Aristotle University of Thessaloniki
Dept. Computer Science
MSc Artificial Intelligence

# Advanced Topics in Machine Learning

## Final Project

Kalitsios Georgios - kalitsios@csd.auth.gr
Pantelidou Konstantina - konpanpan@csd.auth.gr
Tsechelidou Konstantina – kmtsechel@csd.auth.gr

GitHub repository: https://github.com/constseche/Advanced-ML

Thessaloniki, 16/6/2021

# Table of Contents

# 1. Introduction

Nowadays, technology plays an important role to every society, as it is growing beyond the bounds, normally comprehensible by the human mind. Recent technology introduction has been following an exponential growth, as most clearly illustrated by the information technologies of computers, communications and the Internet. There are numerous other technologies emerging with the same or greater rapidity that will have as much or a greater effect. Especially, in healthcare industry, more and more technologies and techniques are being applied in order more solutions to be given faster and more accurately. Healthcare systems across the globe are struggling with increasing costs and worsening outcomes. As a result, it is widely said that a key part of the solution is Artificial Intelligence and machine learning techniques.

The aim of this report is the introduction on three different issues, based on healthcare data. The issues that are introduced are Class Imbalance, Cost-Sensitive learning, and Explainable Machine Learning.

# 2. Class Imbalance for Fetal Health

## Fetal Health Dataset

The dataset that was used was taken from Kaggle and it is related to fetal health classification.

Through the last few years, fetal mortality is an important problem that is observed, and as a result its reduction is a key indicator of human progress. Parallel to notion of child mortality is of course maternal mortality. Cardiotocograms (CTGs) are a simple and cost accessible option to assess fetal health, allowing healthcare professionals to take action in order to prevent child and maternal mortality.

This dataset consists of 22 features:

1. 'baseline value': This is the fetal heart rate baseline (beats per minute).
2. 'accelerations': Accelerations are short-term rises in the heart rate of at least 15 beats per minute, lasting at least 15 seconds.
3. 'fetal_movement': Fetal movement as felt by the pregnant woman and visualized by real-time ultrasonography is correlated to FHR accelerations in 52 normal and high-risk pregnant women.
4. 'uterine_contractions': Contractions are the tightening of the muscles of the uterus.
5. 'light_decelerations': Decelerations are temporary drops in the fetal heart rate.
6. 'severe_decelerations': Number of severe decelerations per second.
7. 'prolongued_decelerations': Decelerations are an abrupt decrease in the baseline fetal heart rate of greater than 15 bpm for greater than 15 seconds. A prolonged deceleration is defined as a deceleration that lasts more than 3 minutes.
8. 'abnormal_short_term_variability': It is the beat-to-beat variation in fetal heart rate. It is computed as mean difference between successive heart beat interval epochs in all analyzable one minute sections.
9. 'mean_value_of_short_term_variability': Mean value of short term variability
10. 'percentage_of_time_with_abnormal_long_term_variability'
11. 'histogram_width': Width of fetal heart rate histogram

12. 'histogram_min': Minimum (low frequency) of fetal heart rate histogram
13. 'histogram_max': Maximum (high frequency) of fetal heart rate histogram
14. 'histogram_number_of_peaks': Number of histogram peaks
15. 'histogram_number_of_zeroes': Number of histogram zeros
16. 'histogram_mode': The mode is the data value that occurs the most often in a data set. For a histogram, the values of the mean, median, and mode are all the same and are all located at the center of the distribution.
17. 'histogram_mean': Mean, also called "average", Sums up all the values in a column and divides them by the number of values in a histogram.
18. 'histogram_median': Histogram median gives the value that would be in the middle of an ordered list of your values. Ignores outliers.
19. 'histogram_variance': The curves looking like hills in a histogram represent clumps of data that are close together, hence a low variability. Variability in a histogram is higher when the taller bars are more spread out away from the mean and lower when the taller bars are close to the mean.
20. 'histogram_tendency'
21. 'mean_value_of_long_term_variability': Mean value of long term variability
22. 'fetal_health': This is feature is used for the target..

Then, the features are being plotted so that a relation to the target variable can be found. Let's understanding the data and features.

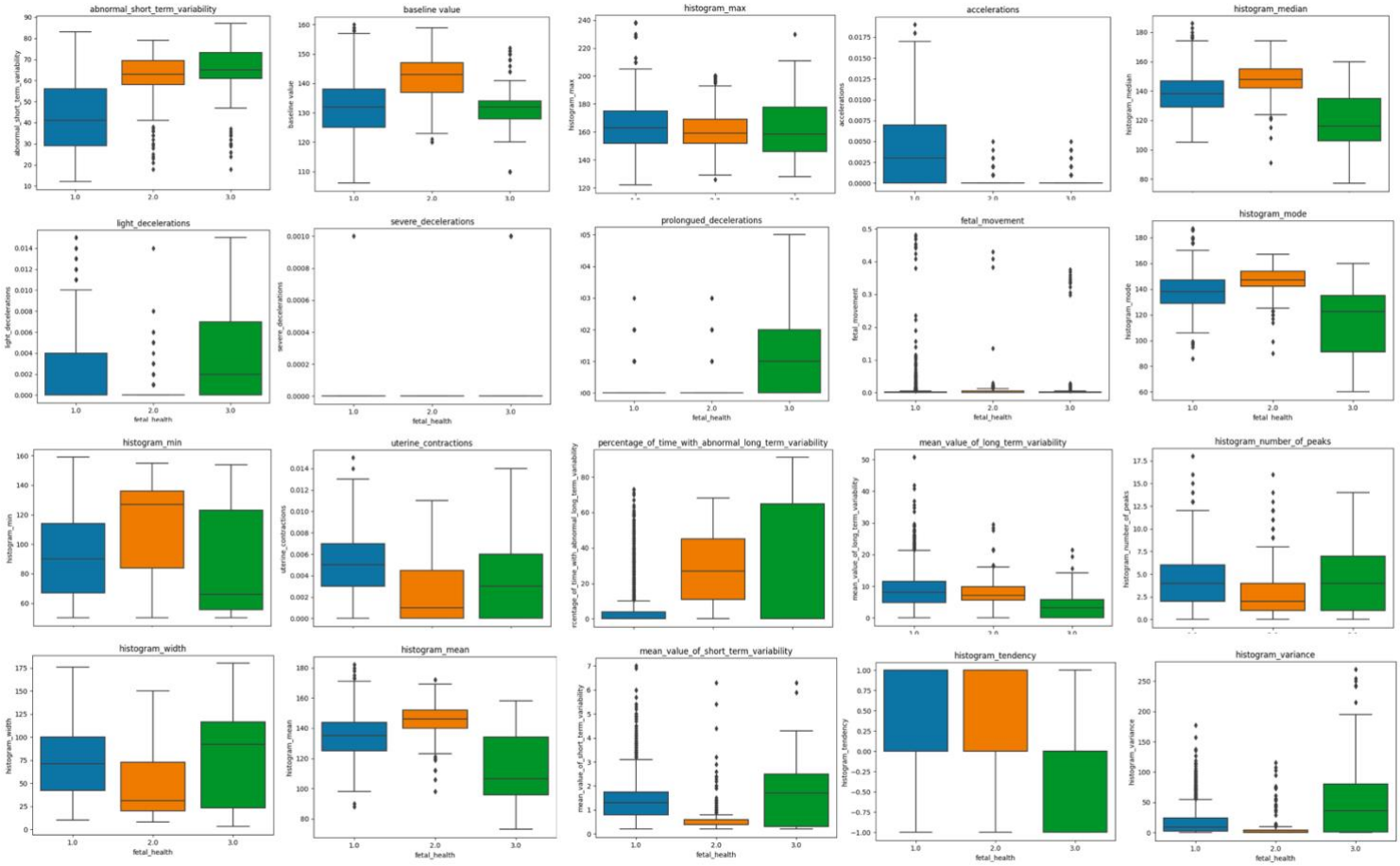



Figure 1-1: Box and whisker plot of features

Then a correlation matrix is displayed.

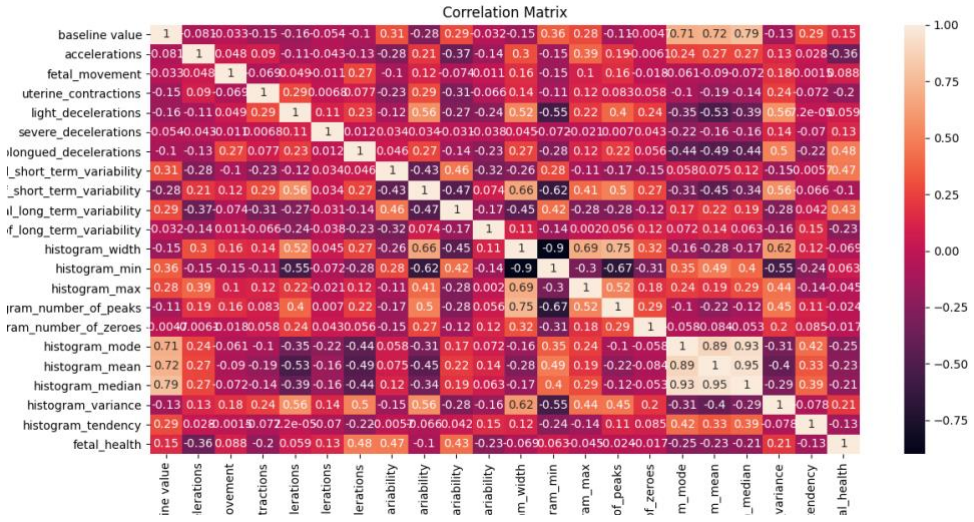



**Figure 1-1: Correlation Matrix**

Features with high correlation, like the following, are more linearly dependent and have almost the same effect on the dependent variable. Comparing the correlation between features, we could remove one of two features that have a correlation higher than 0.9, such as histogram_median - histogram_mode and histogram_median - histogram_mean.

Also, it is obvious from the above correlation matrix that the following features show some correlation with the target variable fetal health: abnormal short term variability (positive, 0.47), prolonged_decelerations (positive, 0.48), percentage of time with abnormal long term variability (positive, 0.43), accelerations (negative, -0.36)

**Pre-processing**: First, of all it is checked if there are any null or missing values on the dataset, but it is observed there are no such values. Then, data are scaled using StandardScaler() function.

After that, a plot is made where the class imbalance problem is obvious, as seen below, comparing the three classes.

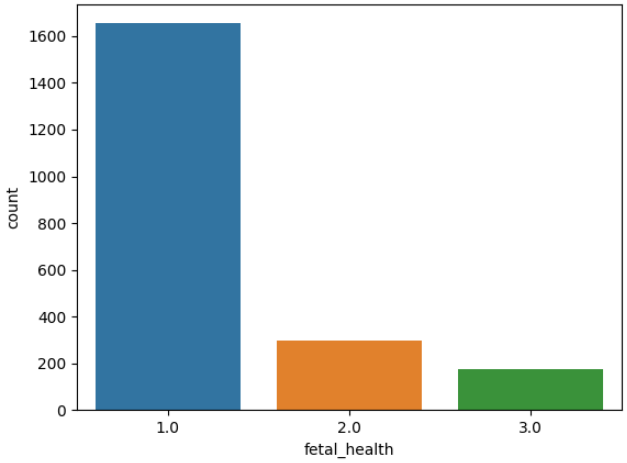



**Figure 3-1: Class imbalance between 3 classes**

The dataset contains 2126 records of features extracted from CTG exams, which were then classified into 3 classes:

- Normal (1655)
- Suspect (295)
- Pathological (176)

The normal class means that all four assessment criteria are normal. The suspect class means that at least one assessment criterion is suspicious and all the others are normal. Finally, the pathological class means that at least one assessment criterion is pathological or two or more are suspicious.

The class imbalance problem is obvious, so three different techniques were used to reduce the imbalance. To handle this problem, some over-sampling and under-sampling techniques were used and are described below. The techniques that were used were SMOTE, Tomek Links and the three versions of NearMiss.

**SMOTE** is an over-sampling technique. It creates synthetic samples of the minority class. This method plays an important role on imbalanced datasets, as it can make the minority class equal to the majority class. Smote manages this process by selecting similar records and altering that record one column at a time by a random amount within the difference to the neighboring records.

**Tomek Links** is an under-sampling technique that is developed by Tomek (1976) and is a modification from Condensed Nearest Neighbors. Unlike the CNN method that are only randomly select the samples with its k nearest neighbors from the majority class that wants to be removed, this method uses the rule to selects the pair of observation (say, a and b) that are fulfilled with some properties. More specifically, the observation a's nearest neighbor is b and the observation b's nearest neighbor is a and observation a and b belong to a different class, to the minority or majority class

**Near Miss** is an under-sampling technique that selects examples based on the distance of majority class examples to minority class examples. There are three versions of the technique, named NearMiss-1, NearMiss-2, and NearMiss-3.

NearMiss-1 selects examples from the majority class that have the smallest average distance to the three closest examples from the minority class.

NearMiss-2 selects examples from the majority class that have the smallest average distance to the three furthest examples from the minority class.

NearMiss-3 involves selecting a given number of majority class examples for each example in the minority class that are closest.

This distance is determined in feature space using Euclidean distance or similar. It is a fact that the third version seems ideal, because it will only keep those majority class examples that are on the decision boundary.

After this, some models were built. More specifically, DecisionTree, Random Forest, LinearSVC, AdaBoostClassifier, SGD algorithms were used.

The evaluation metrics that were estimated were F1 Score, Precision, Recall, Balanced Accuracy and Roc Auc Score. The results below show the scores for every algorithm after every technique was applied.

