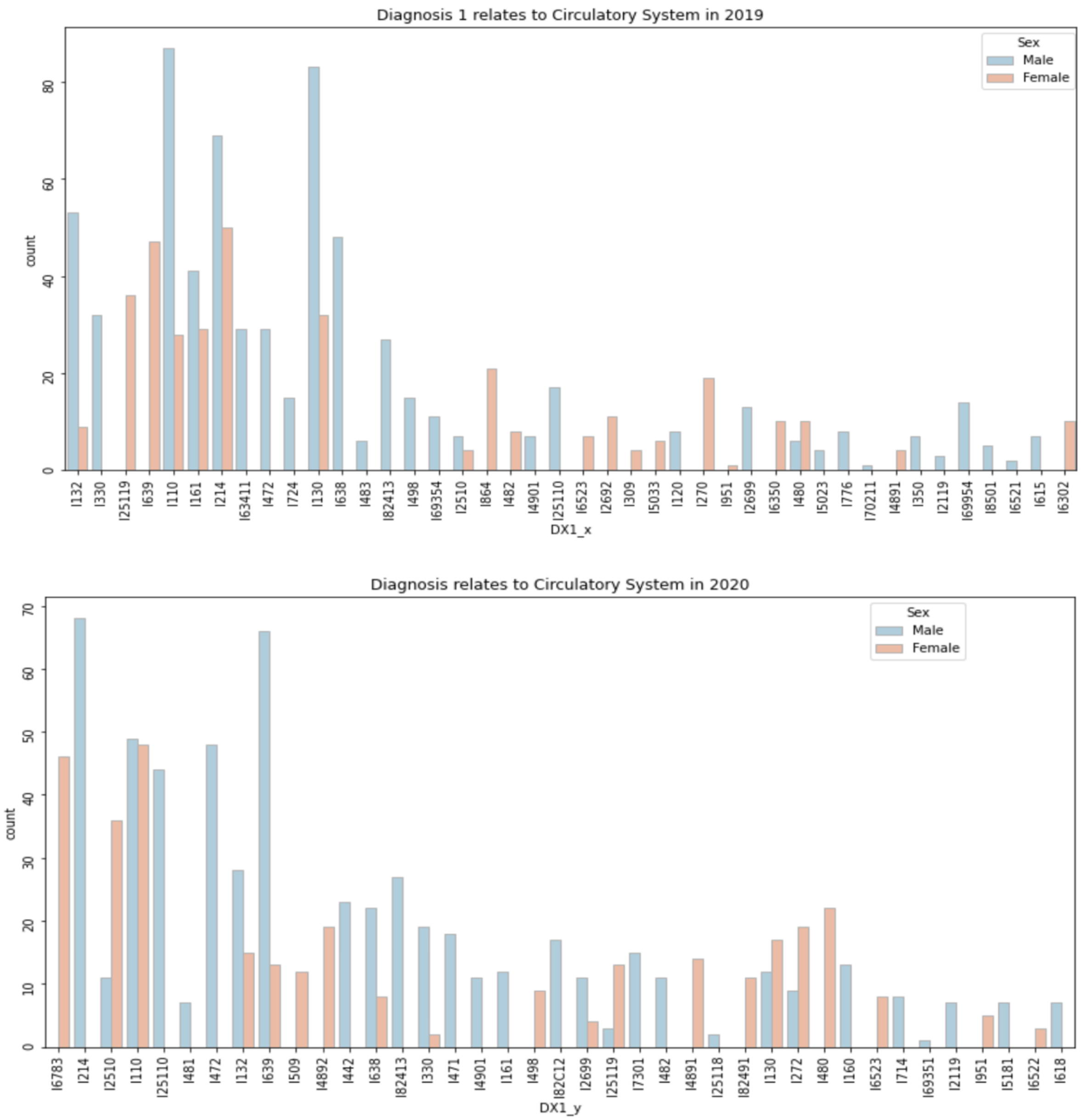


Fig 8. Diagnosis code of Circulatory System in 2019 and 2020

The above figure shows the diagnosis codes relates to circulatory system which start with letter "I". there is more female diagnosis within circulatory system in 2019 than 2020. Notice the highest distribution code is I110 which represents high blood pressure.

