Classification & Prediction of Acute Heart Disease

C964 TASK 2

Bilal Sayed

4/19/2021

Table of Contents

A1. Letter of Transmittal

April 15, 2021

Board of Directors

Parkland Memorial Hospital

5524 Main St

Dallas, TX 75546

Dear Board of Directors,

With the cost of healthcare in the United States rising, as well as an increased potential in the amount of patients at risk from heart disease due to the higher median age of the population, healthcare facilities like ours require an improved screening process for this serious condition. Normally, patients come in for yearly physicals, during which we collect blood samples and other data. This data then makes its way to one of our many doctors here on staff. The doctor then evaluates the data and makes a decision on the case, determining the patient's risk of heart disease. This works well but due to the increased screenings, our doctors are not able to keep up with the demand. Also, with a shortage of doctors around the country, hiring more doctors is not a cost effective strategy to combat this problem. We here in the IT department at Parkland Memorial Hospital have been made aware of the problem and have come up with a solution that will alleviate the load from the doctors and improve the efficiency and accuracy of the screenings.

We propose an application that, with the help of Artificial Intelligence (AI) technology, will be able to classify heart disease patients as well as predict at risk patients. AI is an up and coming technology that is being used more and more to help take strain off employees and allow them to focus on other aspects of their jobs. Our plan is to train an AI model to classify heart disease patients using heart disease and non-heart disease patients' data that we have collected over the years here at Parkland Memorial Hospital. Once the model is trained, we can then feed in other patients' data, and the model will be able to predict whether the patient is at risk for

heart disease. We hypothesize that our model will be at least 75% accurate, and the remaining 25% will be sent to doctors for review. We will eventually fine tune the model to a higher accuracy but that will take more time.

This project will not be a small undertaking. It will require at least 4 months of development and a budget of around $412,288 to $551,688. However, once you compare that to the cost of and time invested in hiring more doctors, it becomes apparent that this is the way to go. Also, based on the success of this project, we can apply this method to other screenings here at the hospital. Of course, changes like these raise questions about the impact on stakeholders. We strongly believe this will only create a positive outcome for all stakeholders here at the hospital. It will save money for the hospital in the coming years. The increased efficiency means more screenings, which in turn means more patients, which in turn creates more revenue for the hospital. And, overall, there will be an increase in the health and well-being of community members.

You may question the legality and ethicality of such a project. We will ensure the utmost care is taking when handling patients' data. Previous patients will be contacted and approval will be required to use their data for this project. Depending on the response, we may offer compensation for using their data. Because the data is considered health records, we will follow all HIPAA guidelines when handling patients' records.

It may seem strange that the IT department here at the hospital will be developing this application. You might think that it makes more sense to contract the project out to a development company. However, developing this in house makes the most sense for the business. Firstly, we know the business inside and out, so less time will be wasted at the beginning because we will not need introductions into anything. We will also keep the hospital's best interest in our mind when making decisions related to this project. And finally, we will own this product, which we can then lease out to other hospital, and add another revenue stream.

For this project, we will hire an additional two developers. We already have three senior developers on staff, two of whom will be assigned to this project. Together, they will develop this product with me as the project manager. I have worked in the IT healthcare industry for 27

years, and have extensive experience managing projects like these. This project has a high chance of success, and will propel our hospital into the future of healthcare technology.

Sincerely,

Bilal Sayed

Senior Project Manager

Parkland Memorial Hospital

A.2 Project Recommendations

A.2.1 Problem Summary

The problem here at the hospital is simple; doctors don't have enough time to diagnose each individual case of heart disease. The median age of the population is increasing, and there is more and more of a demand for heart disease screenings. In order to keep up with the increased demand for heart disease screenings, a cost effective solution must be put in place to increase the throughput and efficiency of the screenings. While hiring doctors might seem like an easy solution to the problem, it comes with added costs that can be overcome with an AI application. An AI application has a high upfront cost, but will outperform a doctor and pay for itself for years to come.

A.2.2 Application Benefits

The proposed application would provide many benefits to the business. Firstly, it will greatly increase our ability to process heart disease screenings. This application will take the burden of heart disease diagnostics from doctors, allowing them to be free to do other important work and automate the process. The application will greatly increase the number of tests the hospital can handle per day, increasing our screening process efficiency and capability. Patients can expect same day or next day results, instead of weeklong waits for diagnostic results. Because the application will be developed by the hospital, we will be the owner of the product. Depending on the success, we can later on lease rights to other hospitals, diversifying our revenue streams.