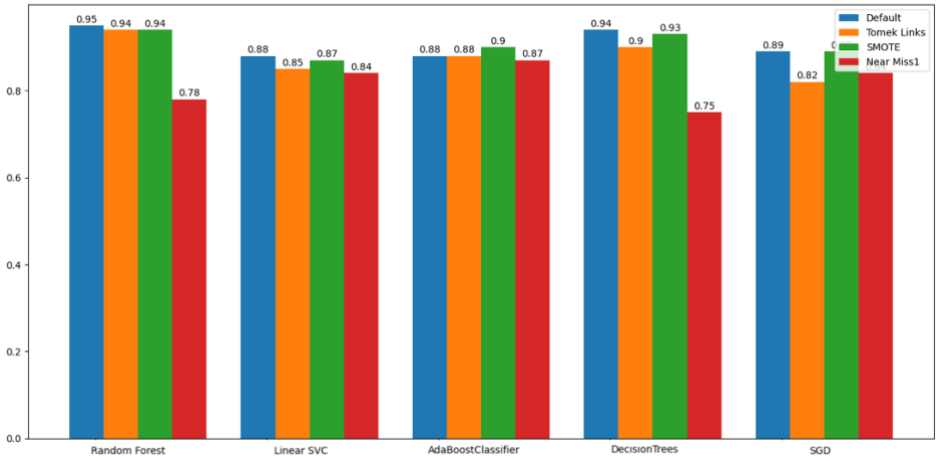



**Figure 3-2: F1 Score**

F1 score is an important metric for class imbalance problem, but we can easily understand that its value is not increased. This is obvious, because the default model, where no technique is applied, can achieve similar scores.

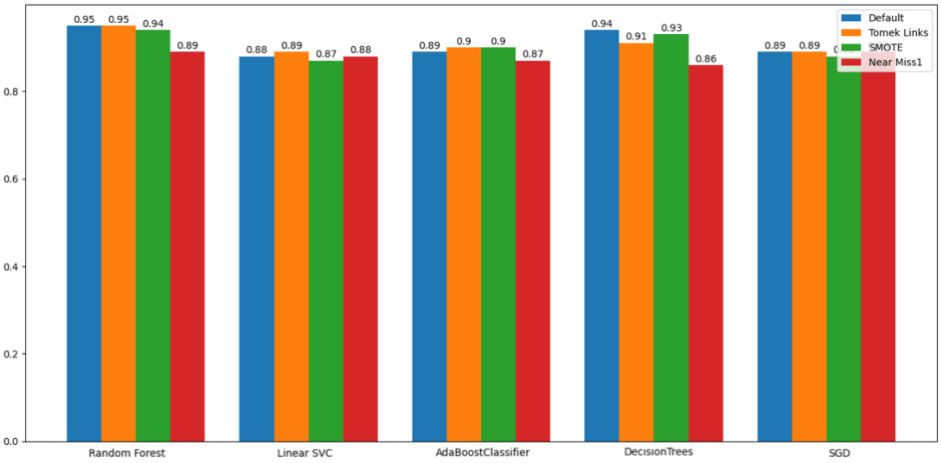



**Figure 3-3: Precision**

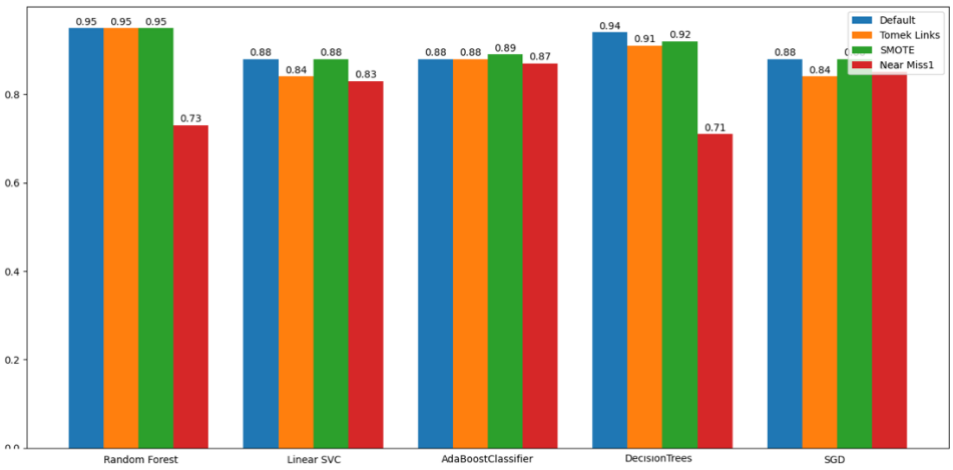



**Figure 3-4: Recall**

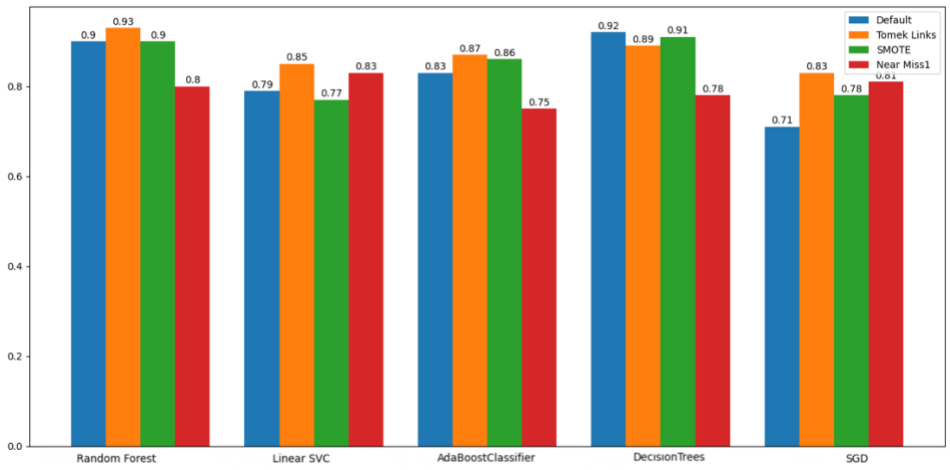



**Figure 3-5: Balanced Accuracy**

Balanced accuracy is a really good metric to use with imbalanced data, as it accounts for both the positive and negative outcome classes and doesn't mislead with imbalanced data. From the above plot, it can be seen that the model can achieve high scores, and especially higher scores than the default model.

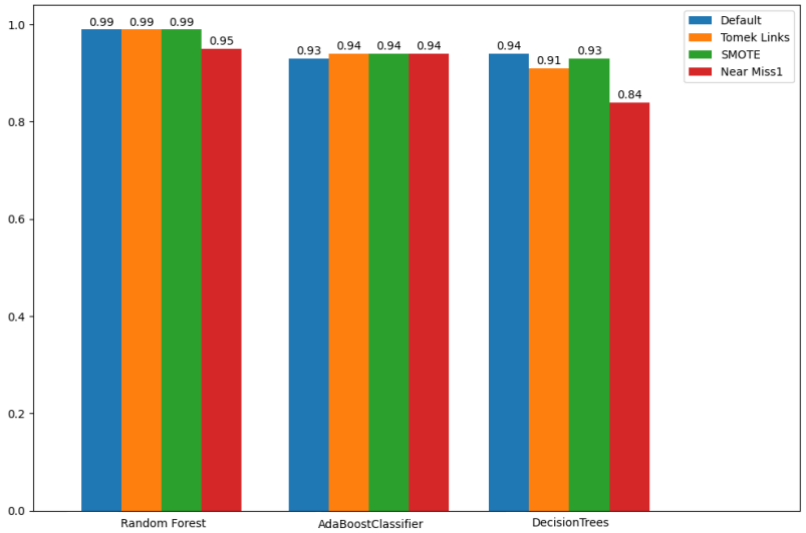



**Figure 3-6: Roc Auc Score**

The ROC AUC score is sensitive to class imbalance in the sense that when there is a minority class, it is defined as the positive class and it will have a strong impact on the AUC value. This is very much desirable behaviour. Accuracy is for example not sensitive in that way. It can be very high even if the minority class is not well predicted at all.

# 3. Cost-Sensitive Learning for Fetal Health

In most real-world applications and, specifically, in healthcare, a false negative is worse than a false positive prediction. Because of the imbalanced classification problem and the importance of diagnosing this medical condition we will also apply in this dataset (Fetal health Dataset) the following Cost-Sensitive Learning techniques, which will be examined below.

1. Class Weighting
2. Stratification aka Rebalancing
    a. Under-Sampling
    b. Over – Sampling
    c. Combination of the above
3. Costing-Rejection Sampling combining with hard voting (Ensemble models)
4. Minimizing the cost: (this is used only in binary classification problem because some costcla features don't offer possibilities for multiclass cost-matrix)
    a. Without calibration
    b. With sigmoid calibration
    c. With isotonic calibration

It is essential to minimize the occurrence of pathological and the suspicious cases being classified as normal, because not all classification errors are equal. So, we follow two scenarios. The first scenario indicates that the dataset has two classes, so we transformed it into binary problem. Then, a classic cost-matrix is used as shown in Table I. In fig. 1-1, total cost for a Random Forest Classifier using the Cost-Sensitive techniques number 1, 2,4 as listed above, are shown in figure 3-1. In most cases, when applying a cost-sensitive technique the total cost is decreased.

| | | Ground Truth | |
|---|---|---|---|
| | | **Normal** | **Pathological** |
| **Prediction** | **Normal** | 0 | 5 |
| | **Pathological** | 1 | 0 |

Table I: Binary Classification problem: Cost Matrix

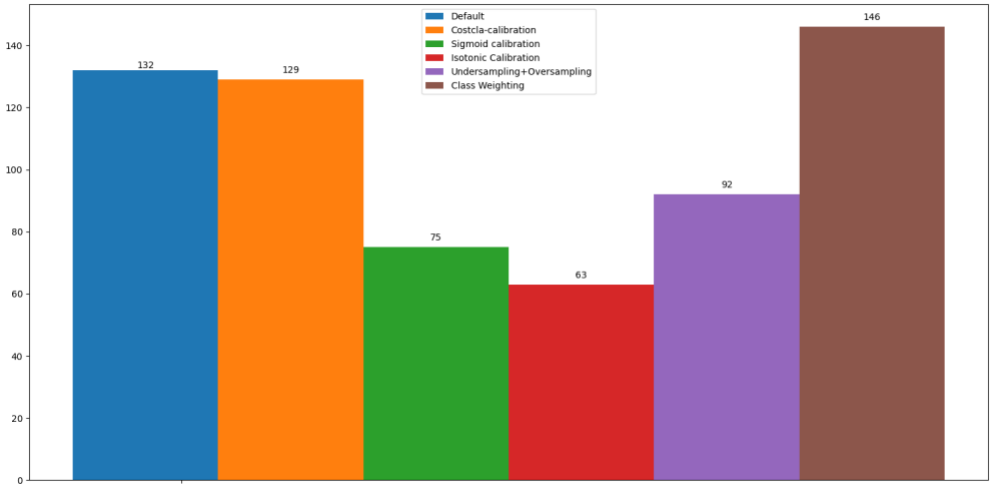



Figure 3-1: Total Cost for Random Forest Classifier using Cost-Sensitive techniques

In terms of the second scenario that we are mainly focusing on, we treated the dataset with three class, so we had to create a 3x3 cost-matrix, where a false "negative" penalty w1=5 and w2=4 are incurred whenever a pathological or suspicious are classified as normal, whereas a normal case is classified as pathological or suspicious simply incur a unit cost and zero cost presents when a class chosen is the correct class. Thus, given three classes, we typically use a cost matrix as shown below in Table II.

| | | **Ground Truth** | | |
|---|---|---|---|---|
| | | **Normal** | **Suspect** | **Pathological** |
| **Prediction** | **Normal** | 0 | 4 | 5 |
| | **Suspect** | 1 | 0 | 1 |
| | **Pathological** | 1 | 1 | 0 |

Table II: Multi-class Classification problem: Cost Matrix

We will train a Random Forest Classifier, a SVM with linear kernel and Gaussian Naïve Bayes model using mainly default parameters. The training set consists of: Normal: 1241, Suspect: 221, Pathological: 132

## 3.1 Stratification aka Rebalancing:

We rebalanced the classes according to their misclassification costs. So, for class $j$, we calculated the sum of all $C(i,j)$, that means the sum of each $j$ column and the result is 2, 5 and 6 accordingly.

a. Under-Sampling:
A Random Under-sampler was used in order to under-sampling the majority class (and achieve the following result for the number of examples for each class: 200 normal, 221 suspect and 132 pathological.

b. Over-Sampling:
A Random Over-sampler was used for the minorities classes, so the class contains examples of size: [1241, 1000, 1200]

c. Combination:
A Random Under-sampler and a Random Over-sampler are used. The weights are selected according to:

   i. under-sampling majority class: we have a dataset with a 1 to 6 ratio (and 1 to 5) of examples in the minority classes (pathological and suspect accordingly) to examples in the majority class. This ratio can be inverted and used as the cost of misclassification errors, as already defined from the cost-matrix.
   ii. over-sampling minority classes in order to achieve: [200, 1000, 1200] examples.

## 3.2 Class Weighting

We can weigh each example according to its misclassification cost. So, when using from sklearn the Random Forest Classifier or the SVC, we can use the "class_weights" parameter defined as below, or create a "sample weights" array containing the misclassification cost of each example for Gaussian Naïve Bayes model.

class weights: {1:2, 2:5, 3:6}

## 3.3 Rejection Sampling – Costing:

Multiple runs (20) of Rejection Sampling, combining with Hard Voting (Ensembe models)

a. z = 6
b. c = [2, 5, 6]

so the probability to keep an example will be 2/6, 5/6 and 1 accordingly for each class.

