Milestone 3 (Preliminary Analysis) - Week 6

Milestone 3 should include all the information from Milestone 2, updated if necessary.

Milestone #2

- **What types of model or models do you plan to use and why?**
    - A Logistic Model and Decision Tree Classifier will be used on the dataset to determine which features are most related to or most predictive of heart disease, where the HeartDisease feature is the target. This will be a supervised learning model where “the supervisor is the target variable, a column in the data representing values to predict from other columns in the data. The target variable is chosen to represent the answer to a question the organization would like to answer or a value unknown at the time the model is used that would help in decisions. Sometimes supervised learning is also called predictive modeling. The primary predictive modeling algorithms are classification for categorical target variables or regression for continuous target variables.” (Abbott, 2014, p. 5).
    - In addition to the logistic model with 1 target and 17 features, a second logistic model will be used with the 5 best features only, where the 5 features are derived with highest chi-squared statistics.
- **How do you plan to evaluate your results?**
    - I plan to calculate the accuracy, precision, recall and F1 score of both the logistic regression model with 17 features and best 5 features, as well as visualize a confusion matrix and an ROC curve where "The area under the ROC curve (AUC) results were considered excellent for AUC values between 0.9-1, good for AUC values between 0.8-0.9, fair for AUC values between 0.7-0.8, poor for AUC values between 0.6-0.7 and failed

for AUC values between 0.5-0.6." (El Khouli, 2009). All of these together will be used to evaluate the results of the logistic regression models.

- **What do you hope to learn? Why is this data useful to solve the problem?**
    - This data references individual lifestyle and other disease information, which is useful as they could have an impact on heart disease risk potential. I hope to learn what features in the dataset, if any, are predictive of heart disease.
- **Assess any risks with your proposal**
    - Logistic regression models are not perfect and have their disadvantages, “Logistic Regression is a statistical analysis model that attempts to predict precise probabilistic outcomes based on independent features. On high dimensional datasets, this may lead to the model being over-fit on the training set, which means overstating the accuracy of predictions on the training set and thus the model may not be able to predict accurate results on the test set. This usually happens in the case when the model is trained on little training data with lots of features. So on high dimensional datasets, Regularization techniques should be considered to avoid over-fitting (but this makes the model complex). Very high regularization factors may even lead to the model being under-fit on the training data.” (Grover, 2020). If overfitting is identified as an issue via cross-validation, two types of regularization techniques will be used: Lasso (L1 Regularization) or Ridge (L2 Regularization), where λ is called the regularization parameter and controls the trade-off between fitting the training data well and keeping the parameters small to avoid overfitting.
- **Identify a contingency plan if your original project plan does not work out**
    - If this plan does not work out, the contingency plan will be to use a similar medical dataset on HCC (Hepatocellular Carcinoma dataset) survival. This dataset has more

features (50) than the selected Heart Disease dataset (17), with less samples (165). The target variable would be the “Class” feature (nominal - 1 if patient survives, 0 if patient died), and a logistic regression model would be used to predict patient survival from HCC.

- **Include anything else you believe is important:**
- **Problem Statement**
    - Which dataset feature(s) are most related to or most predictive of heart disease (single characteristics and interactions)?
- **Explain why the problem is important/interesting**
    - This problem is important because many people die of heart disease every year “Heart disease is the leading cause of death for men, women, and people of most racial and ethnic groups in the United States. One person dies every 36 seconds in the United States from cardiovascular disease. About 659,000 people in the United States die from heart disease each year—that’s 1 in every 4 deaths. Heart disease costs the United States about $363 billion each year from 2016 to 2017. This includes the cost of health care services, medicines, and lost productivity due to death.” (CDC, 2022). Understanding what factors contribute to heart disease could save lives and healthcare costs.
- **Who would be interested in solving this problem, i.e., who would you be trying to sell this project to?**
    - Doctors, healthcare professionals, patients, and life & health insurance companies would all be interested in understanding what factors are most predictive of heart disease. Life & health insurance companies would be interested in solving this problem to better understand how risky a particular customer would be to insure. Doctors,

healthcare professionals and patients would be interested to ensure to be able to advise on heart disease risk factors and reduce heart disease events (coronary heart disease (CHD) or myocardial infarction (MI)).