A.2.3 Application Benefits

The application will be written in the Python language. This is a common language used in AI applications and will give us an advantage when developing the application. We will use Jupyter Notebook to develop the AI model. Jupyter Notebook will aid us in using Python to its fullest potential when developing the application. It will serve as an approximation to a server and will allow us to execute the Python model we will design. The historic data will be fed into the Jupyter Notebook where we will use the scikit-learn machine learning package to have the computer learn to classify heart disease. These 3 components will form the main backend of the application. The frontend will be

A.2.4 Data Description

To train an AI, data is required. We will, with individual approval, use hospital patients' historic data to train the AI. The data will be a comma-separated value file (.csv) with a header and 14 columns. Within the 14 columns will be the routine data collected when screening for heart disease. These include age, sex, chest pain, resting blood pressure, cholesterol, fasting blood sugar, resting electrocardiogram (ekg), maximum heart rate, exercise induced angina, ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment, number of major vessels colored by fluoroscopy, thalassemia, and acute heart disease result. These data points along with the resulting diagnosis will allow the AI to learn what conditions result in a heart disease diagnosis. A model will be created for future use in heart disease prediction using this dataset.

A.2.5 Objective and Hypothesis

The objective of this application is to take patients' data and predict an outcome based on the data, in this case, heart disease. To achieve this objective, we hypothesize that our AI model will have a 75% true positive diagnosis. This will allow the doctors workload to be reduced by 75%. The remaining 25% will be flagged for manual review by a human doctor. While 75% accuracy is our initial goal, we will set a future accuracy

goal and work to get the model as accurate as possible, and lower the doctors' workload by as much as possible. Depending on what search algorithms are used and other factors, our final goal is 85% accuracy.

A.2.6 Methodology

Every project needs a development methodology to ensure success. We will use, as we have in the past, the Agile methodology for the development of this application. The Agile method is designed around iterative steps, allowing for flexibility, testing and change through the development of the project. This process will allow us to quickly get an application out there for the nurses to start using. We can then, based on their feedback, improve and shape the application to better suit their needs. All development work will be completed during 4 week sprints. Between each sprint, we will, as a team, meet with each other and discuss what has been developed, what the customer requires, and the plan for the next 4 week sprint. This constant feedback loop will allow the product to suit the needs of customer, reducing the need of custom development later on down the road. We expect the initial development of the application to take 4 months to complete. With these 4 months, we will have QAs and developers working closely with the nurses and doctors to ensure the application suits their needs. After the 4 months, the application will be deployed and there will be a 1 month testing phase. This 1 month phase is there to ensure that nothing is forgotten and the customer is satisfied with the product. The application will then move into a maintenance phase, where we will support the application as necessary for the lifetime of the application.

A.2.7 Funding Requirements

The purpose of developing this application is because it is more cost effective than hiring more doctors. However, it still comes with certain funding requirements for which there will need to be a budget. The language, Python, development environment, Jupyter Notebook, and packages such as scikit-learn are all either free and/or open source. This will save us money in the software aspect of the development. The cost associated is mainly in the development of the software. We will have 4 developers

working for 4 months on the project. With the average hourly salary for a developer in this range of development being $125-$175 an hour, and an average of 21 working days a month, we can expect a cost between $336,000 and $470,400 for 4 months of development. We can also budget an additional $40,000-$45,000 for the project manager. QA hourly rates will be $27 an hour, and we will have 2 working on the team. That adds an additional $36,288 to the budget of the project. The total cost of the project is expected to be around $412,288 to $551,688.

A.2.8 Stakeholders Impact

Our stakeholders here at the hospital should fully support this project, as it only adds value to the hospital. This project will help lead the hospital towards the future. The project can be marketed to the public and create an image of technological advancement for the hospital. Additionally, this project will create an application for the hospital, which can be licensed out to other hospitals. The success of this project will allow for other screenings to also be automated, and will create more applications owned by the hospital. This project will be an asset for the hospital, and the stakeholders alike.