The total loss of default, combination of under-sampling and over-sampling, class weighting and costing- rejection sampling is shown below. It is obvious that in most cases, when applying a cost-sensitive technique the total loss is decreased. For example, in the case of Random Forest Classifier, the default loss is 109, but then it is decreased to 86, 99, 77 compared to three techniques.

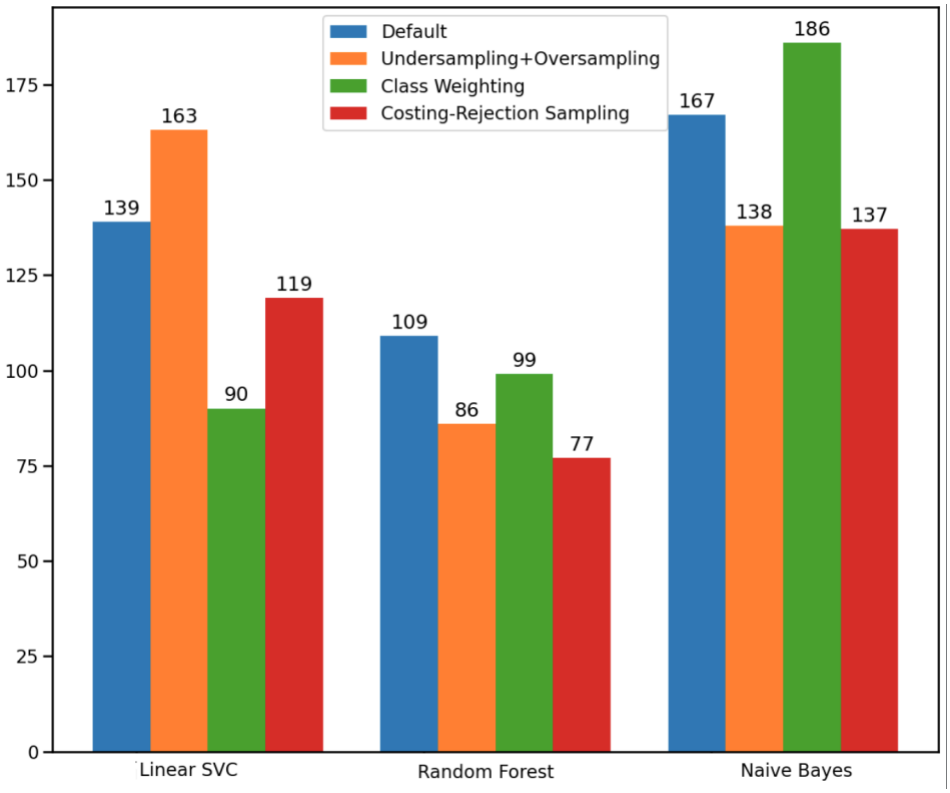



Figure 3-2: Total Cost of different training models

The confusion matrices (**rows:predictions and columns:ground truth**) of Linear SVC without a cost-sensitive approach (default) (fig. 1-10), over-sampling (fig. 1-11), under-sampling (fig. 1-12), combination (fig. 1-13), class weighting (fig.1-14) and rejection sampling-costing (fig. 1-15) are shown below. We could easily understand that, only after applying the techniques mentioned above, many cases of costly classes that were classified as normal, but the actual class was suspect or pathological, now they are classified correctly. For example, in terms of under-sampling, 21 cases (18 + 3) of suspect and pathological class are not misclassified as normal and this is what we actually tried to achieve, how many cases we could "save". Finally, in the fig. 1-16, the total result of how many cases are saved using Linear SVC model and applying the cost-sensitive techniques.

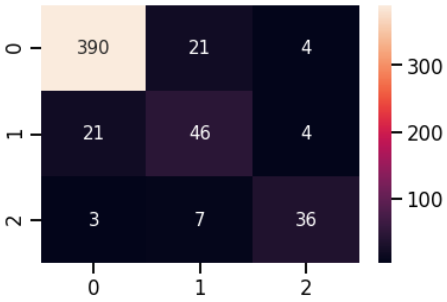


Figure3-3: Default

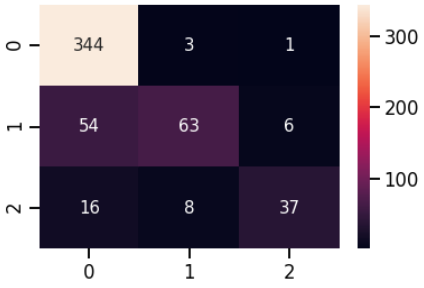


Figure 3-4: Over-sampling

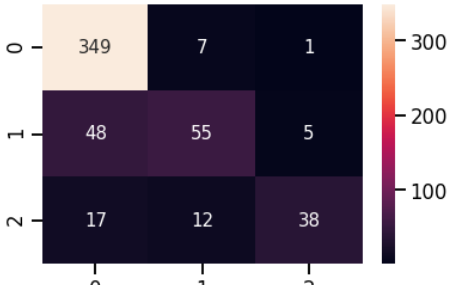


Figure 3-5: Under-sampling

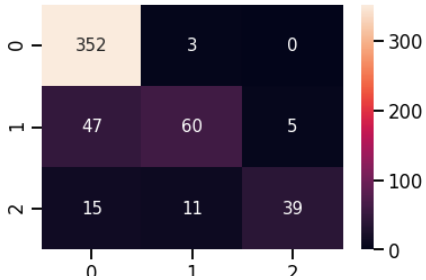


Figure 3-6: Combination

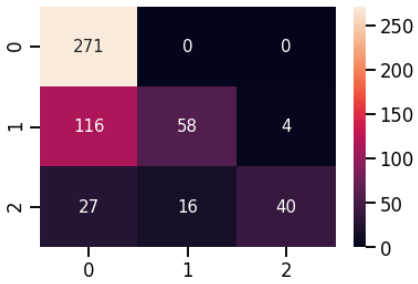


Figure 3-6: Class Weighting

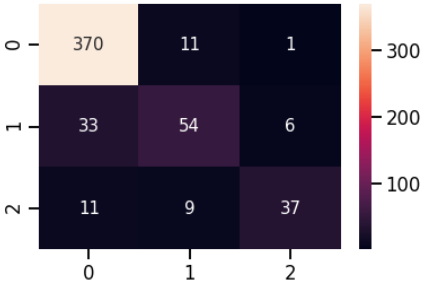


Figure 3-7: Rejection sampling-Voting

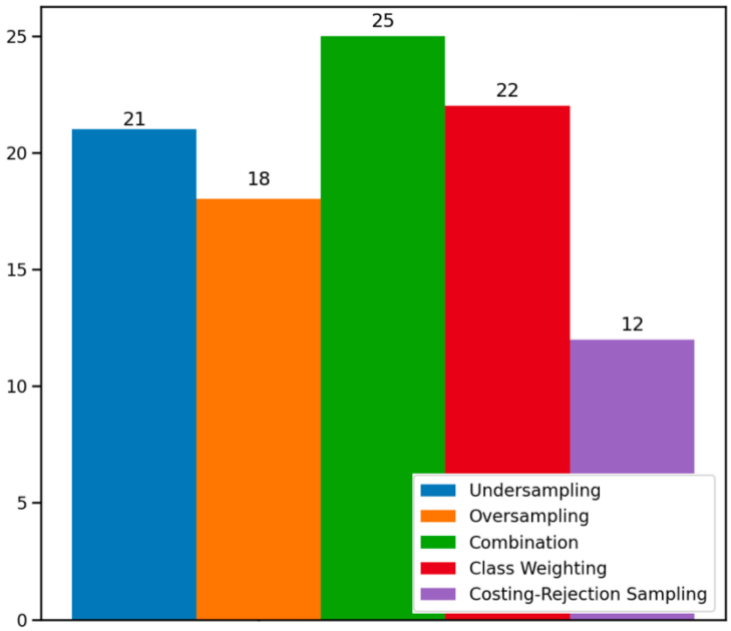


Figure 3-8: How many examples of the costly classes are now classified correctly, after applying Linear SVC with Cost-Sensitive techniques

# 4.Machine Learning Interpretability for Heart Disease Prediction

It’s time to get rid of the black boxes and cultivate trust in Machine Learning

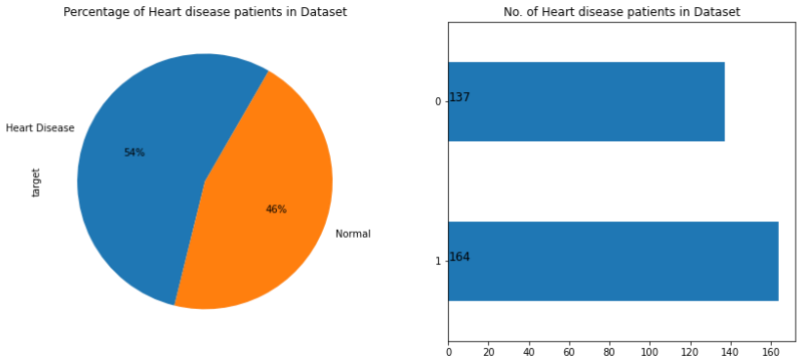
## Dataset

Heart Disease UCI | link : https://www.kaggle.com/ronitf/heart-disease-uci

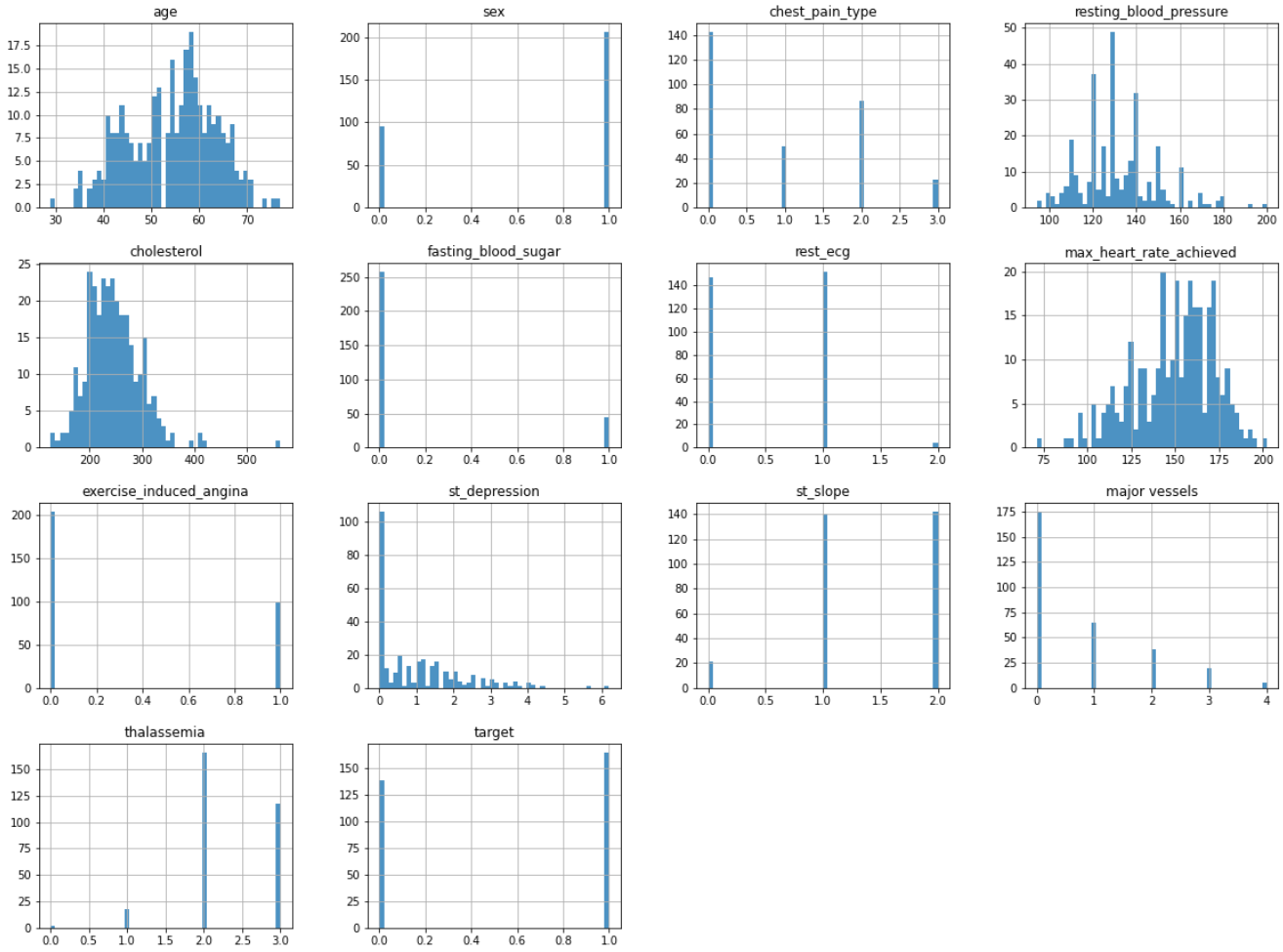
## Dataset features description

This dataset consists of 13 features and a target variable. The detailed description of all the features are as follows:

1. **Age :** Patients Age in years (Numeric)
2. **Sex :** Gender of patient [Male - 1, Female - 0] (Nominal)
3. **Chest Pain Type :** Type of chest pain experienced by patient categorized into [0 typical, 1 typical angina, 2 non- anginal pain, 3 asymptomatic] (Nominal) understanding :(https://www.harringtonhospital.org/typical-and-atypical-angina-what-to-look-for/)
4. **resting bps :** Level of blood pressure at resting mode in mm/HG (Numerical) understanding :(https://www.healthline.com/health/high-blood-pressure-hypertension/blood-pressure-reading-explained#normal) / https://www.hcs.gr/artiriaki-piesi.aspx
5. **cholestrol :** Serum cholestrol in mg/dl (Numeric) understanding :(https://www.mikroviologos.gr/arthra/cholesterol)
6. **fasting blood sugar :** Blood sugar levels on fasting > 120 mg/dl represents as 1 in case of true and 0 as false (Nominal)
7. **resting ecg :** Result of electrocardiogram while at rest are represented in 3 distinct values 0 : Normal 1: Abnormality in ST-T wave 2: Left ventricular hypertrophy (Nominal)
8. **max heart rate :** Maximum heart rate achieved (Numeric)
9. **exercise angina :** Angina induced by exercise 0 depicting NO 1 depicting Yes (Nominal)
10. **oldpeak :** Exercise induced ST-depression in comparison with the state of rest (Numeric)
11. **ST slope :** ST segment measured in terms of slope during peak exercise [0 = Normal, 1 = Upsloping, 2 = Flat 3: Downsloping ] (Nominal)
12. **ca :** The number of major vessels [0-4]
13. **thal :** A blood disorder called thalassemia [1 = normal, 2 = fixed defect, 3 = reversable defect] understanding :( https://bioiatriki.gr/index.php/thalassoaimies)
14. **target :** Heart disease [0 = no, 1 = yes]



**Dataset features**




## Training the models

We will train a random forest and logistic regression model.