- **Where did you get your data?**
    - This dataset was obtained from an open source data website, Kaggle (https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease?resource=download). The dataset contains 319,795 rows and 18 columns or features (9 booleans, 5 strings and 4 decimals), which are:
        - HeartDisease: Respondents that have ever reported having coronary heart disease (CHD) or myocardial infarction (MI) - either yes or no
        - BMI: Body Mass Index (BMI) - float
        - Smoking: Have you smoked at least 100 cigarettes in your entire life? [Note: 5 packs = 100 cigarettes] - either yes or no
        - AlcoholDrinking: Heavy drinkers (adult men having more than 14 drinks per week and adult women having more than 7 drinks per week - either yes or no
        - Stroke: (Ever told) (you had) a stroke? - either yes or no
        - PhysicalHealth: Now thinking about your physical health, which includes physical illness and injury, for how many days during the past 30 days have you exercised - integer
        - MentalHealth: Thinking about your mental health, for how many days during the past 30 days was your mental health not good? - integer
        - DiffWalking: Do you have serious difficulty walking or climbing stairs? - either yes or no
        - Sex: Are you male or female? - string

- AgeCategory: Fourteen-level age category - string
- Race: Imputed race/ethnicity value- string
- Diabetic: (Ever told) (you had) diabetes? - either yes, no, no-borderline diabetes, or yes pregnancy
- PhysicalActivity: Adults who reported doing physical activity or exercise during the past 30 days other than their regular job - either yes or no
- GenHealth: Would you say that in general your health is... - Excellent, Very Good, Good, Fair, Poor
- SleepTime: On average, how many hours of sleep do you get in a 24-hour period? - integer
- Asthma: (Ever told) (you had) asthma? - either yes or no
- KidneyDisease: Not including kidney stones, bladder infection or incontinence, were you ever told you had kidney disease? - either yes or no
- SkinCancer: (Ever told) (you had) skin cancer? - either yes or no

Milestone #3

- **Will I be able to answer the questions I want to answer with the data I have?**
  - **Problem Statement:** Which dataset feature(s) are most related to or most predictive of heart disease (single characteristics and interactions)?
  - Yes, I will be able to answer my question with the dataset that I have. The dataset consists of 17 features (both numerical and categorical) with “HeartDisease” being the target (Yes/No option transformed into 1/0, respectively). There are no missing data points, and the categorical features will have dummy variables for the modeling.

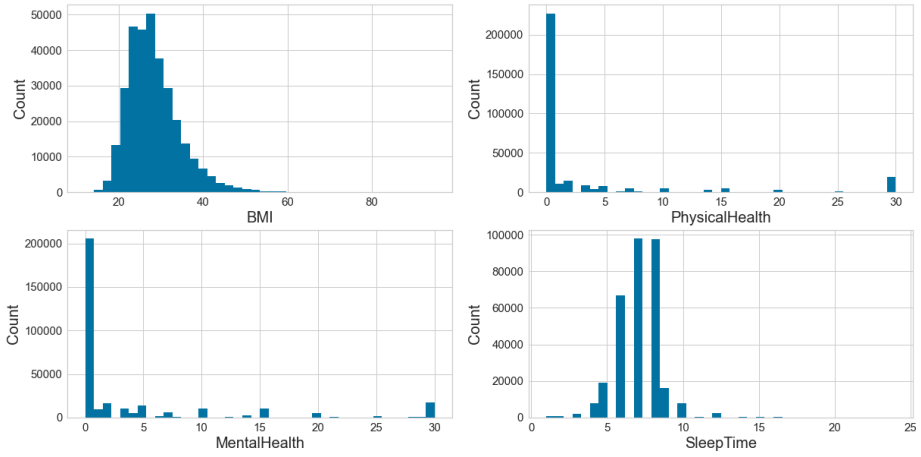
- Observations

  - The dataset has 319795 rows and 18 columns
  - All of the features will be useful in this analysis. Any categorical features will have dummy variables assigned to them for modeling
  - There is a mix of numerical and categorical data
  - The target of the model will be 'HeartDisease': No means no heart disease reported, Yes means heart disease reported

- Observations

  - There is no missing data to deal with (all features have 319795 observations which make them complete)