A.2.9 Ethical and Legal Considerations

Because this project deals with patient data, strict ethical and legal considerations must be taken into account. First, patient approval will be acquired and stored on file before it is used. We may decide to compensate patients' for their data, depending on the response. The data will be anonymous and will be stored in encrypted format on protected servers. We will be sure to follow all HIPAA regulations when storing and transmitting this data. If the application is licensed out to other hospital, it will not be necessary to ship it with the initial data, therefore protecting the patients' data. We will also have in place policies regarding data and procedures that must be followed. If anyone breaks the procedures, they may face termination. Anyone who interacts with the data will be screened and have a background check done to ensure that they are an honest and trustworthy person. The exact policies will be decided at a later date, when this

project is approved. However, you can be sure that patient privacy is a high priority for us, and there will be no exceptions.

A.2.10 Developer's Expertise

The team working on this project will consist of 4 developers and 2 quality analysts working under one project manager. We currently have 2 developers on staff with 15 years of experience between them. We will have the 2 developer on staff as senior developers, and hire 2 additional developers, with junior to mid-level experience. We will look to interview candidates with Python experience as well as some kind of machine learning. The timeline for this project will be tight with not too much room for time spent learning, so having people who know what they are doing is a big concern for us. We have QA already on staff, and they will be used as a resource in this project. I will serve as the project manager, with 27 years of experience working in this field. Together, this team will be able to develop, test, deploy, and maintain this project in a timely manner.

B. Project Proposal

B.1 Problem Statement

Our hospital's heart disease screening system is backed up, and not in a good way. Our doctors are not able to keep up with the rise in demand for screenings. While management may see an easy fix to this problem, hiring more doctors, we as IT professionals may see a different opportunity. To recognize the opportunity, you must first ask, how do doctors make the screening diagnosis? What is the result based on? It is simple, data. Patient data is collected, and then analyzed to make a decision. You might have recognized the opportunity now. It is artificial intelligence (AI). AI is very good at making predictions and categorizations based on large data sets. It is here when this technology and our problem intersect, creating an opportunity for us to introduce AI into the hospital industry.

With AI in the medical field being a controversial topic, we must create a solution that offers more benefit than risk. This necessitates creating an application that is not only cost effective, but builds trust in AI being used in this manner. A crucial aspect of building that trust is guaranteeing the accuracy of the result. Using our expertise as IT professionals to build trust in this application means using the best tools we have available. We will use Python, one of the most suitable languages for machine learning programs along with Jupyter Notebook, as well as third party packages. Using these tools will allow us to put our best effort forward in creating an exemplary application.

B.2 Customer Summary

The customer described here is us, the hospital. Because we will be developing this application in house, we will save time that would have originally been lost to an outside company familiarizing themselves with the inner workings of our hospital. Because the IT team is already embedded within the hospital, there will be no doubt that the hospital's best interests are always put first.

The two main benefits of this application for the hospital are first, greater efficiency in diagnosing heart disease, and second, the creation of an asset for the hospital in the form of a licensable application. This application will deliver efficiency by reducing the amount of time doctors need to spend on each patient's heart disease screening which both reduces the amount of time a patient needs to get a diagnosis as well as frees up the doctors' time to engage in other tasks. The development of this application will also create an asset for the hospital by because its success rates will create a demand in competitor hospitals for a similar diagnostic tool. Therefore, we can license out our already existing tool and generate revenue.

B.3 Existing Systems Integration

This application will use free and open source software and programming languages. It will not require replacing or modifying any existing infrastructure. The data will be exported from the hospital databases and loaded into our test environments.

B.4 Data

The data will be in the form of a comma-separated value file (.csv), and will be acquired from the hospital's databases (http://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/processed.cleveland.data). The dataset comprises 14 columns: age, sex, chest pain, resting blood pressure, cholesterol, fasting blood sugar, resting electrocardiogram (ekg), maximum heart rate, exercise induced angina, ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment, number of major vessels colored by fluoroscopy, thalassemia, and acute heart disease result.

Before using any of this data to train the model, we will first clean the data using a set of procedures. We will first search the data for null values and update them accordingly. Next, we will search the columns that require numbers for any non-number values and update them to a number. Then, we will search the dataset for missing values, and fill them in fittingly. These processes will ensure our data to be free from defects, creating a more accurate model.

B.5 Project Methodology

This project will be developed using the Agile software development methodology. The continuous cycle of development and feedback guarantees the best application is created for the hospital. The process includes different phases listed below.