Logistic regression is a readily interpretable model which allows us to determine the linear relationship between the features and the target. A random forest is a 'black box' model, which will require a little more work to interpret, but which will allow us to view the non-linear relationships between the features and the model predictions. As we can see from the above result random forest has equal F1 score, Precision, ROC and Recall with Logistic Regression but has lesser accuracy in comparison to Logistic Regression. But apart from that, we will be using Random forest for applying machine learning interpretation strategies. the major reason for selecting random forest is it is tree based algorithm which supports and compatible with most of the interpretation techniques.

| Model | Recall | Accuracy | F1 Score | Precision | ROC |
|---|---|---|---|---|---|
| Base - Logistic Regression | 0.878 | 0.802 | 0.878 | 0.878 | 0.867 |
| Random Forest | 0.878 | 0.788 | 0.865 | 0.852 | 0.850 |

## Feature Importance

### Native methods

Global importance measures. i.e they answer the question 'On average, how important is feature i for making a prediction'.

Both random forests and logistic regression have native methods of formulating the relative importance of features.

- Decision trees come with a method for measuring the importance of a feature which works by adding up the total decrease in the gini coefficient from nodes that depend on that feature. For a random forest, we can average this value across all trees to get an estimate of festure importance. This is called the mean decrease in gini.
- For logistic regression, the importances of a feature is just given by the absolute value of its weight. Note that this is only true since we have standardised the variables. Of course the weights themselves have a straightforward interpretation in terms of how the associated feature affects the predicted outcome.

Random Forest:

| Weight | Feature |
|---|---|
| 0.1304 ± 0.3144 | thalassemia_fixed defect |
| 0.1231 ± 0.2355 | st_depression |
| 0.1147 ± 0.2075 | major vessels |
| 0.0986 ± 0.1801 | max_heart_rate_achieved |
| 0.0798 ± 0.2276 | chest_pain_type_typical angina |
| 0.0614 ± 0.1811 | thalassemia_reversable defect |
| 0.0600 ± 0.1019 | age |
| 0.0464 ± 0.1577 | st_slope_downsloping |
| 0.0455 ± 0.0894 | cholesterol |
| 0.0396 ± 0.0742 | resting_blood_pressure |
| 0.0393 ± 0.1484 | exercise_induced_angina_no |
| 0.0353 ± 0.1348 | exercise_induced_angina_yes |
| 0.0228 ± 0.0676 | chest_pain_type_non-anginal pain |
| 0.0222 ± 0.0759 | st_slope_flat |
| 0.0218 ± 0.0632 | sex_male |
| 0.0124 ± 0.0442 | sex_female |
| 0.0097 ± 0.0375 | rest_ecg_ST-T wave abnormality |
| 0.0095 ± 0.0306 | rest_ecg_normal |
| 0.0076 ± 0.0495 | chest_pain_type_atypical angina |
| 0.0063 ± 0.0274 | chest_pain_type_asymptomatic |
| | *… 5 more …* |

Logistic Regression:

**y=1** top features

| Weight[?] | Feature |
|---|---|
| +1.179 | <BIAS> |
| +0.803 | max_heart_rate_achieved |
| +0.731 | st_slope_downsloping |
| +0.702 | thalassemia_fixed defect |
| +0.495 | chest_pain_type_non-anginal pain |
| +0.481 | sex_female |
| +0.452 | exercise_induced_angina_no |
| +0.430 | chest_pain_type_atypical angina |
| +0.405 | rest_ecg_ST-T wave abnormality |
| | *… 2 more positive …* |
| | *… 4 more negative …* |
| -0.278 | rest_ecg_left ventricular hypertrophy |
| -0.304 | st_slope_flat |
| -0.349 | age |
| -0.427 | st_slope_upsloping |
| -0.452 | exercise_induced_angina_yes |
| -0.481 | sex_male |
| -0.659 | thalassemia_reversable defect |
| -0.788 | major vessels |
| -0.979 | chest_pain_type_typical angina |
| -1.120 | st_depression |
| -1.182 | resting_blood_pressure |

**One thing that both methods have in common is that they examine which features play the biggest role in fitting to the training set, not which features are most useful for generalization. If we want to know which risk factors are the most impactful on one's chance of developing heart disease, it is really this second kind of feature importance that we want.**

There are other drawbacks that are specific to each model:

- Random forest: Mean decrease in gini tends to overestimate the importance of continuous and high cardinality categorical data. It also underestimates the importance of highly correlated features.
- Logistic regression: By its nature, logistic regression can only model linear relationships between the features and target, and this obviously limits its ability to demonstrate the importance of features which have a non-linear effect on the target.

The primary advantage of both is that they are trivial to compute once you have fitted the model.

## First Technique - Permutation Importance

**What features does a model think are important ? Which features might have a greater impact on the model predictions than the others ?**

This concept is called feature importance and Permutation Importance is a technique used widely for calculating feature importance. It helps us to see when our model produces counterintuitive results, and it helps to show the others when our model is working as we'd hope.

Permutation Importance works for many scikit-learn estimators. The idea is simple: Randomly permutate or shuffle a single column in the validation dataset leaving all the other columns intact. A feature is considered "important" if the model's accuracy drops a lot and causes an increase in error. On the other hand, a feature is considered 'unimportant' if shuffling its values doesn't affect the model's accuracy.

**Permutation importance is calculated after a model has been fitted!**

Permutation Importance

| Weight | Feature |
|---|---|
| 0.0426 ± 0.0161 | chest_pain_type_typical angina |
| 0.0393 ± 0.0161 | thalassemia_reversable defect |
| 0.0262 ± 0.0334 | rest_ecg_ST-T wave abnormality |
| 0.0230 ± 0.0445 | major vessels |
| 0.0230 ± 0.0262 | thalassemia_fixed defect |
| 0.0164 ± 0.0207 | chest_pain_type_atypical angina |
| 0.0164 ± 0.0207 | rest_ecg_normal |
| 0.0033 ± 0.0482 | st_depression |
| 0.0033 ± 0.0131 | exercise_induced_angina_no |
| 0.0033 ± 0.0131 | thalassemia_normal |
| 0.0000 ± 0.0293 | cholesterol |
| 0.0000 ± 0.0359 | sex_female |
| 0.0000 ± 0.0207 | age |
| 0 ± 0.0000 | rest_ecg_left ventricular hypertrophy |
| 0 ± 0.0000 | fasting_blood_sugar_greater than 120mg/ml |
| 0 ± 0.0000 | fasting_blood_sugar_lower than 120mg/ml |
| -0.0033 ± 0.0245 | sex_male |
| -0.0033 ± 0.0131 | st_slope_downsloping |
| -0.0033 ± 0.0131 | st_slope_upsloping |
| -0.0066 ± 0.0161 | chest_pain_type_non-anginal pain |
| -0.0066 ± 0.0161 | chest_pain_type_asymptomatic |
| -0.0098 ± 0.0161 | st_slope_flat |
| -0.0098 ± 0.0262 | exercise_induced_angina_yes |
| -0.0230 ± 0.0334 | max_heart_rate_achieved |
| | *… 1 more …* |

**Interpretation :**

- The features at the top are most important and at the bottom, the least.
- The number after the ± measures how performance varied from one-reshuffling to the next.
- Some weights are negative. This is because in those cases predictions on the shuffled data were found to be more accurate than the real data.

**Here top 5 important features :**

- **chest_pain_type_typical angina**
- **thalassemia_reversable defect**
- **rest_ecg_ST-T wave abnormality**
- **major vessels**
- **thalassemia_fixed defect**

Next, to explain individual prediction by random forest model there is a method in eli5 library called show_prediction().

- We explain the the 10th record of test set having following prediction as shown in below figure.

**y=1** (probability **0.820**) top features

| Contribution[?] | Feature | Value |
|---|---|---|
| +0.547 | <BIAS> | 1.000 |
| +0.075 | max_heart_rate_achieved | 0.786 |
| +0.068 | major vessels | 0.000 |
| +0.062 | thalassemia_fixed defect | 1.000 |
| +0.056 | chest_pain_type_typical angina | 0.000 |
| +0.043 | st_slope_downsloping | 1.000 |
| +0.039 | thalassemia_reversable defect | 0.000 |
| +0.023 | exercise_induced_angina_yes | 0.000 |
| +0.019 | st_slope_flat | 0.000 |
| +0.014 | chest_pain_type_non-anginal pain | 1.000 |
| +0.014 | exercise_induced_angina_no | 1.000 |
| +0.004 | chest_pain_type_asymptomatic | 0.000 |
| +0.004 | rest_ecg_ST-T wave abnormality | 1.000 |
| +0.001 | thalassemia_normal | 0.000 |
| +0.000 | st_slope_upsloping | 0.000 |
| +0.000 | rest_ecg_normal | 0.000 |
| +0.000 | rest_ecg_left ventricular hypertrophy | 0.000 |
| -0.000 | fasting_blood_sugar_greater than 120mg/ml | 0.000 |
| -0.002 | chest_pain_type_atypical angina | 0.000 |
| -0.009 | fasting_blood_sugar_lower than 120mg/ml | 1.000 |
| -0.015 | st_depression | 0.258 |
| -0.018 | sex_male | 1.000 |
| -0.021 | sex_female | 0.000 |
| -0.022 | resting_blood_pressure | 0.528 |
| -0.029 | cholesterol | 0.085 |
| -0.035 | age | 0.622 |

To make random forest predictions more interpretable, every prediction of the model can be presented as a sum of feature contributions (plus the bias), showing how the features lead to a particular prediction. In above plot, ELI5 does it by showing weights for each feature with their actual value depicting how influential it might have been in contributing to the final prediction decision across all trees.

In the above individual prediction, the top 3 influential features seems to be, after the bias,

- the major vessels,
- thalassemia_fixed defect
- max_heart_rate_achieved.

- 42th record of test set having following prediction as shown in below figure :

**y=0** (probability **0.830**) top features

| Contribution[?] | Feature | Value |
|---|---|---|
| +0.453 | <BIAS> | 1.000 |
| +0.093 | major vessels | 3.000 |
| +0.084 | thalassemia_fixed defect | 0.000 |
| +0.080 | chest_pain_type_typical angina | 1.000 |
| +0.077 | thalassemia_reversable defect | 1.000 |
| +0.053 | max_heart_rate_achieved | 0.328 |
| +0.050 | st_depression | 0.161 |
| +0.028 | st_slope_downsloping | 0.000 |
| +0.016 | st_slope_flat | 1.000 |
| +0.010 | chest_pain_type_non-anginal pain | 0.000 |
| +0.010 | rest_ecg_normal | 1.000 |
| +0.008 | resting_blood_pressure | 0.528 |
| +0.007 | chest_pain_type_atypical angina | 0.000 |
| +0.006 | rest_ecg_ST-T wave abnormality | 0.000 |
| +0.003 | thalassemia_normal | 0.000 |
| +0.002 | chest_pain_type_asymptomatic | 0.000 |
| -0.000 | rest_ecg_left ventricular hypertrophy | 0.000 |
| -0.000 | fasting_blood_sugar_greater than 120mg/ml | 0.000 |
| -0.003 | st_slope_upsloping | 0.000 |
| -0.004 | fasting_blood_sugar_lower than 120mg/ml | 1.000 |
| -0.015 | sex_male | 0.000 |
| -0.016 | cholesterol | 0.217 |
| -0.027 | age | 0.800 |
| -0.027 | sex_female | 1.000 |
| -0.028 | exercise_induced_angina_yes | 0.000 |
| -0.029 | exercise_induced_angina_no | 1.000 |

In the above individual prediction, the top 3 influential features seems to be, after the bias,

