**Specific Aims**

Myocardial infarction is one of the dangerous diseases. The wide spread of this disease over the past half century has made it one of the most acute problems of modern medicine. The incidence of myocardial infarction (MI) remains high in all countries. This is especially true of the urban population of highly developed countries, exposed to the chronic effects of stress factors, irregular and not always balanced nutrition. In the United States annually, more than million people become ill with myocardial infarction.

Even though the introduction of modern treatment and prophylactic measures has somewhat reduced mortality from heart attacks, it continues to be quite high. Every year in the United States 200-300 thousand people die from acute myocardial infarction before arriving at the hospital. In the United States, every 29 seconds, one person becomes ill with MI, and every minute one patient with MI dies.

The course of the disease in patients with MI is different. MI can occur without complications or with complications that do not worsen the long-term prognosis. At the same time, about half of patients in the acute and subacute periods have complications leading to a worsening of the course of the disease and even death. Even an experienced specialist can not always foresee the development of these complications. In this regard, predicting the complications of myocardial infarction in order to timely carry out the necessary preventive measures is an important task.

My long-term goal is to develop a Prediction Model and use data science techniques to handle these sparse and high dimensional datasets. I aim to efficiently impute missing data and determine the most related positive feature and construct model to predict the complications accurately, the target feature in my project is Myocardial rupture and I going to build model base on different time interval.

* Aim 1 – To properly handle anomalies and missing data. In other words, clean the dataset so that I can effectively use it for interpolation in our second aim, I will use imputation or dropout features to deal with missing data depending on which performance is better for my model performance.
* Aim 2 – Determine the best model to predict the likelihood of a combined patient having Myocardial rupture in four different time intervals, I will compare the model performance of Kera’s Deep neural network, decision trees, random forests, and select the best performing one as last Model.

I expect that this project can accurately predict the probability of Myocardial rupture and the most critical factors for patients to develop Myocardial rupture. I can use the patient's case characteristics to predict the probability of Myocardial rupture, achieve early prevention, and reduce the patient's death probability due to late treatment.

**Approach**

* First, I will explore the dataset, load the data and applying basic Expooratory Data Analysis techniques, determine the number and types (numeric, categorical, ordinal) for each feature in the dataset, determine if there are any missing/null values for any features and do data processing with dropout or imputation, also encode features variables as appropriate - ordinal, one-hot, at the end do splitting data set for next step
* Second, I’ll spend time learning and applying different models in united framework, allowing us to compare and contrast different metrics, both visually and computationally. This process will take significant time but will give us a portfolio of estimation methods, I can refine this portfolio into a shortlist and determine with model performs best, the model includes but not limited by DNN and decision trees.
* Third, I will compare the performance of different models, after I select the best model. I will fine-tune the best-performing model to improve accuracy and evaluating performance on the test set, repeat this process until I get good model performance.

**Data**

1. the time of admission to hospital: all input columns (2-112) except 93, 94, 95, 100, 101, 102, 103, 104, 105

2. the end of the first day (24 hours after admission to the hospital): all input columns (2- 112) except 94, 95, 101, 102, 104, 105 can be used for prediction

3. the end of the second day (48 hours after admission to the hospital) all input columns (2- 112) except 95, 102, 105 can be used for prediction

4. the end of the third day (72 hours after admission to the hospital) all input columns (2- 112) can be used for prediction.

The target feature needs to be predicted is:

119. Myocardial rupture (RAZRIV)

some variable in the input columns are:

* Gender (SEX)
* Quantity of myocardial infarctions in the anamnesis (INF\_ANAM):
* Exertional angina pectoris in the anamnesis (STENOK\_AN):
* Functional class (FC) of angina pectoris in the last year (FK\_STENOK)
* Coronary heart disease (CHD) in recent weeks, days before admission to hospital
* Heredity on CHD (IBS\_NASL)
* Presence of an essential hypertension (GB)
* Symptomatic hypertension (SIM\_GIPERT)
* Duration of arterial hypertension (DLIT\_AG)
* Presence of chronic Heart failure (HF) in the anamnesis (ZSN\_A)
* Observing of arrhythmia in the anamnesis (nr11)
* Premature atrial contractions in the anamnesis (nr01)
* Premature ventricular contractions in the anamnesis (nr02)

**Methodologies aim 1**

* Sub aim 1 – Load data by Python module pandas from excel, determine Numeric features, Categorical features, Ordinal features and use Scikit-learn and NumPy library to encode categorical variables if there are any.
* Sub aim 2 – Implement imputation method or dropout irrelevant features to handle missing value in dataset, split data to training, validation, and test dataset.

**Methodologies aim 2**

* Sub aim 1 – make a plot by *matplotlib.pyplot* module to observe label distribution of target feature *RAZRIV*, Construct a Decision Tree model using Scikit-learns framework with default parameters. Determine if model experience overfitting by make a plot with accuracy metrics, if decision tree model is overfitting, apply k-folder split cross validation technique, if dataset is imbalanced, apply stratified k-fold to split data to multiple folders.
* Sub aim 2 – if overfitting is not resolved by step in sub aim 1, use feature selection technique to select most relevant features, Larger weights indicate that their corresponding features have more influence in the model with most Significant Positive Features which resulting factors represent the most relevant indicators of patient-acquired Myocardial rupture(RAZRIV).
* Sub amin 3 - Evaluate multiple models to measure performance. after sub aim 2, the top choice would be DNN, then use hype parameter turning technique to improve every model's performance base on metrics: accuracy, f1 score, precision and AUC, make a plot for each graph, determine the most important metrics for the model and set up cost function to get best result. in the end, evaluate the best model as result.

**Concern**

One potential problem is overfitting, which is very common in machine learning, the model fits the training data too well, but fails to generalize to new, unseen data, I may use multiple machine learning method to handle this problem.

Another potential problem is an imbalanced dataset. The negative and positive cases can be extremely imbalanced due to a lack of patient case records, which could make it challenging to handle missing data in a proper way and effect model performance.

**Timeline and Resources**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Weeks from Project Start (Nov. 15)** | | | | | | | | |
| **Project Task/Deliverable** | **11/15** | **11/21** | **11/22** | **11/28** | **11/29** | **12/5** | **12/6** | **12/12** |
| *Project Plain* |  |  |  |  |  |  |  |  |
| Aim 1 |  |  |  |  |  |  |  |  |
| Sub-aim 1 |  |  |  |  |  |  |  |  |
| Sub-aim 2 |  |  |  |  |  |  |  |  |
| Aim 2 |  |  |  |  |  |  |  |  |
| Sub-aim 1 |  |  |  |  |  |  |  |  |
| Sub-aim 2 |  |  |  |  |  |  |  |  |
| Sub-aim 3 |  |  |  |  |  |  |  |  |
| *Final Report* |  |  |  |  |  |  |  |  |
| *Final Presentation* |  |  |  |  |  |  |  |  |

**Resource**

I will use Anaconda software to build the environment and run the code locally, or Google Colab for training if GPU processing is slow in local environment.