Initiation/Inception:

- Discuss the project vision with the customer
- Identify team members who will work on the project
- Take stock of time and resources required to determine feasibility of the project

Planning:

- Get together with customer and stakeholders and get requirements
- Meet with users and record what they want from the product and why
- Estimate risk involved in developing this project
- Develop milestones to measure progress
- Create a backlog to follow during development

Development:

- Start the first sprint with the intent of having the first iteration of the product working and usable at the end
- We will use 4 week sprints to develop each iteration of the product
- Constant testing will be carried out by QA
- Continue iterating and testing until customer is satisfied

Deployment/Production

- Deliver the product to the customer
- Monitor closely for missed bugs or defects
- Train users to use the product
- Continue to monitor feedback and create additional iterations

Retirement

- Removal of product from production
- Usually because of an update or move to another solution

B.6 Project Outcomes

The project deliverable is a working model that will classify and predict heart disease in patients with accuracy.

B.7 Implementation Plan

Our implementation plan is a 2 part plan. We will start by releasing the first iteration as a beta to a select group of users. These users will work closely with the development team, doing usability testing and providing feedback on each iteration of the product. Using the feedback, the developers will develop the next iteration during the next sprint. After the final iteration is ready, the production version will be released. This will be the second part of the implementation. It will include training for all users, with live training, videos, and a wiki. The beta testers will serve as trainers because they will be familiar with the application. The software will then move into a production phase, and future updates will be pushed to the application remotely. Change notes will be emailed to each user.

B.8 Evaluation Plan

The evaluation plan will verify that our developed application meets the requirements specified by the customer in the planning phase. We will take a set of case studies and feed the data to the model while also giving doctors the same data. We will then compare the results and assess the accuracy level of the model to the accuracy level of the doctors. This will also help verify our hypothesis. If the accuracy level is below the approved limit, we will develop the model further to reach the accuracy goal.

We will also collect user data from the users using the application. This will help us evaluate whether the application is meeting their needs. Another way to evaluate the same factor is to send out surveys to the users. This feedback will provide us with an evaluation of the application and whether it meets the requirements laid out in the planning phase.

Also included in our evaluation plan is testing by QAs and beta testers. They will work to find as many bugs and errors within the program, which we will work to fix. When the application reaches a certain threshold, we will declare it ready for release.

B.9 Resources and Costs

The programming languages used:

- Phython 3.9
- Packages:
    - Scikit Learn
    - Numpy
    - Pandas
    - Matplot
    - Seaborn
    - Xgboost

These are free opens source packages and require no budget.

The programming environments used:

- Jupyter Notebook
- We have existing computers, network, and office space

` The resources needed:

- 1 project manager
    - $40,000-$45,000 for 5 months of work
- 4 developers
    - $125-$175 an hour
    - 672 hours of development
    - $336,000 and $470,400

- 2 QA
    - $27 an hour
    - 226 hours of QA
    - $36,288

The project is estimated to take 4 months. The programming language and environments are free. The human resources require are the bulk of the expense, requiring $412,288 to $551,688. There may be additional expense along the project, which is why we will go with the higher number, and request a budget of $551,688 for this project.

B.10 Timeline and Milestones

Event	Start Date	End Date	Duration	Dependencies	Resources Assigned
Project Start	5/3/2021	5/7/2021	5	N/A	Project Manager, Stakeholders
Sprint 1	5/10/21	6/7/21	20	Project Start	Project Manager, Developers, QA
Sprint 2	6/7/21	7/5/21	20	Sprint 1	Project Manager, Developers, QA
Sprint 3	7/5/21	8/2/21	20	Sprint 2	Project Manager, Developers, QA
Sprint 4	8/2/21	8/30/21	20	Sprint 3	Project Manager, Developers, QA
Deployment	8/30/21	-	-	Sprint 4	Project Manager, Developers, QA, End Users

D. Developed Product Documentation

D.1 Business Requirements and Project Purpose

The purpose of this project is to help relieve our hospital's over flooded screening system. With a rise in the demand for heart disease screenings, we look to the future to help us solve this problem. We will develop and implement an AI application that will accurately diagnose heart disease.

The requirements for this project are a working, fully functional heart disease diagnostic program. This will be developed using machine learning using past patients' data.

D.2 Raw and Cleaned Data

The data will be provided from the hospitals databases (http://archive.ics.uci.edu/ml/machine-learning-databases/heart-disease/processed.cleveland.data), and will be in comma-separated value file (.csv) format. This data will represent the raw data used in the creation of the model. Because the data is from the hospital databases, we can assume it to be already fairly clean data. However, we will still clean the data to ensure there are no problems later on.