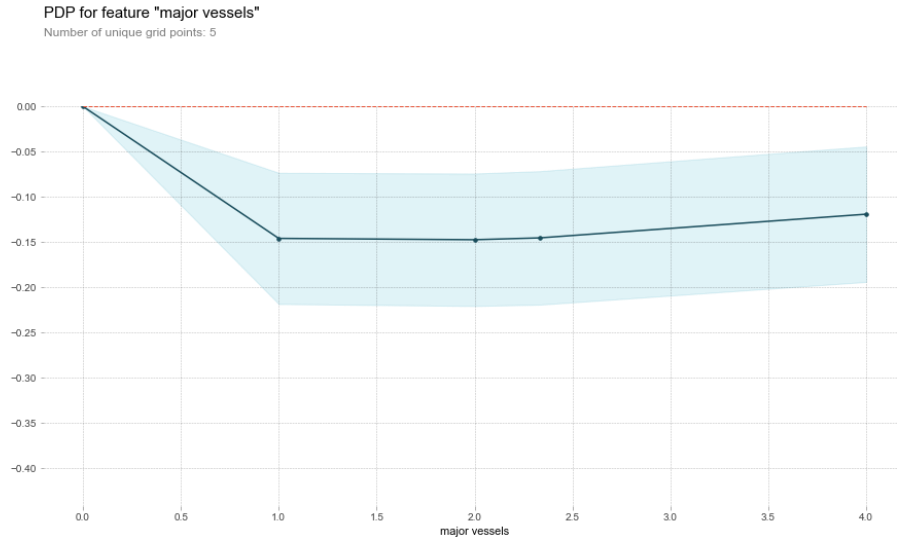
- major vessels
- thalassemia_fixed defect
- chest_pain_type_typical angina

## Second Technique - Partial Dependence Plots

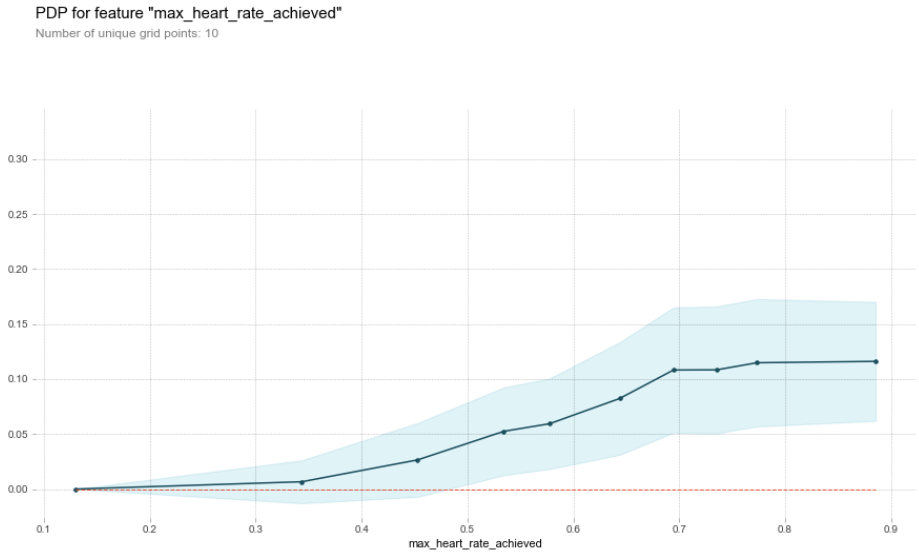
### How does each feature affect your predictions ?

The partial dependence plot (short PDP or PD plot) shows the marginal effect one or two features have on the predicted outcome of a machine learning model. PDPs show how a feature affects predictions. PDP can show the relationship between the target and the selected features via 1D or 2D plots.

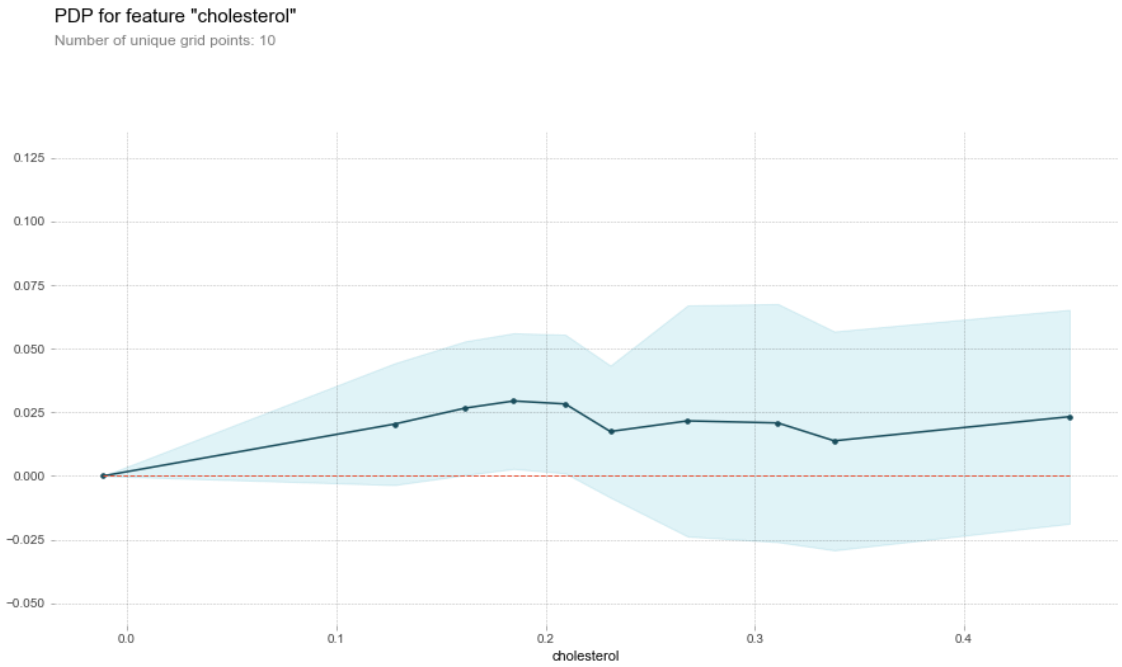
Like permutation importance, partial dependence plots are calculated after a model has been fit. The model is fit on real data that has not been artificially manipulated in any way.
A few items are worth pointing out as you interpret this plot The y axis is interpreted as change in the prediction from what it would be predicted at the baseline or leftmost value. A blue shaded area indicates level of confidence.



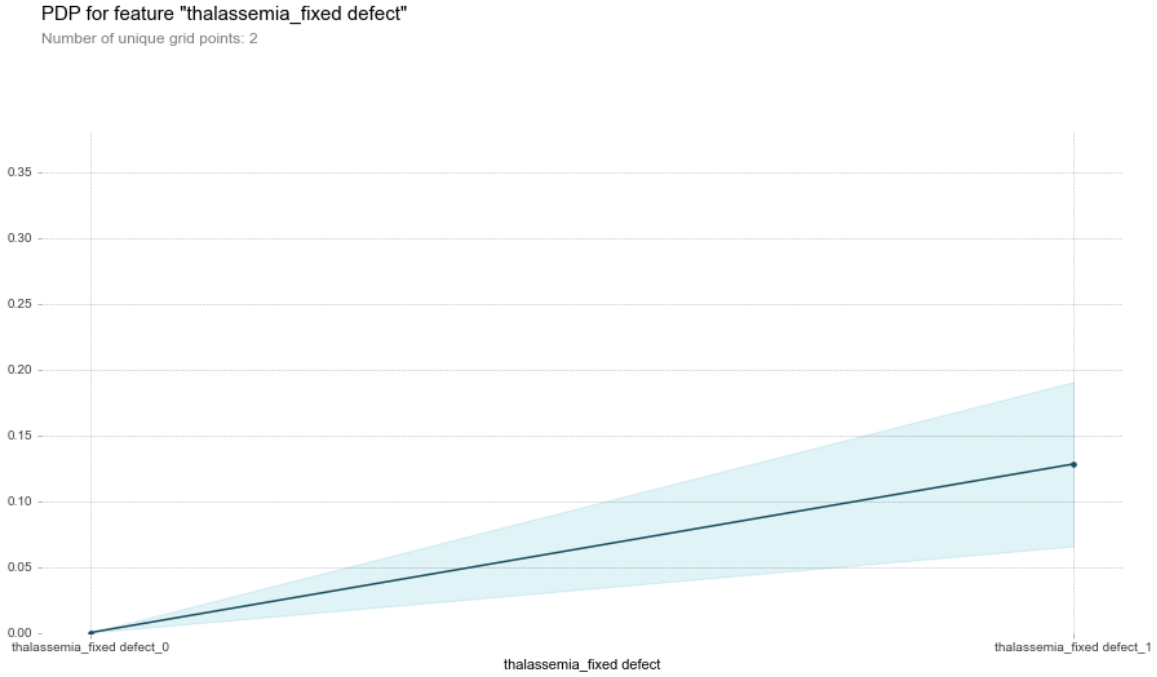

- **So, we can see that as the number of major blood vessels increases, the probability of heart disease decreases. That makes sense, as it means more blood can get to the heart.**



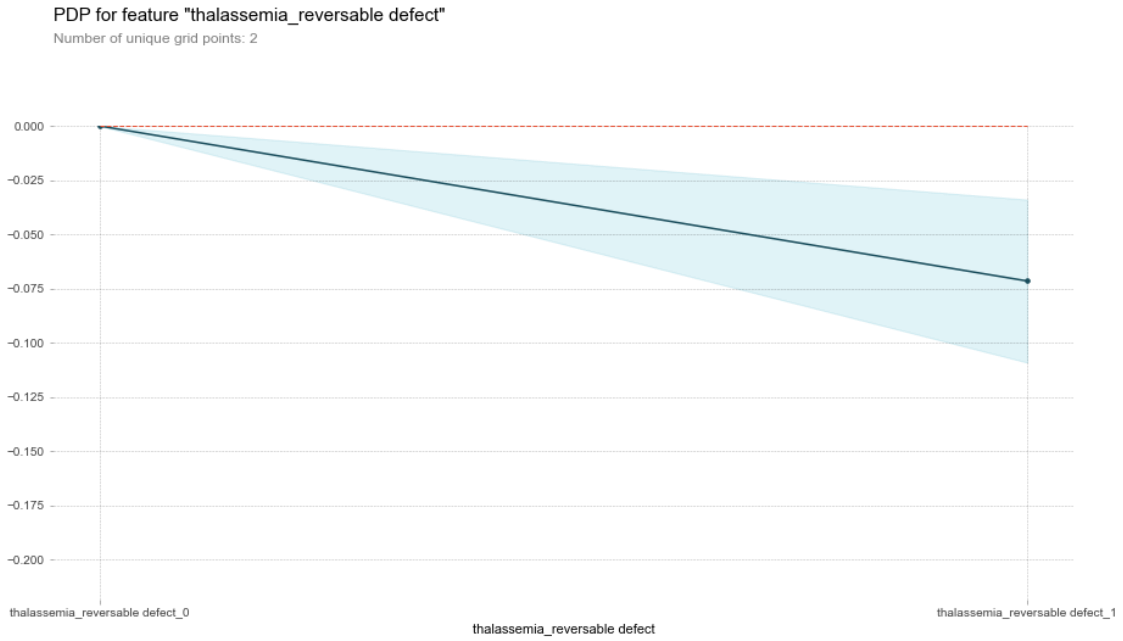

- **Here we can see when max_heart_rate_achieved increases, the probability of heart disease increases. That makes sense**



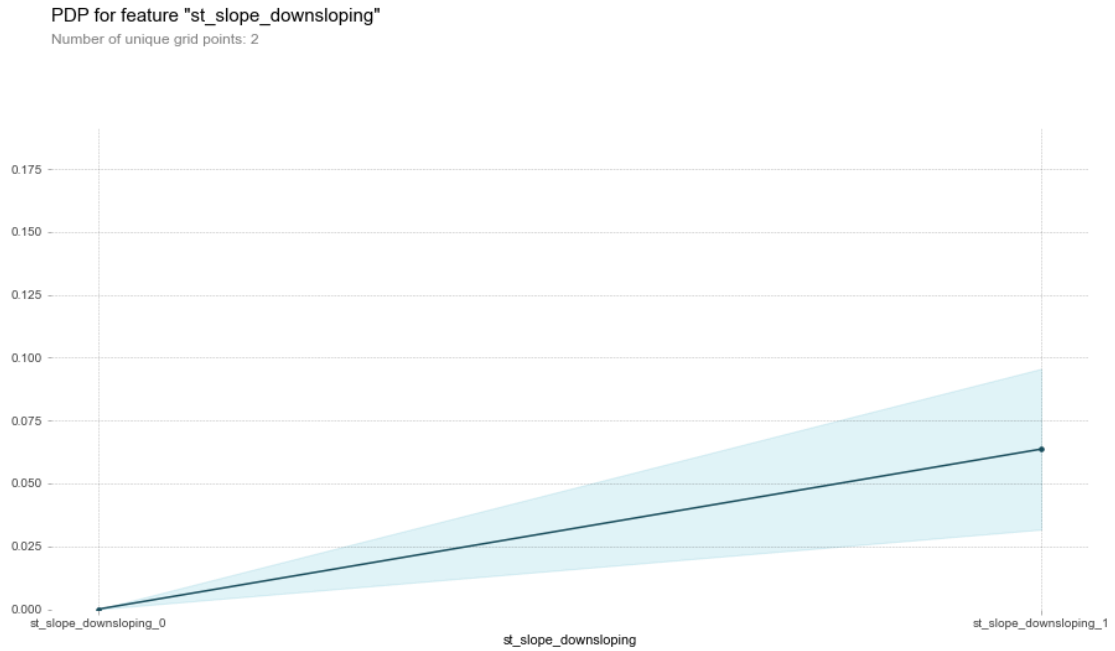

- **we can see when cholesterol increases, the probability of heart disease increases. That makes sense,because high cholesterol increases the risk of cardiovascular disease due to a) an increase in blood pressure and b) the extra load that the heart has to face.**