- The distribution of diagnosis code of all kinds of disease categories

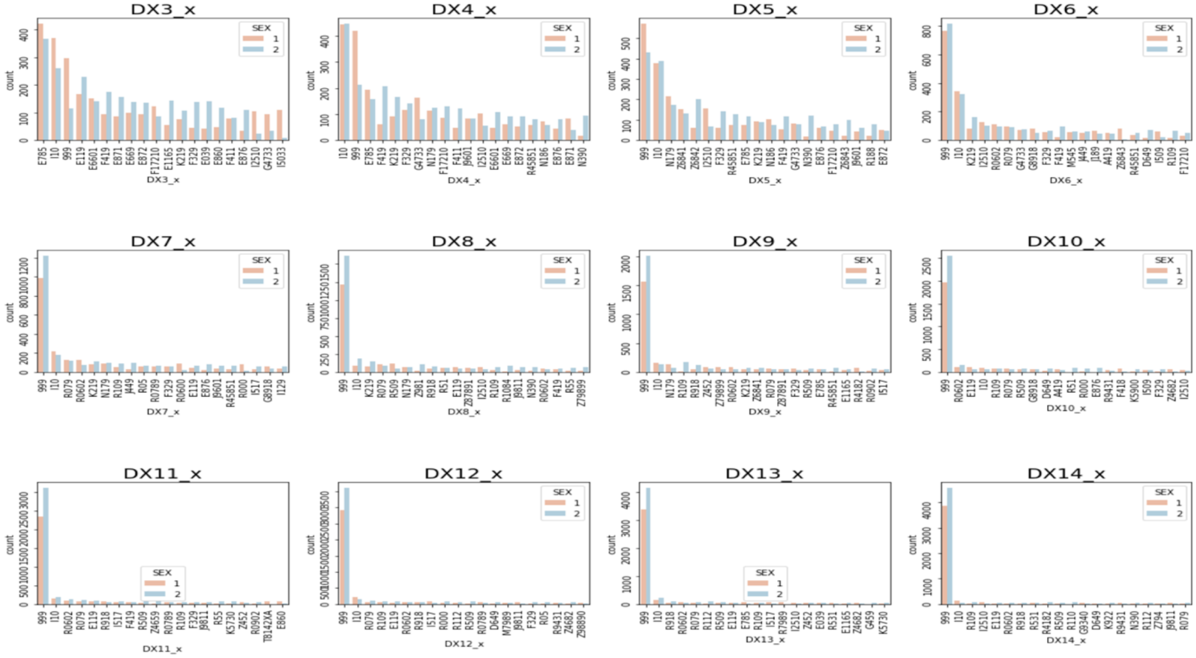


Fig 9. Distribution of all kinds of disease

From above figure, notice from Dx5 to Dx15 the highest distribution code is 999 which is all missing values before computing to 999.