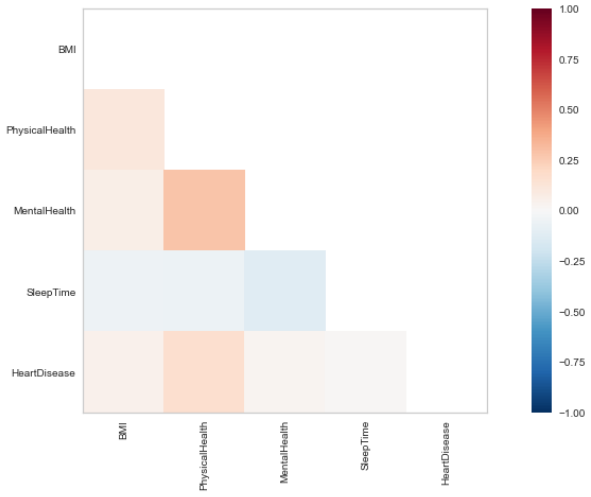
- **What visualizations are especially useful for explaining my data?**
  - Histograms, bar charts, Pearson's correlation matrix (numerical data) and Spearman's correlation matrix (categorial & numerical data) will be used to explain my data.
  - **Numerical Data:** To show the distribution of the numerical data (BMI, PhysicalHealth, MentalHealth, and SleepTime), histograms were used



Observations

- Most people in the dataset have a BMI between 20 and 40
- Most people in the dataset exercise between 0-5 days in the past 30 days
- Most people in the dataset have 0-5 bad mental health days in the past 30 days
- Most people in the dataset get between 5-9 hours of sleep a night

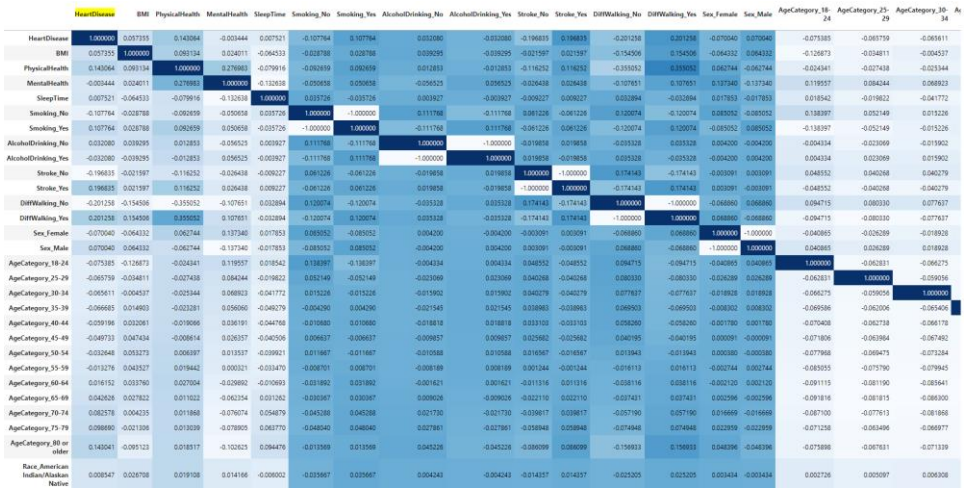
  - A Pearson's Correlation Matrix was then created to understand the correlation between 'HeartDisease" and the numerical features (BMI, PhysicalHealth, MentalHealth, SleepTime)



Observations

- The numerical feature most highly correlated with 'HeartDisease' is 'PhysicalHealth'
- There is not significant collinearity between the features (Rule of thumb: If the correlation > 0.8 then severe multicollinearity may be present)