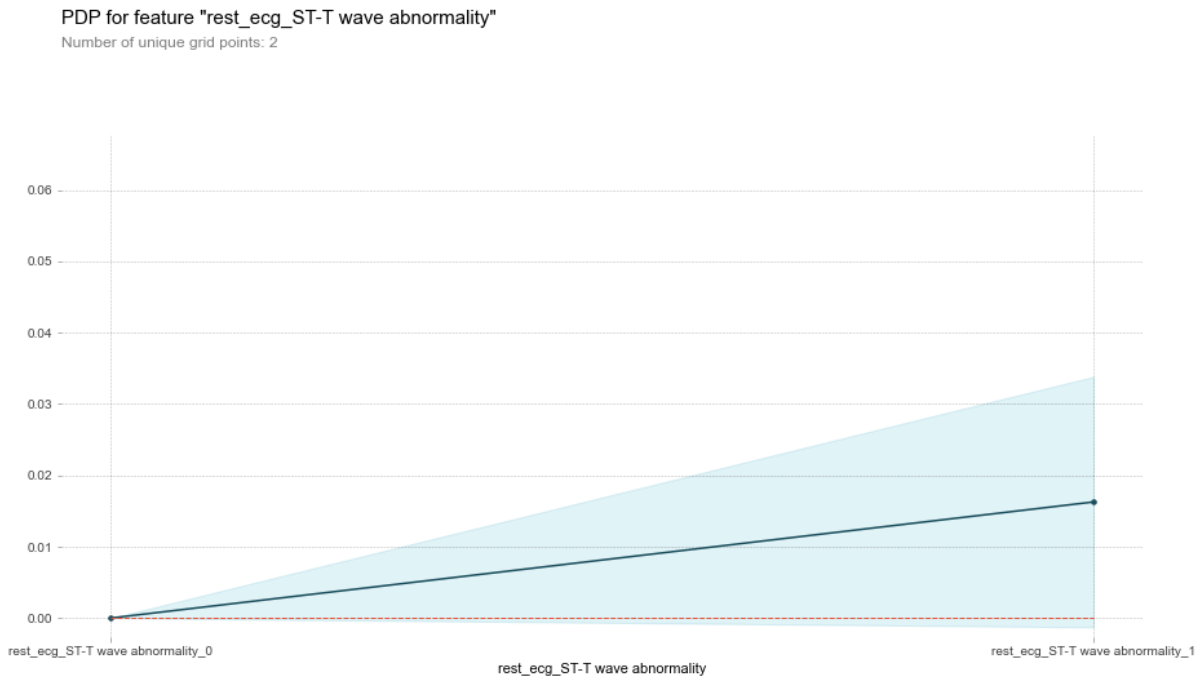

- **So, we can see when thalassemia_fixed defect increases, the probability of heart disease increases. That makes sense**




- **So, we can see when thalassemia_reversable defect increases, the probability of heart disease decreases. That makes sense**



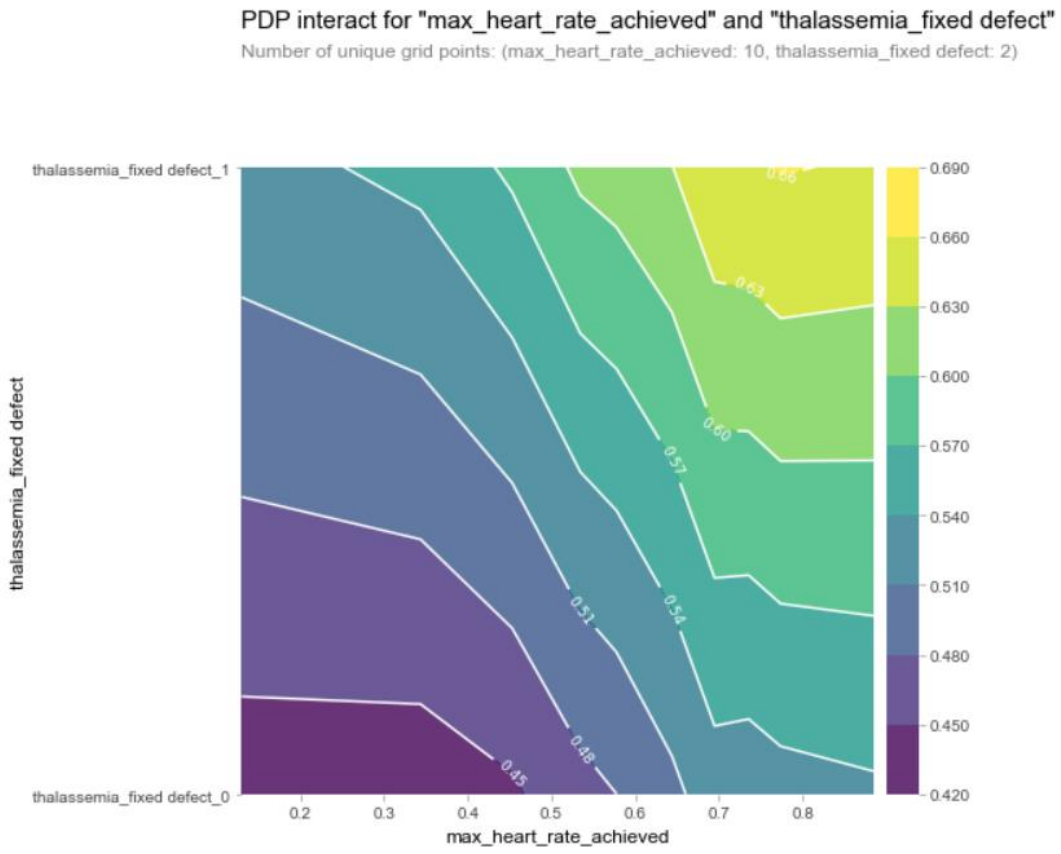

- **So, we can see when st_slope_downsloping increases, the probability of heart disease increases. That makes sense Upward or downward shifts can represent decreased blood flow to the heart from a variety of causes, including heart attack**




- **So, we can see when rest_ecg_ST-T wave abnormality increases, the probability of heart disease increases. That makes sense**

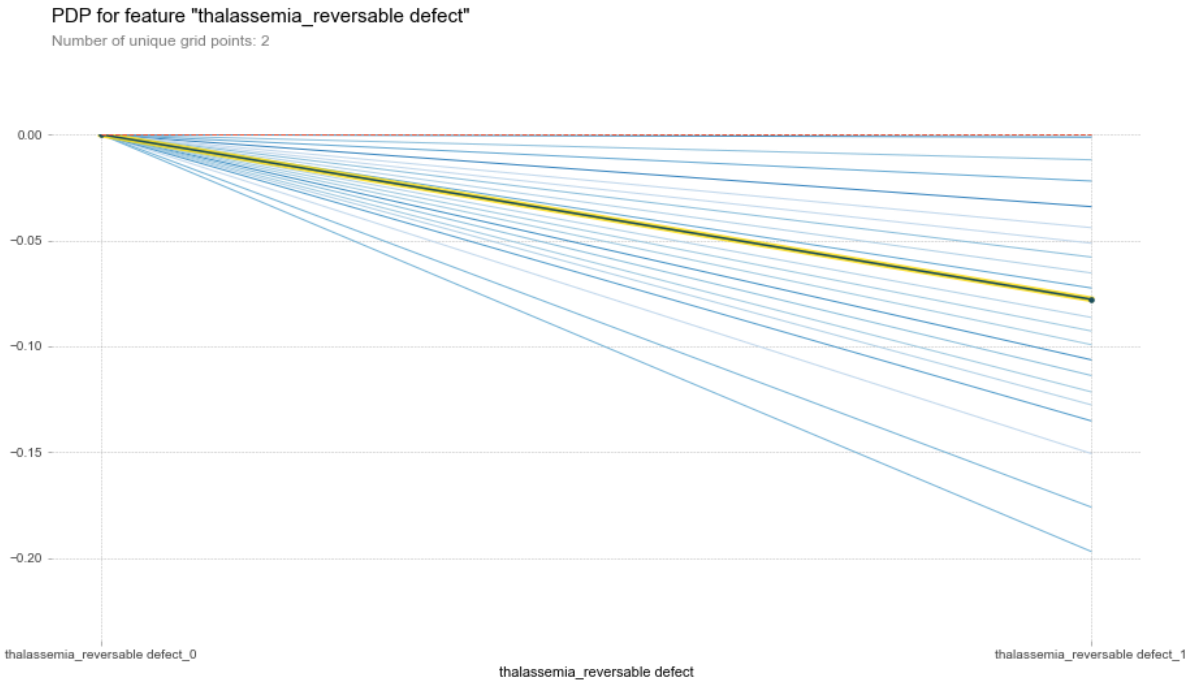
### 2D Partial Dependence Plots

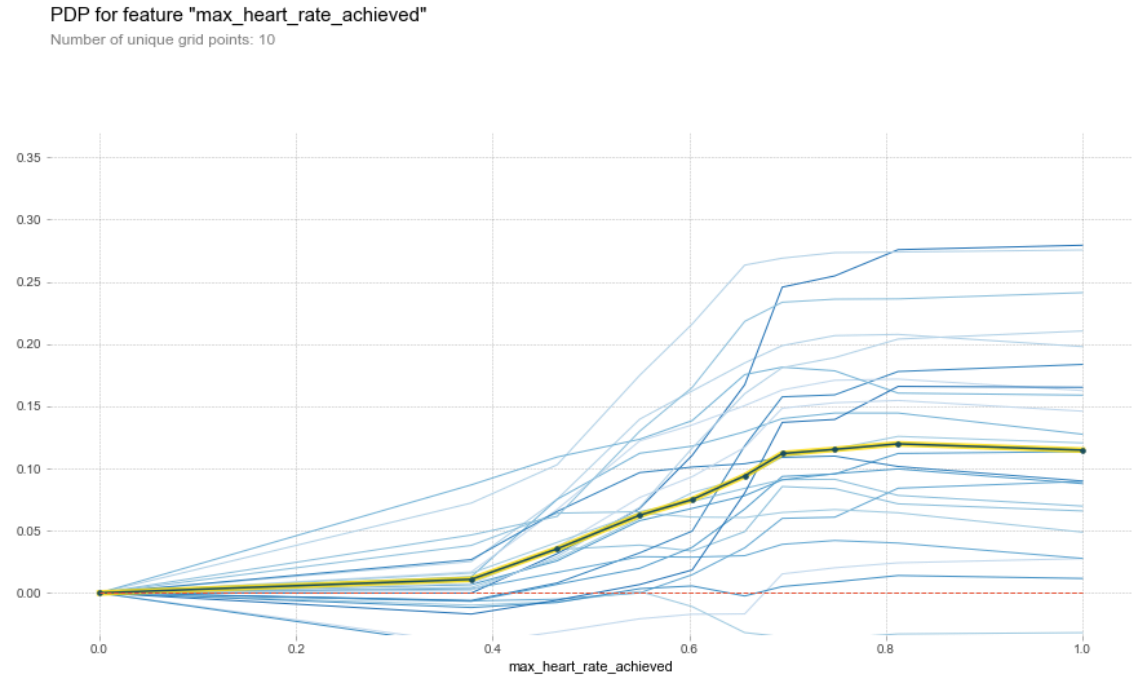
We can also visualize the partial dependence of two features at once using 2D Partial plots.




**we can see when max_heart_rate_achieved & thalassemia_fixed defect increases, the probability of heart disease increases.**

**ICE plots are similar to PD plots but offer a more detailed view about the behavior of near similar clusters around the PD plot average curve. ICE algorithm gives the user insight into the several variants of conditional relationships estimated by the black box.**



## Third Technique - SHAP Values

### Understanding individual predictions

Finally, we will take a look at the Shapley values for different features. Shapley values are an idea that comes from game theory. For an input vector x , to compute the Shapley value of feature i , we consider all the possible subset of features that don't include i , and see how that model prediction would change if we included i . We then average of all such possible subsets. There are many theoretical properties of Shaply values which make them attractive. In particular, they are the only measure of feature importance which satisfy the following four properties simultaneously (we state these informally).

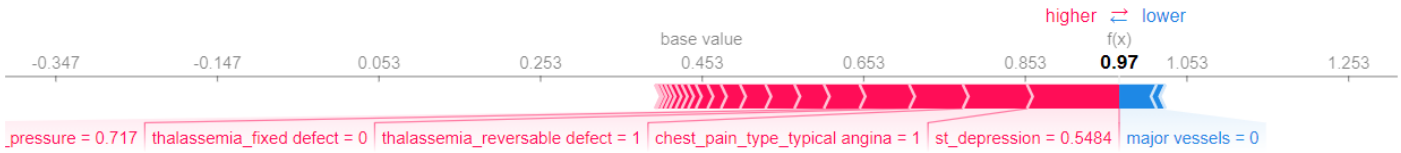
- Efficiency: The Shapley values of all features for a given prediction sum to the output of the model (i.e. the probability that target = 1).
- Symmetry: Any two features which have the same effect on the prediction are given the same Shapley value.
- Linearity: The Shapley value of a collection of features is the sum of the Shapley values of the features.
- Dummy: A feature which has no effect on the prediction has a Shapley value of 0.

We only compute the shap values for the random forest, and not logistic regression. There are a couple of reasons for this.

- TreeSHAP is a fast algorithm that comutes the exact Shapley values for the features and is unaffected by correlations in the data. For logistic regression we would have to use the KernelSHAP algorithm, which is much slower, is affected by correlations and only provides an approximation of the Shapley values.
- We will want to use Shapley values to plot dependency plots. Since logistic regression is a linear model, we already know that these plots will be linear.