- Top 20 Most Used Drugs and Doctor Visited Patients

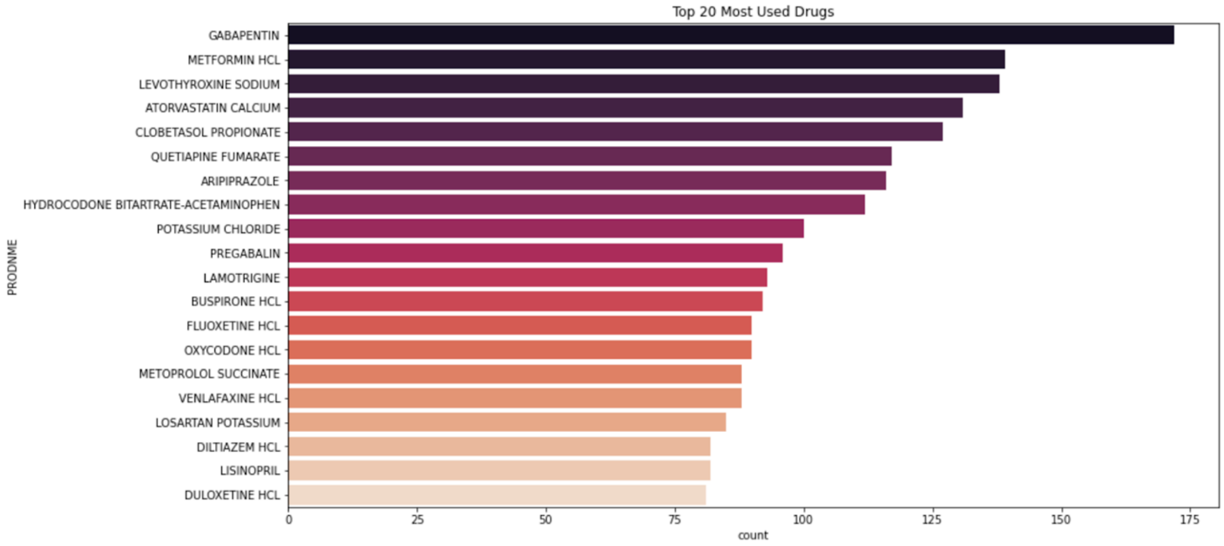


Fig 10. Top 20 most used drugs

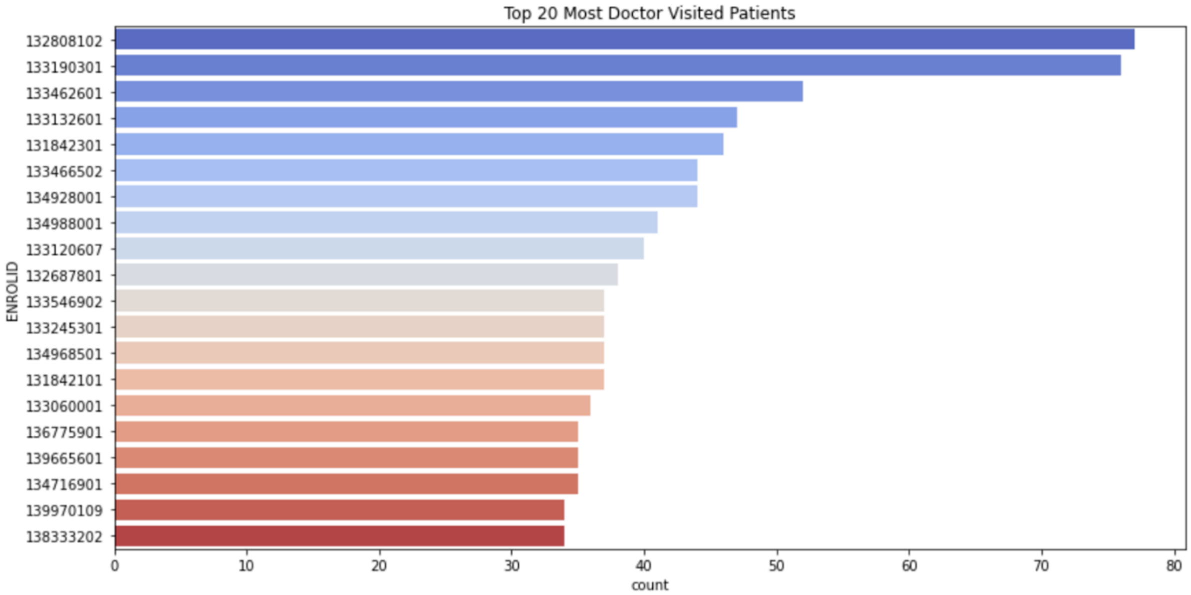


Fig 11. Top 20 most doctor visited patients

- Age Distribution

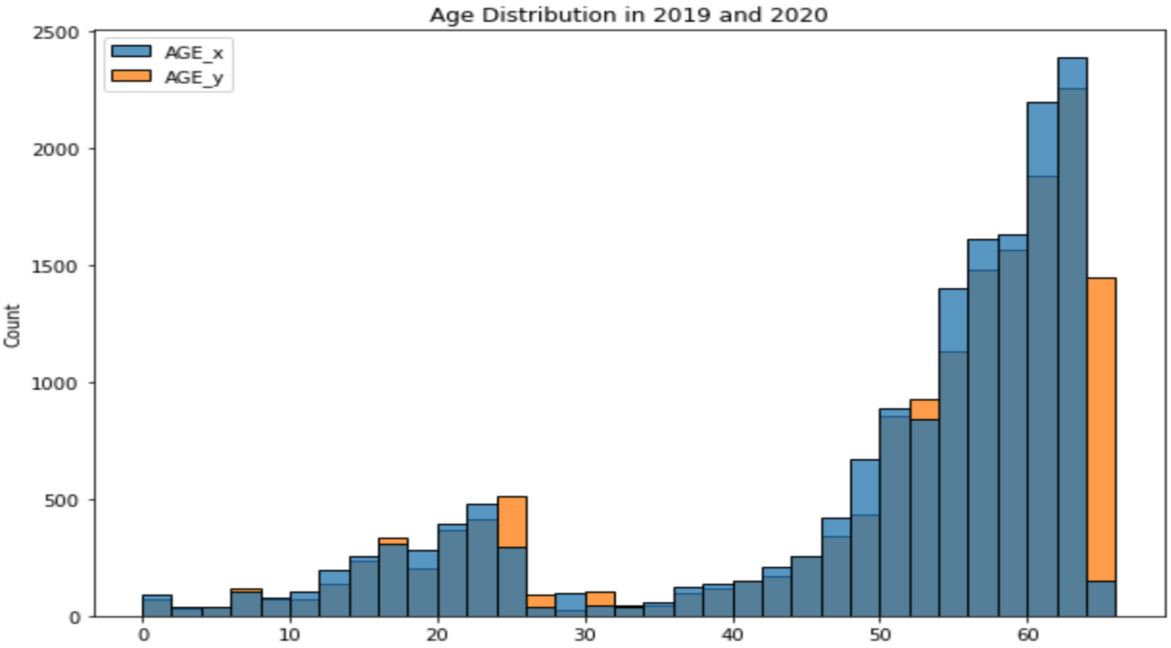


Fig 12. Age distribution

Most the patients around 50 – 65 years old.