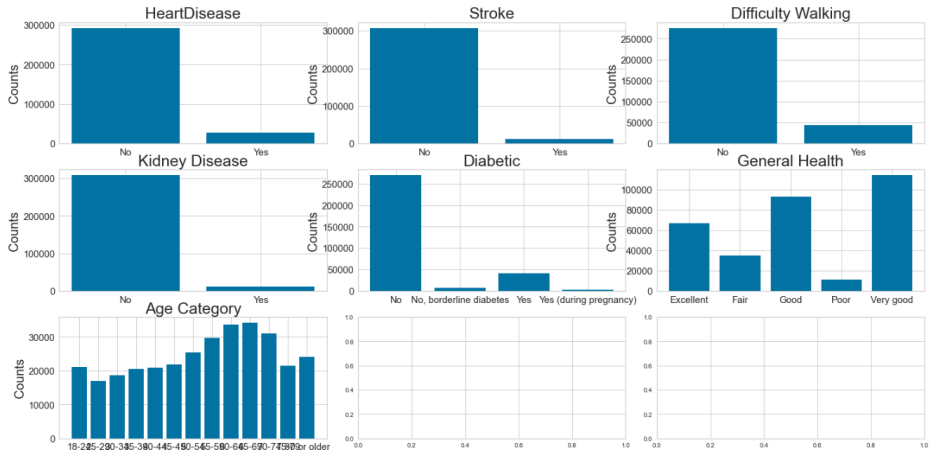
- **Categorial Data:** In this section, we will start out first with the Spearman's correlation matrix to find the features most correlated to the target variable, 'HeartDisease'



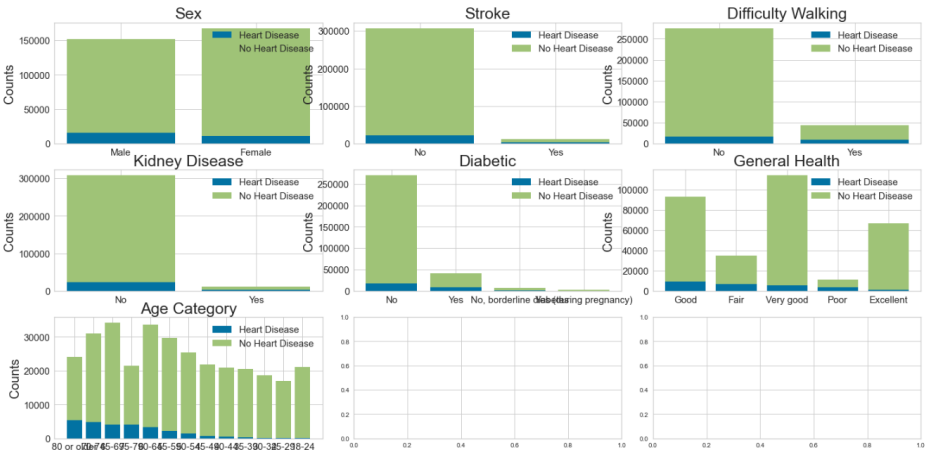
Observations

- The categorical features most highly correlated with 'HeartDisease' are 'Stroke_Yes', 'DiffWalking_Yes', 'AgeCategory_80 or older', 'Diabetic_Yes', 'GenHealth_Fair', 'GenHealth_Poor', 'KidneyDisease_Yes'
- There is not significant collinearity between the features (Rule of thumb: If the correlation > 0.8 then severe multicollinearity may be present)

- From here, histograms were created for only the most correlated features to 'HeartDisease' ('Stroke, 'DiffWalking', 'AgeCategory', 'Diabetic, 'GenHealth' 'KidneyDisease') to show their distributions



- Additionally, stacked bar charts were created to compare heart disease and no heart disease by the most correlated features found in the Spearman's Correlation Matrix



- **Do I need to adjust the data and/or driving questions?**
  - Although the dataset is fairly simple and not overly complex with no missing data, the data does not need to be replaced or the driving questions changed.
  - The only adjustments to the data were the target variable ('HeartDisease') was converted into a binary option (No is 0 and Yes is 1), and the categorical features were

transformed into dummy variables. This was all done to support the Correlation Matrices and the Logistic Regression Modeling.

- **Do I need to adjust my model/evaluation choices?**
    - As the problem statement looks to understand what features are most predictive of heart disease (a binary field), a classification algorithm would need to be used as it predicts a class/category (heart disease detection). Therefore, a logistic regression model will be used on the dataset to determine which features are most related to or most predictive of heart disease, where the HeartDisease feature is the target.
    - The model choice does not need to be adjusted as this problem statement lends itself to a categorical decision which is supported by a logistic regression model.
- **Are my original expectations still reasonable?**
    - Yes, after analyzing and visualizing the data, my original expectations that the Heart Disease dataset will be able to help us understand which features are most predictive of heart disease is still valid.
    - There is no missing data, all categorical data has been transformed into dummy variables, and the data is ready to be prepared for the logistic regression model.