***Shap values show how much a given feature changed our prediction (compared to if we made that prediction at some baseline value of that feature).***

- 20th record of test set :

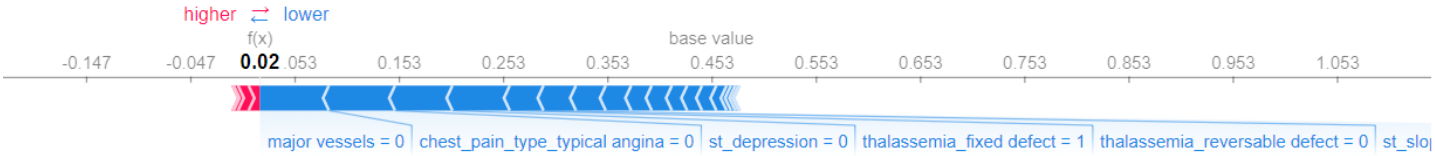



Interpretation

The above explanation shows features each contributing to pushing the model output from the base value (the average model output over the training dataset we passed) to the model output. Features pushing the prediction higher are shown in red, those pushing the prediction lower are in blue

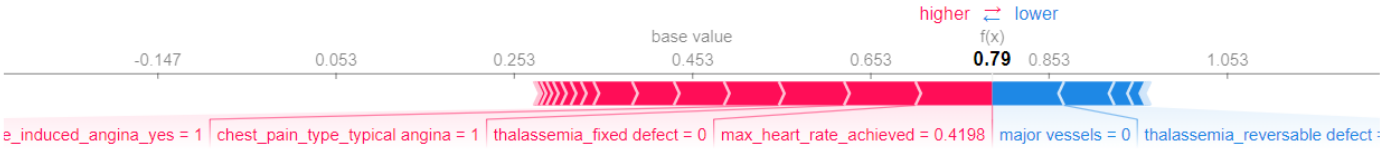
- The base_value here is 0.453 while our predicted value is 0.97.
- st_depression=0.5484 has the biggest impact on increasing the prediction, while
- major vessels=0 the feature has the biggest effect in decreasing the prediction.

- 22th record of test set :

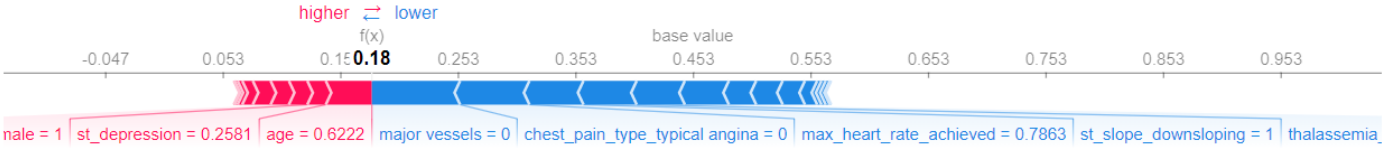



major vessels=0, chest_pain_type_typical angina=0 features has the biggest effect in decreasing the prediction.

- 23th record of test set :



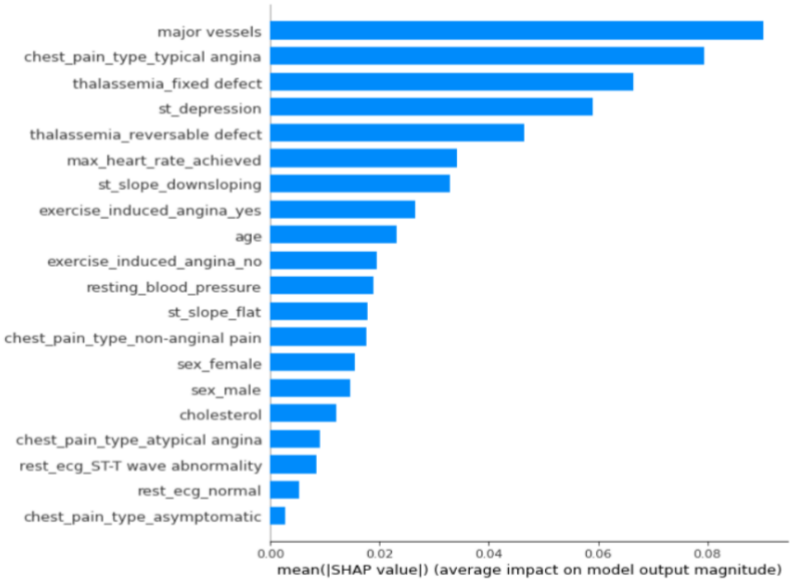

- 10th record of test set :




- **Feature values causing increased predictions are in pink, and their visual size shows the magnitude of the feature's effect.**
- **Feature values decreasing the prediction are in blue**

## SHAP Feature Importance Plot

The global mean(|Tree SHAP|) method applied to the heart disease prediction model. The x-axis is essentially the average magnitude change in model output when a feature is "hidden" from the model (for this model the output has log-odds units).

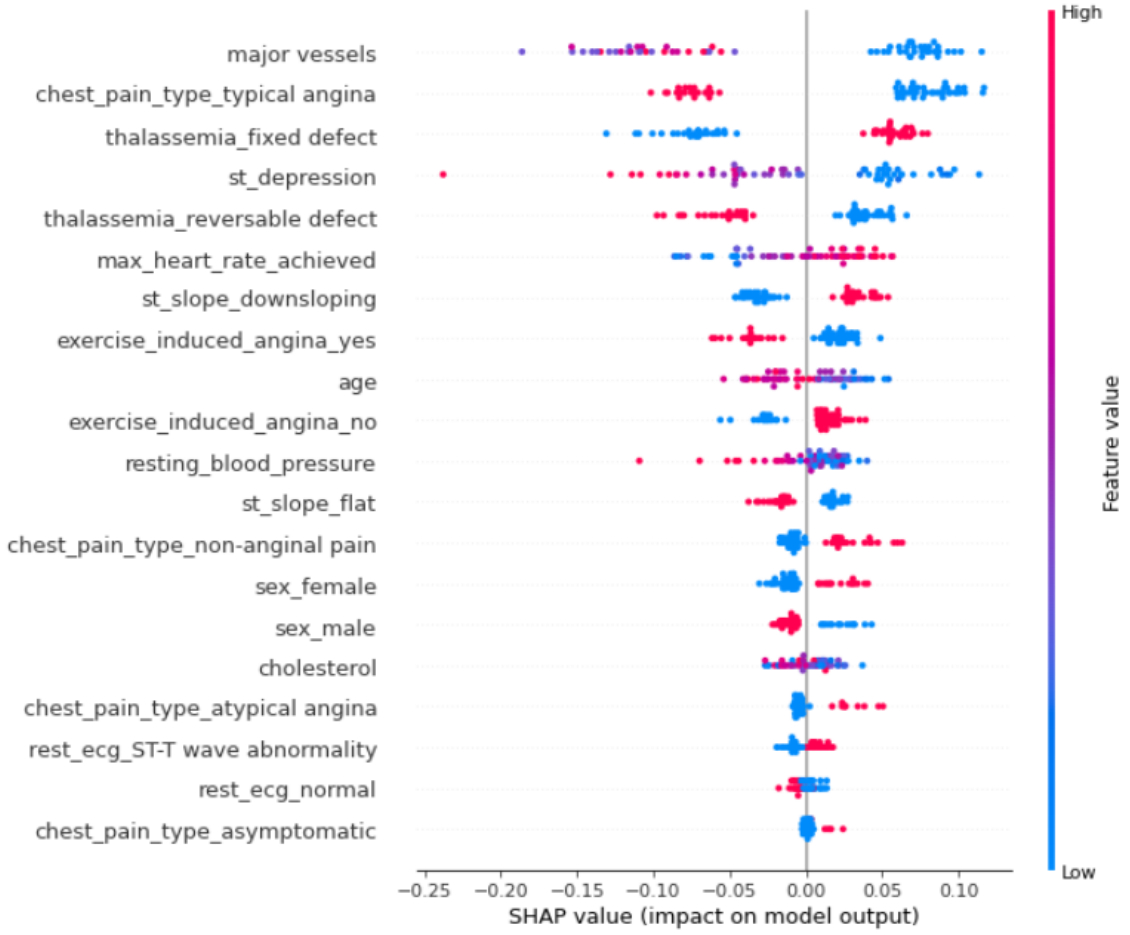


**SHAP Summary Plot**

To get an overview of which features are most important for a model we can plot the SHAP values of every feature for every sample. The summary plot tells which features are most important, and also their range of effects over the dataset.

For every dot:

- Vertical location shows what features it is depicting.
- The color shows whether that feature was high or low for that row of the dataset.
- Horizontal location shows whether the effect of that value caused a higher or lower prediction.

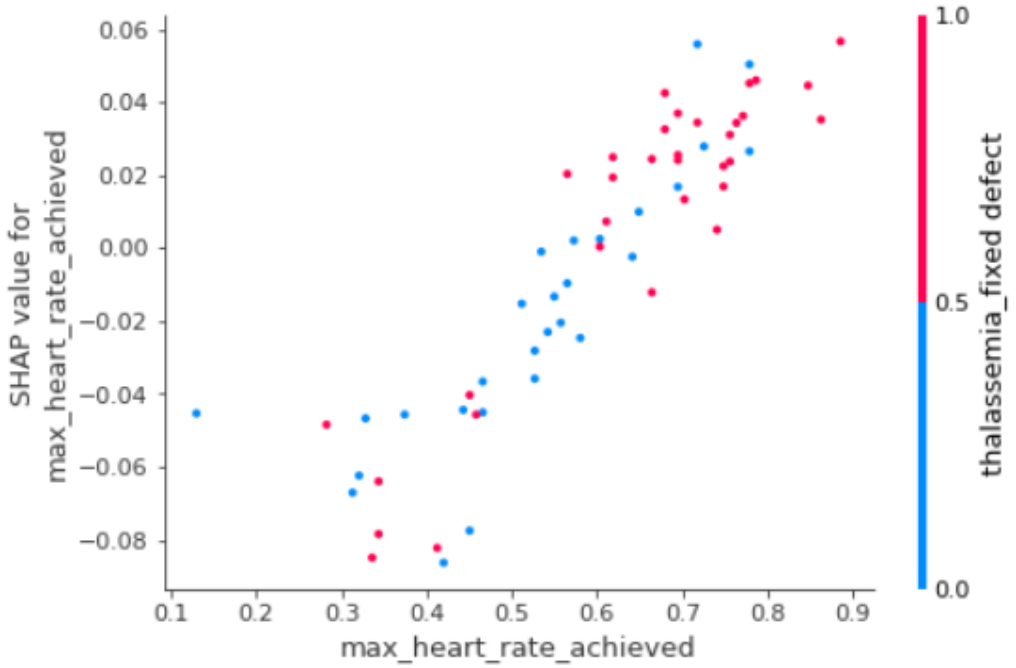



The higher the SHAP value of a feature, the higher is the log odds of heart disease in this heart disease prediction model. Every patient in the dataset is run through the model and a dot is created for each feature attribution value, so one patient gets one dot on each feature's line. Dot's are colored by the feature's value for that patient and pile up vertically to show density.

- **The number of major vessels division is pretty clear, and it's saying that low values are bad (blue on the right), the probability of heart disease increases.**
- **The number of chest_pain_type_typical angina division is pretty clear, and it's saying that low values are bad (blue on the right).**
- **Higher values of thalasemia_fixed_defect increases the risk of heart disease whereas its lower values decreases the chances of heart disease.**
- **The thalassemia 'reversable defect' division is very clear (yes = red = good, no = blue = bad).**
- **The thalassemia 'thalassemia_fixed defect' division is very clear (yes = red = bad, no = blue = good).**
- **The thalassemia 'st_slope_upsloping' division is very clear (yes = red = bad, no = blue = good).**

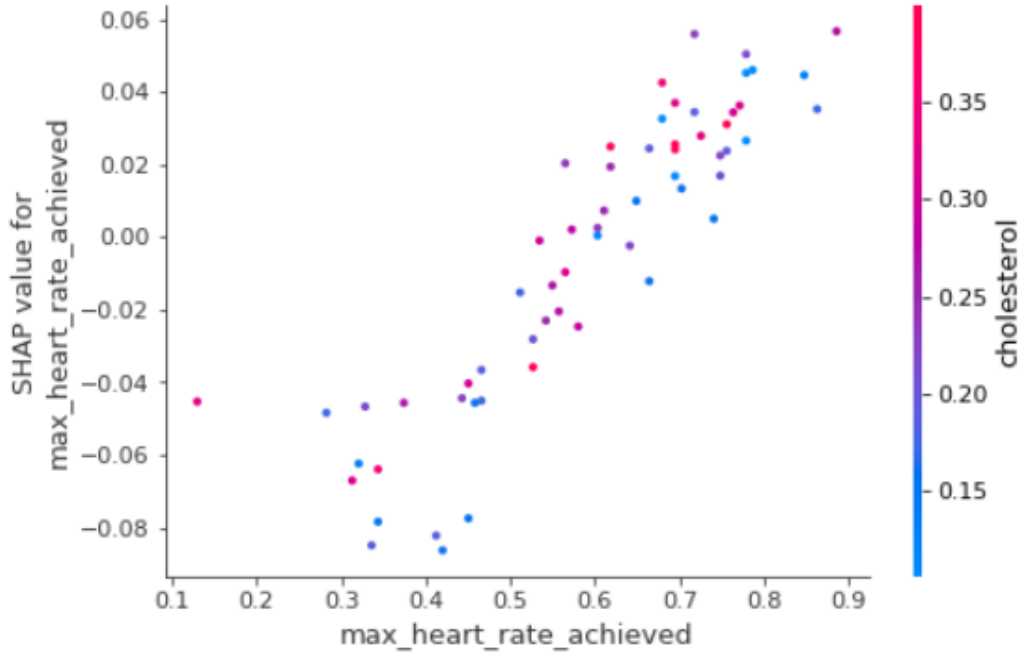
**SHAP Dependence Contribution Plots**

While a SHAP summary plot gives a general overview of each feature, a SHAP dependence plot shows how the model output varies by a feature value. SHAP dependence contribution plots provide a similar insight to PDPs, but they add a lot more detail.