- Disease Categories Distribution After Feature Engineering

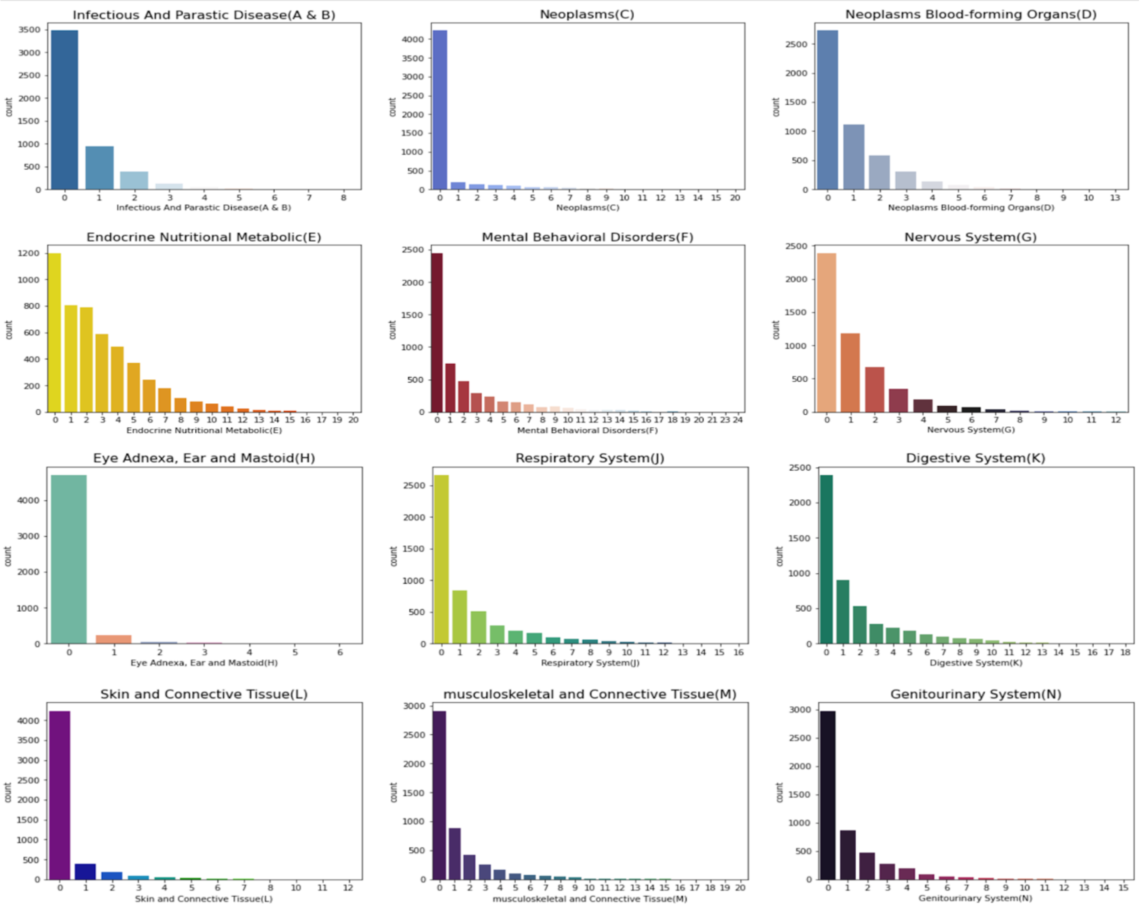


Fig 13. Disease categories

- Target Variable vs. Sex Distribution

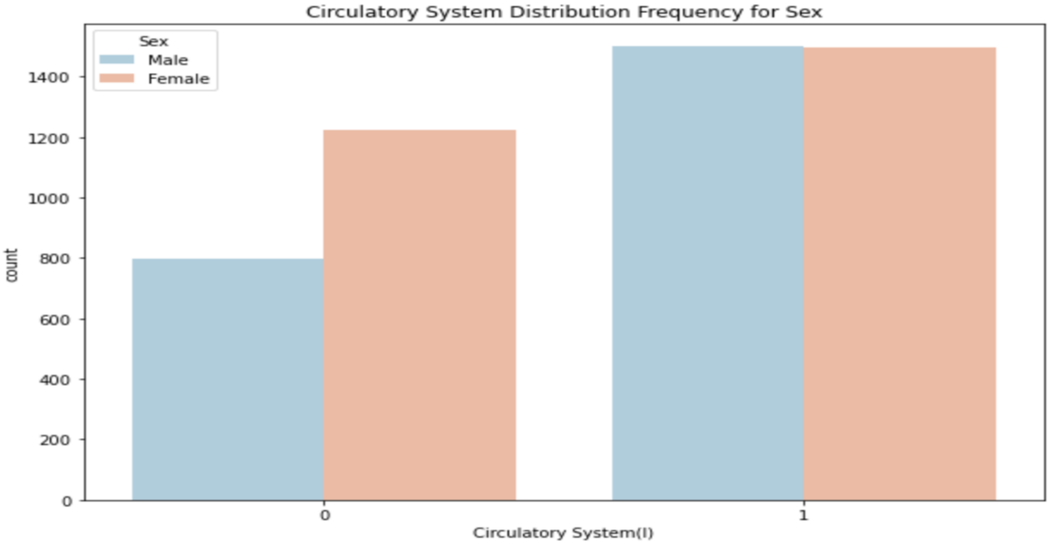


Fig 14. Target Variable

The target variable shows about 60% of the patients diagnosed with circulatory system and 40% of the patients with no circulatory system. Although it's not really balanced, it makes sense though because it's not realistic to say that half the population diagnosed with disease and half not in a dataset.

Methodology

The following figure shows the steps in this section.

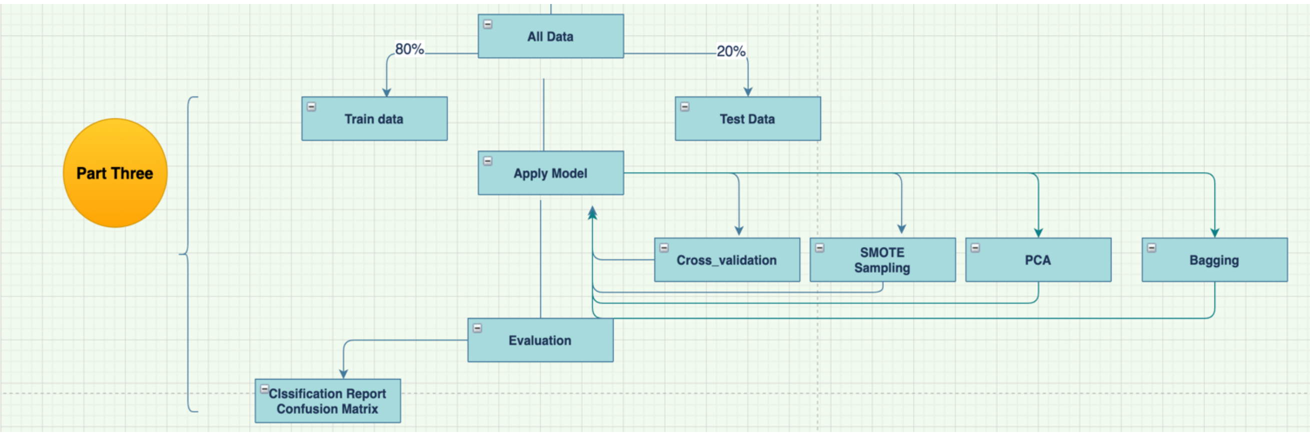


Fig 15. Modeling flow

a. Techniques

The techniques include vectorized and Onehot encoding, principal component analysis, Synthetic Minority Oversampling Technique (SMOTE) and bagging.

During the feature engineering process, how to make diagnosis code in a meaningful way in terms of creating machine learning models. The diagnosis codes are standard International Classification Diseases Code, or ICD-10-CM for short. All these codes structured with letter and numbers. The first one to three characters indicate the category of the diagnosis. The characters of four to six indicate anatomic site, severity, or other clinical detail. In this project, we are mainly cared about the first letter of the codes which tells us the diagnosis category. There are 26 alphanumeric categories that point towards different biological system. Using Onehot encoding technique to change these categorical variables (diagnosis codes) to numerical variables and creating a 26-dimension vector that represents all an individual's diagnoses in a given year. Then we can use these new features to make predictions.

Principal Component Analysis (PCA) is used during the modeling process. To make predictive models performing better than the default settings. PCA is used for dimensionality reduction to a lower-dimensional data while describing as much of the data's variation as possible.