Milestone 4 (Finalizing Your Results) - Week 9

I will determine and describe the data preparation process (e.g., creating new features, hiding features not relevant, removing any null data), and with the cleaned data create several models and evaluate their performance and results for interpretation. With these results, a conclusion and recommendations in reference to the original question and problem will be detailed.

- **Explain your process for prepping the data**
    - Split data into training and test sets with Heart Disease feature (binary Yes/No converted to 1/0) as the target

```
#split the data into a training and test set
#we do this before making any modifications to the data to prevent data snooping
#drop 'HeartDisease' from the features as it is the target
X = df.drop(['HeartDisease'], axis = 1)
#get the target
y = df['HeartDisease']
#split the data into training and test sets
#split the data into training and test sets (80% Training/20% Test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2)

#reset indices in the training and test sets to prevent pandas slicing warnings
X_train = X_train.reset_index(drop = True) #drop = True drops the previous index
X_test = X_test.reset_index(drop = True)
y_train = y_train.reset_index(drop = True)
y_test = y_test.reset_index(drop = True)
```

- Checked that there is no missing data to handle (no missing data)

```
#look for missing data in the training and test sets
print(X_train.isna().sum())
print(X_test.isna().sum())
```

```
BMI                 0
Smoking             0
AlcoholDrinking     0
Stroke              0
PhysicalHealth      0
MentalHealth        0
DiffWalking         0
Sex                 0
AgeCategory         0
Race                0
Diabetic            0
PhysicalActivity    0
GenHealth           0
SleepTime           0
Asthma              0
KidneyDisease       0
SkinCancer          0
dtype: int64
BMI                 0
Smoking             0
AlcoholDrinking     0
Stroke              0
PhysicalHealth      0
MentalHealth        0
DiffWalking         0
Sex                 0
AgeCategory         0
Race                0
Diabetic            0
PhysicalActivity    0
```

- To ensure the data can be processed by the models, dummy variables were created in the dataframe for the categorical data which increased the features from 17 to 37 due to Yes/No options for categorical features

```
X_train = pd.get_dummies(X_train, drop_first = True)
X_test = pd.get_dummies(X_test, drop_first = True)
```

- To find the best 5 features, X2 was used, and additional train & test sets were created

```
# STEP 3: split the data into a training and test set
X_chi = df2[['PhysicalHealth', 'Stroke_Yes', 'DiffWalking_Yes', 'Diabetic_Yes', 'GenHeal
#get the target
y_chi = df2['HeartDisease']

#split the data into training and test sets (80% Training/20% Test)
X_train_chi, X_test_chi, y_train_chi, y_test_chi = train_test_split(X_chi, y_chi, test_s

#reset indices in the training and test sets to prevent pandas slicing warnings
X_train_chi = X_train_chi.reset_index(drop = True) #drop + True drops the previous index
X_test_chi = X_test_chi.reset_index(drop = True)
y_train_chi = y_train_chi.reset_index(drop = True)
y_test_chi = y_test_chi.reset_index(drop = True)
```

- The 5 best features from X2 are as follows: PhysicalHealth, Stroke_Yes, DiffWalking_Yes, Diabetic_Yes, GenHealth_Poor

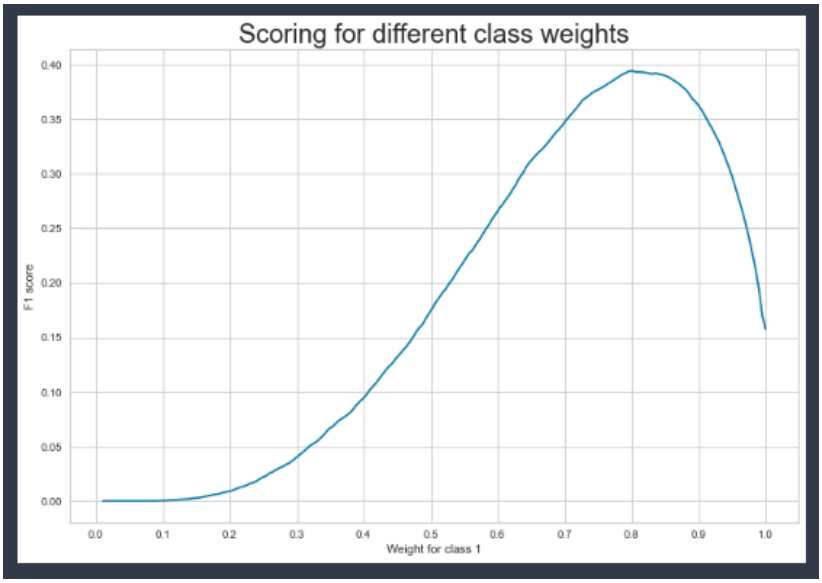
```
# Get columns to keep and create new dataframe with those only
cols = chi2_selector.get_support(indices=True)
features_df_new = features.iloc[:,cols]
features_df_new
```