Start by focusing on the shape, and we'll come back to color in a minute. Each dot represents a row of the data. The horizontal location is the actual value from the dataset, and the vertical location shows what having that value did to the prediction.

- **we can see when max_heart_rate_achieved & thalassemia_fixed defect increases, the probability of heart disease increases.**



**Fourth (Extra) technique - LIME(Local Interpretable Model-agnostic Explanations)**

Local surrogate models are interpretable models that are used to explain individual predictions of black box machine learning models. Surrogate models are trained to approximate the predictions of the underlying black box model. Instead of training a global surrogate model, LIME focuses on training local surrogate models to explain individual predictions.

The recipe for training local surrogate models:

1.Select your instance of interest for which you want to have an explanation of its black box prediction.

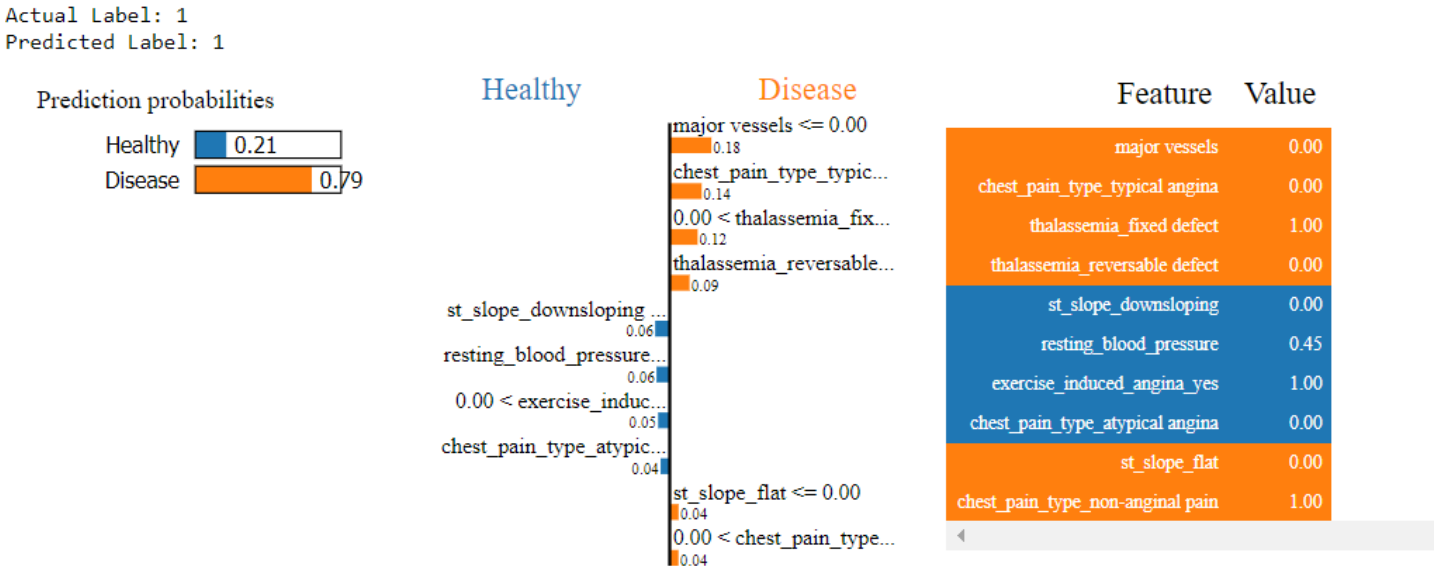
2.Perturb your dataset and get the black box predictions for these new points.

3.Weight the new samples according to their proximity to the instance of interest.

4.Train a weighted, interpretable model on the dataset with the variations.

5.Explain the prediction by interpreting the local model.

Interpretable Machine Learning book : https://christophm.github.io/interpretable-ml-book/lime.html




**Interpretability Conclusion**

Machine Learning doesn't have to be a black box anymore. What use is a good model if we cannot explain the results to others? Interpretability is as important as creating a model. To achieve wider acceptance among the population, it is crucial that Machine learning systems are able to provide satisfactory explanations for their decisions. As Albert Einstein said," If you can't explain it simply, you don't understand it well enough".

# 5. Machine Learning Interpretability for Fetal Health Dataset

Many of the techniques examined above and applied in Heart disease dataset were also applied in Fetal Health Dataset.

**Feature Importance**

Feature Importance was used in order to decide which features might have a greater impact on the prediction of fetal health than the others.

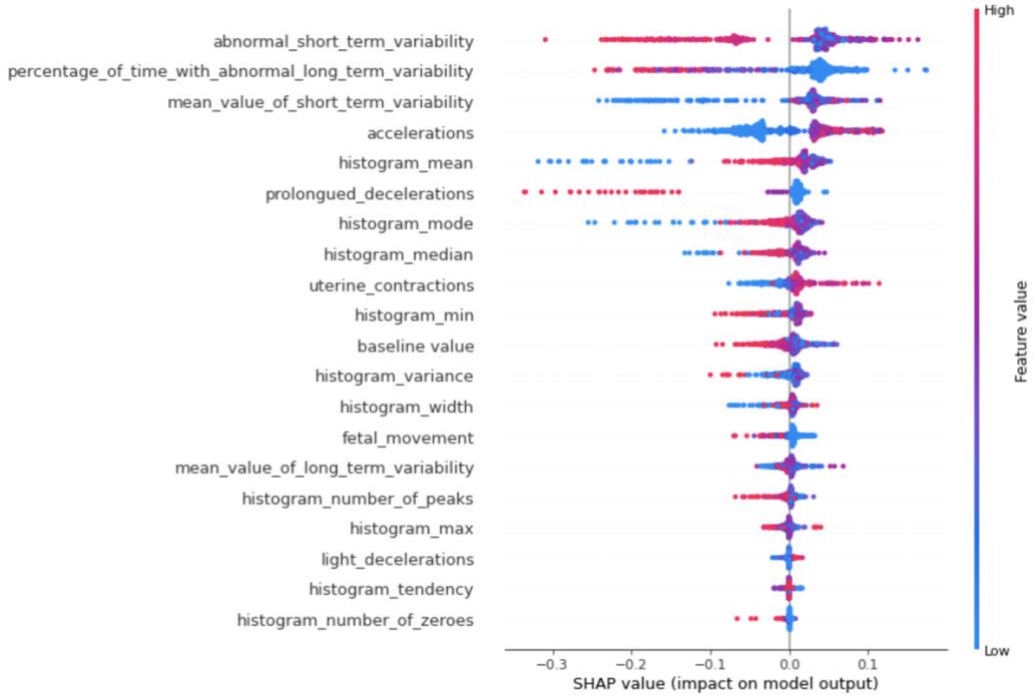
| Weight | Feature |
|---|---|
| 0.0504 ± 0.0186 | abnormal_short_term_variability |
| 0.0274 ± 0.0066 | percentage_of_time_with_abnormal_long_term_variability |
| 0.0241 ± 0.0087 | accelerations |
| 0.0226 ± 0.0063 | mean_value_of_short_term_variability |
| 0.0105 ± 0.0094 | histogram_mean |
| 0.0083 ± 0.0070 | uterine_contractions |
| 0.0083 ± 0.0038 | prolongued_decelerations |
| 0.0064 ± 0.0056 | histogram_median |
| 0.0045 ± 0.0038 | histogram_number_of_peaks |
| 0.0030 ± 0.0038 | light_decelerations |
| 0.0026 ± 0.0077 | histogram_variance |
| 0.0015 ± 0.0055 | baseline value |
| 0.0008 ± 0.0018 | fetal_movement |
| 0 ± 0.0000 | severe_decelerations |
| -0.0004 ± 0.0015 | histogram_number_of_zeroes |
| -0.0015 ± 0.0073 | histogram_max |
| -0.0019 ± 0.0000 | histogram_tendency |
| -0.0019 ± 0.0063 | histogram_min |
| -0.0038 ± 0.0063 | histogram_mode |
| -0.0038 ± 0.0034 | mean_value_of_long_term_variability |
| -0.0056 ± 0.0053 | histogram_width |

Here top 5 important features:

- abnormal_short_term_variability
- percentage_of_time_with_abnormal_long_term_variability
- accelerations
- mean_value_of_short_term_variability
- histogram_mean

## SHAP Summary Plot

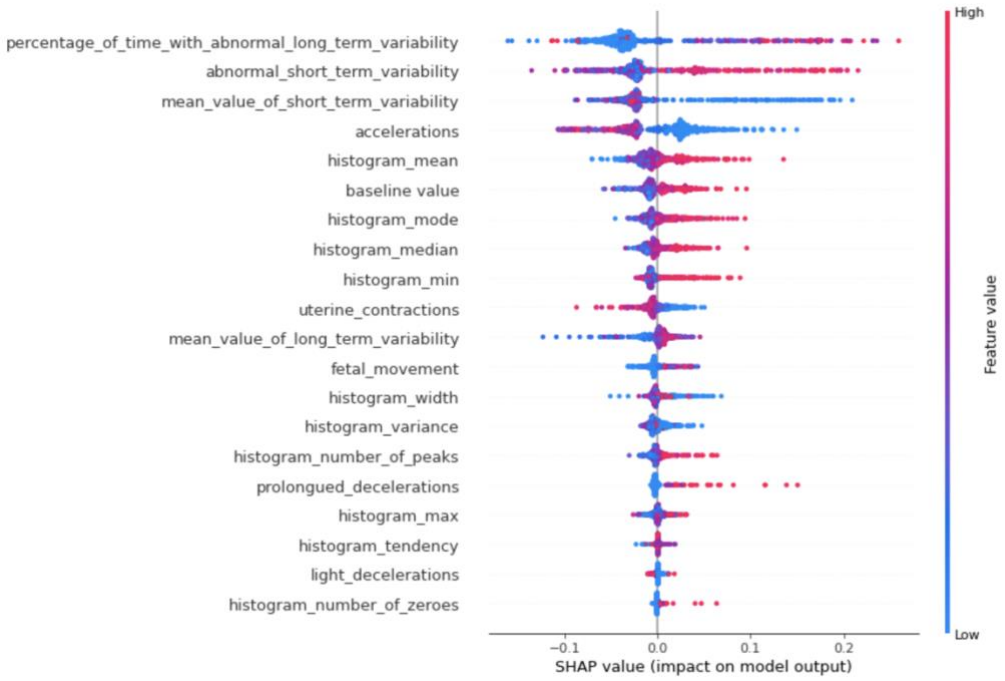
### 1. Class normal



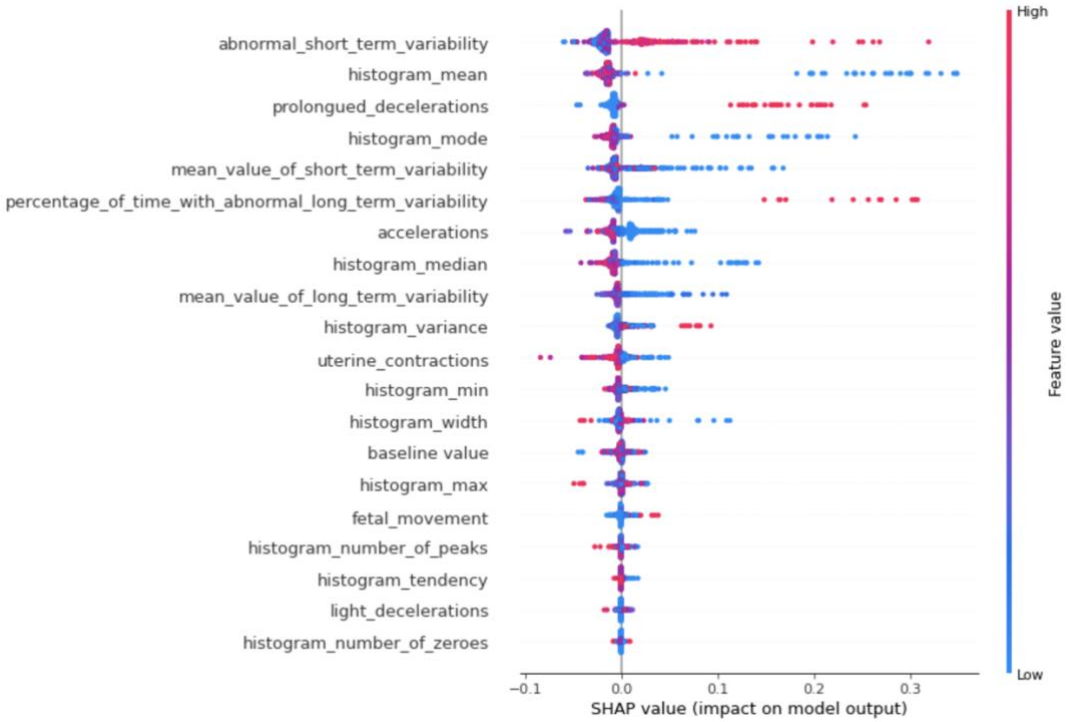

We could easily understand that

- low value of abnormal short term variability increases the chances of fetus not being healthy. (blue, right)
- low value of percentage_of_time_with_abnormal_long_term_variability means a pathological or suspect case (blue, right)
- high value of accelerations means no normal cases (red, right)

### 2. Class suspect



### 3. Class Pathological




## SHAP Feature Importance Plot

The x-axis is essentially the average magnitude change in model output when a feature is “hidden” from the model (for this model the output has log-odds units).