SMOTE technique is also used during the modeling process. Since the targe variable is not balanced. We want to use this technique to balance the training data and test out models' performances. To check whether this technique will improve the performance or not. As you can see from below figure,

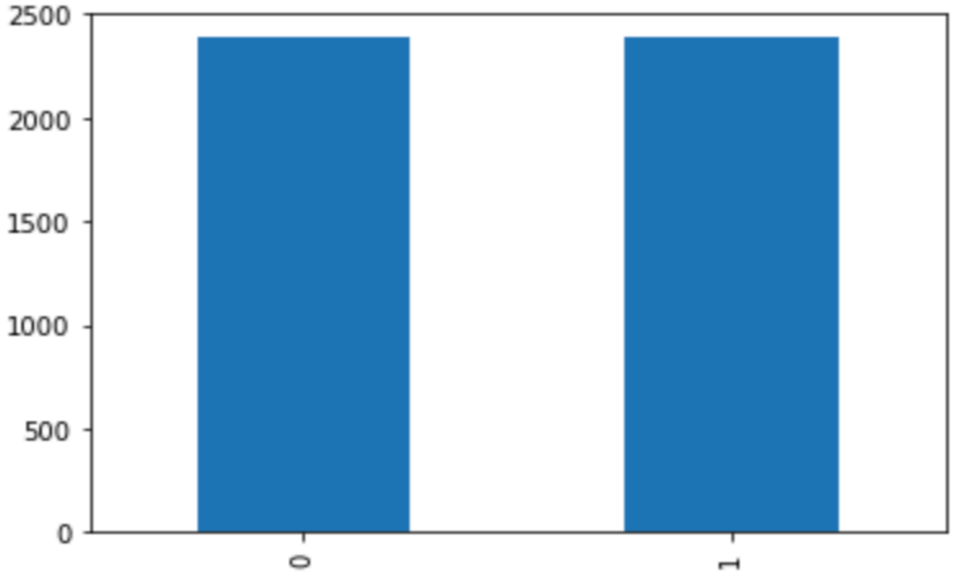


Fig 16. SMOTE technique to balance the target variable

Same as bagging, we want to use this technique to improve the decision tree and random forest models' performance.

b. Procedures

Principle Component Analysis (PCA)

PCA is used to reduce the dimensions. Since there are 24 features in the final dataset which is in 24-dimensional space. It is very hard to visualize and understand the relationship to our models. Hence PCA is used for dimensionality reduction. PCA also can help to identify patterns based on the correlation between features. This algorithm aims to find maximum variance using fewer dimension than the original data.

The following steps are showing where PCA is used,

- Load the cleaned dataset
- Reduce the dimensions using PCA
- Visualize the correlation of the PCA components
- Compare the model's result with and without PCA

Synthetic Minority Oversampling Technique (SMOTE)

To address the imbalanced datasets is to oversample the minority class which is SMOTE. SMOTE involves duplicating examples in the minority class. Although these examples don't add any new information to the model. Instead, new examples can be synthesized from the existing examples.

The following steps are showing where SMOTE technique is taking place,

- Load the cleaned dataset
- Import SMOTE package
- Split training data with SMOTE
- Building models with new training data
- Compare models' results with and without SMOTE

Bagging Classifier

Bagging is the ensemble learning method that is commonly used to reduce variance within a noisy dataset. In bagging, a random sample of data in a training set is selected with replacement. After several data samples are generated, these weak models are then trained independently. To make predictions yield a more accurate estimate, bagging is used for building decision tree and random forest to improve the models' performance.

The following steps are showing the bagging classifier taking place,

- Load cleaned dataset
- From sklearn ensemble import bagging classifier
- Set up the bagging classifier with parameters
- Train the model
- Compare models' result with and without bagging classifier

Results and Analysis

Since this project only focuses demonstrating an approach whether airman applicants diagnosed with circulatory system or not in a given year. The models' evaluation based on following criteria:

- Area Under Curve (AUC)
- F1 Score
- Precision
- Confusion Matrix

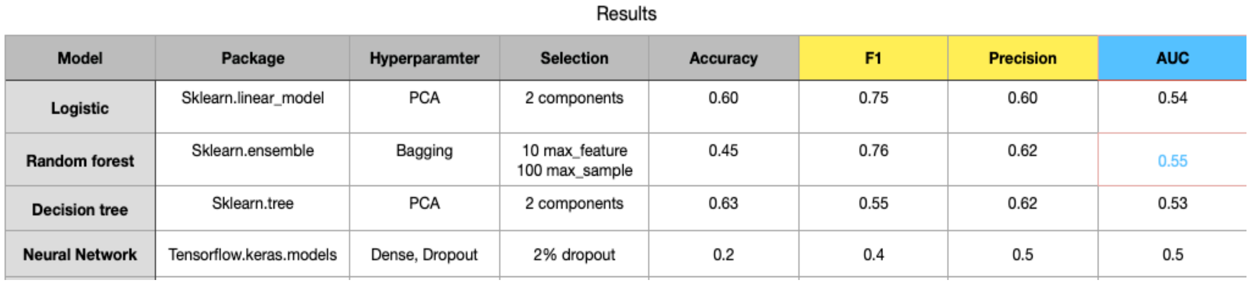
AUC score can tell us whether we can distinguish the positive and negative class or not. If AUC score is greater than 0.5 and less than 1. It can distinguish the positive class values from the negative. Otherwise, if AUC score is less and equal to 0.5, it is not able to distinguish two classes.