	PhysicalHealth	Stroke_Yes	DiffWalking_Yes	Diabetic_Yes	GenHealth_Poor
0	10.0	0	0	0	0
1	0.0	0	0	0	0

- **Automatic & manual class balancing of the target (HeartDisease)**
  - **Class Balancing:** "A classification problem in machine learning is where we have given some input (independent variables), and we have to predict a discrete target. It is highly possible that the distribution of discrete values will be very different. Due to this difference in each class, the algorithms tend to get biased towards the majority values present and don't perform well on the minority values. This difference in class frequencies affects the overall predictability of the model." (Singh, 2021). In this dataset, the distribution of the target variable HeartDisease is very different (ten times as many with HeartDisease_No, count of 58497, compared to HeartDisease_Yes, count of 5462)
  - To remedy this class imbalance, two methods were used: automatic and manual class balancing. In automatic class balance logistic regression "We have added the class_weight parameter to our logistic regression algorithm and the value we have passed is 'balanced'." (Singh, 2021). In manual class balance logistic regression "we are trying to find optimal weights with the highest score using grid search. We will search for weights between 0 to 1. The idea is, if we are giving n as the weight for

the minority class, the majority class will get 1-n as the weights. Here, the magnitude of the weights is not very large but the ratio of weights between majority and minority class will be very high." (Singh, 2021).

- In this dataset, it was found through the graph below that the highest value for the minority class is peaking at about 0.81 class weight. Therefore, the minority class is 0.19. (class_weight={0: 0.19, 1: 0.81}).



- **Automatic class balance** was implemented into the logistic regression models using solver='newton-cg', class_weight='balanced'

```
#instantiate the logistic regression model
#logreg_model = LogisticRegression(max_iter = 500)
lr_weight = LogisticRegression(solver='newton-cg', class_weight='balanced')
#fit the model to the training set
lr_weight.fit(X_train, y_train)
# Predicting on the test data
pred_test_weight = lr_weight.predict(X_test)
```

- **Manual class balance** was implemented into the logistic regression models using solver='newton-cg', class_weight={0: 0.19, 1: 0.81}

```
#importing and training the model
from sklearn.linear_model import LogisticRegression
lr_weight_grid = LogisticRegression(solver='newton-cg', class_weight={0: 0.19, 1: 0.81})
lr_weight_grid.fit(X_train, y_train)

# Predicting on the test data
pred_test_weight_grid = lr_weight_grid.predict(X_test)

#Calculating and printing the f1 score
f1_test_grid = f1_score(y_test, pred_test_weight_grid)
print('The f1 score for the testing data:', f1_test_grid)

#Ploting the confusion matrix
confusion_matrix(y_test, pred_test_weight_grid)
```

- **Build and evaluate at least one model**

Model	Accuracy	F1 Score (Yes-Heart Disease)	F1 Score (No-Heart Disease)	AUC Score
Decision Tree Classifier - all 37 features	86.44%	0.250	0.925	0.59
Decision Tree Classifier - 5 best features using X2	91.5%	0.060	0.955	0.59
Logisitic Regression - No class weighting using PCA (34 features)	91.53%	0.171	0.955	0.83
Logistic Regression - No class weighing using all 37 features	91.71%	0.187	0.956	0.85
Logistic Regression - no class weighting using 5 best features using X2	91.46%	0.102	0.955	0.72
Logistic Regression - automatic class weighting of target variable with 37 features	75.12%	0.353	0.846	0.85
Logistic Regression - Manual class weighting of target variable with PCA (34 features)	87.01%	0.380	0.927	0.83
Logistic Regression - manual class weighting of target variable with 37 features	87%	0.402	0.927	0.85
Logistic Regression - manual class weighting of target variable with 5 best features using X2	88.61%	0.312	0.938	0.72