Since we are predicting diseases and the data is imbalanced. The accuracy score is not suitable in this case. For the trade off, we are focusing on F1 score and Precision to see how accurate of predicated positive.

Confusion Matrix is used since we are dealing with classification problem. We are mainly focused on reducing False Negative (FN) in this case. We don't want to give False hopes to applicants that may cause a huge impact of the safety to operate aircraft.

Here are some of the results from different models:

- Overall models' performance



Results

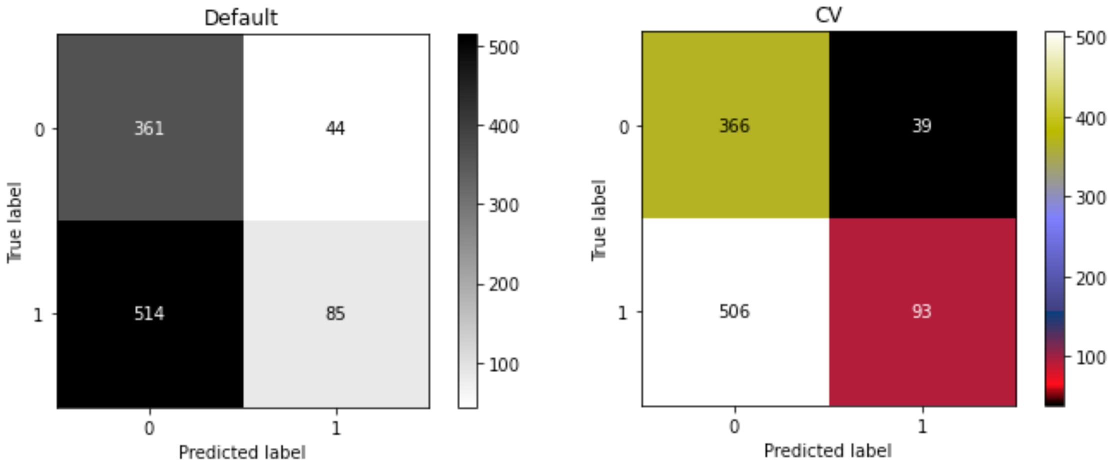
Model	Package	Hyperparamter	Selection	Accuracy	F1	Precision	AUC
Logistic	Sklearn.linear_model	PCA	2 components	0.60	0.75	0.60	0.54
Random forest	Sklearn.ensemble	Bagging	10 max_feature 100 max_sample	0.45	0.76	0.62	0.55
Decision tree	Sklearn.tree	PCA	2 components	0.63	0.55	0.62	0.53
Neural Network	Tensorflow.keras.models	Dense, Dropout	2% dropout	0.2	0.4	0.5	0.5

Fig 17. Models' Results

From the above figure, we can see that Random Forest has the highest AUC score and F1 Score. We will see if it has the lowest False Negative in confusion matrix comparing to other models.

- Confusion Matrix of each model

Random Forest



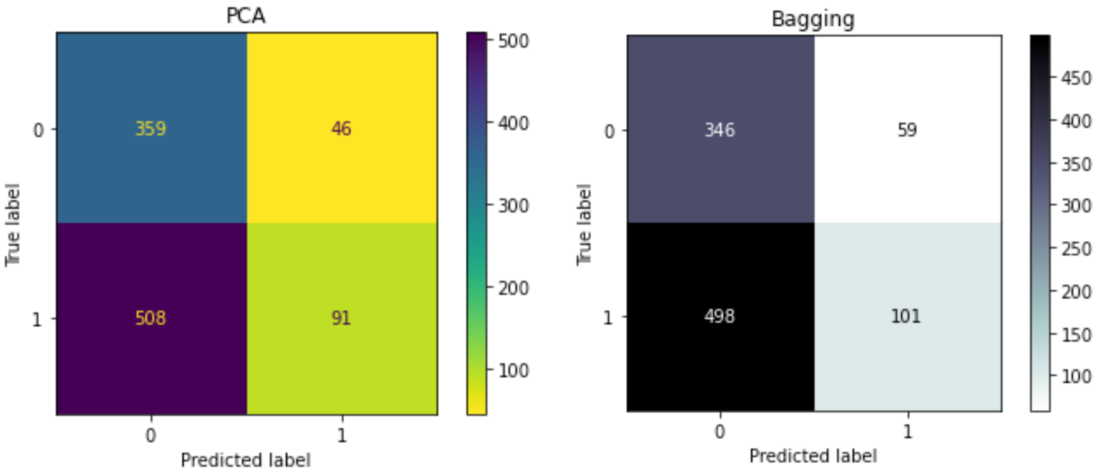


Fig 18. Random Forest's Confusion Matrix

From the above figure, Random Forest with cross-validation has the lowest False Negative.