  - **Accuracy:** "Accuracy represents the number of correctly classified data instances over the total number of data instances" (B, H. N., 2020)
  - **F1 Score:** "F1-score is a metric which takes into account both precision and recall.
    - **Precision:** positive predictive value
    - **Recall:** true positive rate" (B, H. N., 2020)
  - **AUC Score:** ""So, what area under the ROC curve describes good discrimination? Unfortunately there is no "magic" number, only general guidelines. In general, we use the following rule of thumb:
    - 0.5 = This suggests no discrimination, so we might as well flip a coin.
    - 0.5-0.7 = We consider this poor discrimination, not much better than a coin toss.
    - 0.7-0.8 = Acceptable discrimination
    - 0.8-0.9= Excellent discrimination
    - >0.9 = Outstanding discrimination" (Dixon, 2020)
- **Interpret your results**

- **Decision Tree Classifier:** The Decision Tree models with all 37 features and the 5 best features should not be used to find the features that impact heart disease the most as its AUC score was less than that of the Logistic Regression Models, and overall, not much better than random chance (0.5)
- **Logistic Regression Models:** 3 logistic regression models were created to find the best model performance *without class balancing*
    - All 3 logistic regression models created without class balancing (all 37 features, PCA with 34 features, and 5 best features from X2) had relatively the same performance (91% accuracy, 0.1 F1 score for minority class (Yes - heart disease), 0.95 F1 score for majority class (No – heart disease), and 0.7-0.8 AUC score.
    - In all 3 regression models, the F1 score for minority class (Yes - heart disease) is low due to the class imbalance of the target variable (heart disease), and therefore the class should be balanced for the target variable, and the logistic regression model run again (see the next 4 models created below)
- **Logistic Regression Models:** 4 logistic regression models were created to find the best model performance *with class balancing*
    - In this dataset, the distribution of the target variable HeartDisease is very different (many more HeartDisease_No compared to HeartDisease_Yes), and therefore the previous 3 logistic regression models were biased towards the majority class (HeartDisease_No). This is shown in the previous 3 logistic regression model performance metrics where the F1 score for HeartDisease_No is > 0.9, and HeartDisease_Yes is < 0.2
    - All 4 logistic regression models created with class balancing (automatic with all 37 features, manual PCA with 34 features, manual with all 37 features, and manual with 5 best features from X2) had relatively the same performance (75-88% accuracy, 0.3-0.4 F1 score for minority class (Yes - heart disease), 0.84-0.93 F1 score for majority class (No – heart disease), and 0.7-0.85 AUC score.
    - In all 4 regression models with class balancing, the F1 score for minority class (Yes - heart disease) improved from the 3 regression models without class balancing. These performance metrics prove that the class balancing efforts did what they needed to do in terms of improving the model performance, specifically F1 on the minority class.
    - Although the manual class balancing logistic regression models had relatively the same performance, the fact that the 5 best features from X2 did relatively the same as both all 37 features and PCA (34 features) lends itself to show that these 5 best features (PhysicalHealth, Stroke_Yes, DiffWalking_Yes, Diabetic_Yes, GenHealth_Poor) have a bigger impact on the target variable (HeartDisease) than any of the other variables combined

- **Begin to formulate a conclusion/recommendations**
    - A Logistic Regression Model and Decision Tree Classifier were used on the dataset to determine which features are most related to or most predictive of heart disease, where the HeartDisease feature was the target.
    - **Matrices & Tests used to evaluate features in relation to the target variable, HeartDisease**

Matrices & Tests	Features

**Pearson's & Spearman's Correlation Matrix
- Feature correlation to Heart Disease target variable**
	• **Pearson's Correlation Matrix (numeric features)**
	  ○ 'PhysicalHealth'
	• **Spearman's Correlation Matrix (categorical features)**
	  ○ 'Stroke_Yes'
	  ○ 'DiffWalking_Yes'
	  ○ 'AgeCategory_80 or older'
	  ○ 'Diabetic_Yes'
	  ○ 'GenHealth_Fair'
	  ○ 'GenHealth_Poor'
	  ○ 'KidneyDisease_Yes'