Decision Tree

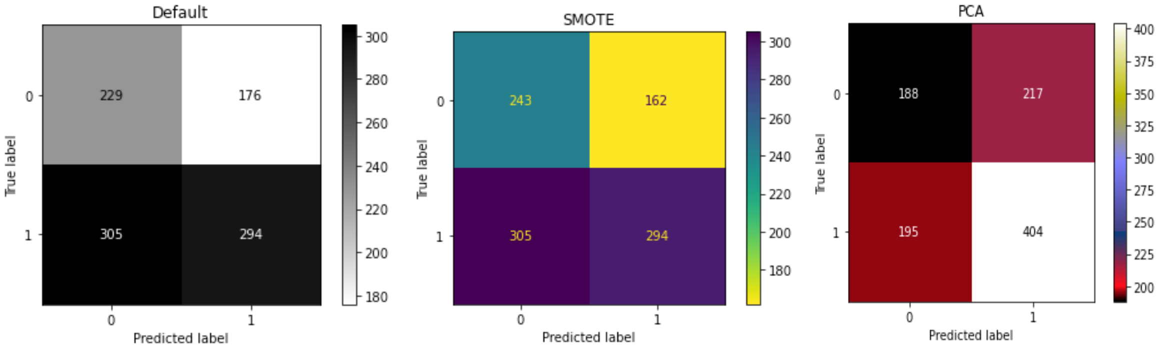
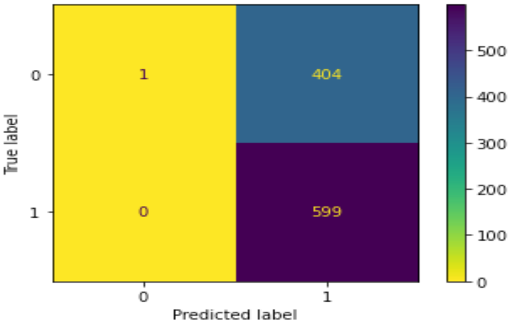


Fig 19. Decision Tree's Confusion Matrix

The lowest False Negative is using SMOTE technique in decision tree model.

Logistic Regression and Neural Network Confusion Matrix



For Logistic Regression and Neural Networks, the confusion matrix doesn't make any sense. Hence, these two models do not give us any insights.

From the above results we can tell this approach works. It clearly distinguishes the binary classes which people diagnosed with circulatory system and without circulatory system. Meaning whether airman applicants health state stable or not during the certification period.

Deliverables

From the results, If based on one safety-critical task in a diagnosis category, such as heart disease. Using machine learning methods to forecasting airman applicants transitioning from unhealthy state to healthy state or healthy to unhealthy can be easily done. This approach matches FAA's interest. Although the model accuracy is not that good, we can dig deeper to find personal medical records information to improve models' accuracy, such as weight, height, blood pressure, etc.

For sure, there are more works to do in the future to make the stakeholders satisfy completely. Here are some thoughts about what to do in the future.

For this project, another approach is to narrow down even further. Instead of focusing whole diagnosis category, we can focus only one safety-critical task disease – heart attack. The diagnosis code for heart attack is I501. We can only focus this code to make predictions. We can look at individuals' procedures and heart attack drugs they have taken. Add more personal clinical information such as weight, chest pain, heart rate to do the analysis. We can just focus whether people will have heart attack or not in a given year.

Another approach to satisfy the overall goal is to work with experts to identify safety-critical categories relates to FAA from 26 disease categories. Instead of one diagnosis category like Circulatory System, it might have several disease categories that are critical to operate aircraft safely. After identifying safety-critical categories, we can apply multi-classes classification to predict the diagnosis categories. Then comparing these categories to see the likelihood from current state to a different state.

The results of the project and future work reference will be shared with the sponsor for further analysis and answer any possible referencing question they might have in the future. The code files and final reports will be shared in Microsoft Teams.

References

MarketScan Database User Guide - https://theclearcenter.org/wp-content/uploads/2020/01/IBM-MarketScan-User-Guide.pdf

SMOTE Technique - https://machinelearningmastery.com/smote-oversampling-for-imbalanced-classification/

Bagging Technique - https://www.ibm.com/cloud/learn/bagging