Chi-Square (X2) Test - 5 best features correlated to Heart Disease	• PhysicalHealth • Stroke_Yes, • DiffWalking_Yes, • Diabetic_Yes • GenHealth_Poor

- **Findings:** The Pearson's & Spearman's Correlation Matrix features and the Chi-Squared (X2) features complement each other (with the exceptions of 'AgeCategory_80 or older', 'GenHealth_Poor' and 'KidneyDisease_Yes'), providing more evidence that the X2 5 best features are accurate

- **Recommendations:**
  - The logistic regression with class balancing using X2 5 best features had about the model performance as all 37 features in the same type of model (logistic regression with class balancing). As a recommendation, the X2 5 best features are easier to implement in a health care environment/setting than all 37 (i.e., it's easier to inform healthcare professionals on 5 features that impact or are most related to heart disease, rather than 37 features)
  - When advising healthcare professionals and patients on what features are most related to or most predictive of heart disease, they are the 5 best features extrapolated from X2 with good logistic regression modeling performance:
    1. **PhysicalHealth:** Subject's report of how many days in the past 30 days they have exercised (integer)
    2. **Stroke_Yes:** Subject has been told or has known that they have had a stroke
    3. **DiffWalking_Yes:** Subject's opinion is that they have difficulty walking or climbing stairs
    4. **Diabetic_Yes:** Subject has had or currently has diabetes
    5. **GenHealth_Poor:** Subject's opinion of their general health is poor
  - When healthcare professionals are advising patients on heart health, these are the 5 recommendations to provide:

1. You should have as much daily exercise in a month as possible
2. You should avoid anything that can cause or increase your risk of stroke (e.g., high blood pressure, high cholesterol)
3. You should keep up and maintain your ability to walk up or climb stairs easily
4. You should avoid anything that can cause or increase your risk of diabetes (e.g., body mass index over 25, inactivity, high blood pressure)
5. You should have a good routine related to your own general health (e.g., exercise and nutritional habits)

References

1. CDC. (2022, February 7). *Heart disease facts*. Centers for Disease Control and Prevention. Retrieved March 21, 2022, from https://www.cdc.gov/heartdisease/facts.htm
2. Abbott, D. (2014). *Applied predictive analytics: Principles and techniques for the professional data analyst*. Wiley.
3. El Khouli, R. H., Macura, K. J., Barker, P. B., Habba, M. R., Jacobs, M. A., & Bluemke, D. A. (2009, November). Relationship of temporal resolution to diagnostic performance for dynamic contrast enhanced MRI of the breast. Journal of magnetic resonance imaging : JMRI. Retrieved February 20, 2022, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2935260/
4. Grover, K. (2020, June 23). *Advantages and disadvantages of logistic regression*. OpenGenus IQ: Computing Expertise & Legacy. Retrieved March 21, 2022, from https://iq.opengenus.org/advantages-and-disadvantages-of-logistic-regression/

5. Mehta, S. (2022, March 16). *A beginner's guide to chi-square test in python from scratch*. Analytics India Magazine. Retrieved May 14, 2022, from https://analyticsindiamag.com/a-beginners-guide-to-chi-square-test-in-python-from-scratch/
6. Dixon, Chris. (2020). Re: What is the value of the area under the roc curve (AUC) to conclude that a classifier is excellent?. Retrieved from: https://www.researchgate.net/post/What-is-the-value-of-the-area-under-the-roc-curve-AUC-to-conclude-that-a-classifier-is-excellent/5eb02bdf39db6760dc56c904/citation/download.
7. B, H. N. (2020, June 1). *Confusion matrix, accuracy, precision, recall, F1 score*. Medium. Retrieved May 14, 2022, from https://medium.com/analytics-vidhya/confusion-matrix-accuracy-precision-recall-f1-score-ade299cf63cd#:~:text=F1%20Score%20becomes%201%20only,a%20better%20measure%20than%20accuracy.
8. Singh. (2021, January 6). *How to dealing with imbalanced classes in machine learning*. Analytics Vidhya. Retrieved May 15, 2022, from https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/