We will import the data to our Jupyter Notebook environment:

```
heart=pd.read_csv("processed.cleveland.csv")
```

Next, we will use the .info() command to see information about the data:

```
heart.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       303 non-null    int64
 1   sex       303 non-null    int64
 2   cp        303 non-null    int64
 3   trestbps  303 non-null    int64
 4   chol      303 non-null    int64
 5   fbs       303 non-null    int64
 6   restecg   303 non-null    int64
 7   thalach   303 non-null    int64
 8   exang     303 non-null    int64
 9   oldpeak   303 non-null    float64
 10  slope     303 non-null    int64
 11  ca        299 non-null    float64
 12  thal      301 non-null    float64
 13  num       303 non-null    int64
dtypes: float64(3), int64(11)
memory usage: 33.3 KB
```

We can now see that there are missing values in the ca and thal columns:

```
heart[[i for i in heart.columns if heart[i].isnull().sum()>0]].isnull().sum()

ca      4
thal    2
dtype: int64
```

We will fill these values by doing the following:

```
heart.ca.value_counts()

0.0    176
1.0     65
2.0     38
3.0     20
Name: ca, dtype: int64
```

```
heart.ca = heart.ca.fillna(0.0)
heart.ca.value_counts()

0.0    180
1.0     65
2.0     38
3.0     20
Name: ca, dtype: int64
```

```
heart.thal.value_counts()

3.0    166
7.0    117
6.0     18
Name: thal, dtype: int64
```

```
heart.thal = heart.thal.fillna(3)
heart.thal.value_counts()

3.0    168
7.0    117
6.0     18
Name: thal, dtype: int64
```

If we use the .info() command again, we can see that all values are filled:

```
heart.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       303 non-null    int64
 1   sex       303 non-null    int64
 2   cp        303 non-null    int64
 3   trestbps  303 non-null    int64
 4   chol      303 non-null    int64
 5   fbs       303 non-null    int64
 6   restecg   303 non-null    int64
 7   thalach   303 non-null    int64
 8   exang     303 non-null    int64
 9   oldpeak   303 non-null    float64
 10  slope     303 non-null    int64
 11  ca        303 non-null    float64
 12  thal      303 non-null    float64
 13  num       303 non-null    int64
dtypes: float64(3), int64(11)
memory usage: 33.3 KB
```

D.3 Code Analysis

The application is made up of 2 components; the frontend GUI which is coded in python using Tkinter, and the AI model, also coded in python using scikit and other packages.

We first code the AI model in Jupyter Notebook. To do this, first the data is imported. Next, the data is cleaned. After the data is cleaned, we find the columns required based on mutual regression:

```
mutual_info = mutual_info_regression(X_train, y_train)
mutual_info = pd.Series(mutual_info)
mutual_info.index = X_train.columns
mutual_info = mutual_info.sort_values(ascending=False)
mutual_info
```

```
thal        0.154531
ca          0.148793
oldpeak     0.135505
cp          0.128294
fbs         0.089178
exang       0.086850
thalach     0.057735
slope       0.030023
restecg     0.029304
age         0.009784
sex         0.000000
trestbps    0.000000
chol        0.000000
dtype: float64
```

```
Req_Columns = list(mutual_info[mutual_info>0].index)
Req_Columns
```

```
['thal',
 'ca',
 'oldpeak',
 'cp',
 'fbs',
 'exang',
 'thalach',
 'slope',
 'restecg',
 'age']
```

We then train the model using these columns:

```
X_train_final = X_train[Req_Columns]
```

We also test the model with the required columns:

```
X_test_final = X_test[Req_Columns]
```

Finally, using a logistic regression algorithm, we train and test our model:

```
np.random.seed(42)
LR_model=LogisticRegression()
LR_model.fit(X_train_final,y_train)
LR_model_y_pred=LR_model.predict(X_test_final)
LR_model_r2_score=round(r2_score(y_test,LR_model_y_pred)*100,2)
print("Accuracy on Training set: ",round(LR_model.score(X_train_final,y_train)*100,2))
LR_model_score = round(LR_model.score(X_test_final,y_test)*100,2)
print("Accuracy on Testing set: ",LR_model_score)
```

```
Accuracy on Training set:  62.4
Accuracy on Testing set:  65.57
```

We then use joblib to export the model to use in our application:

```
import joblib
joblib.dump(LR_model, 'model.joblib')

['model.joblib']
```
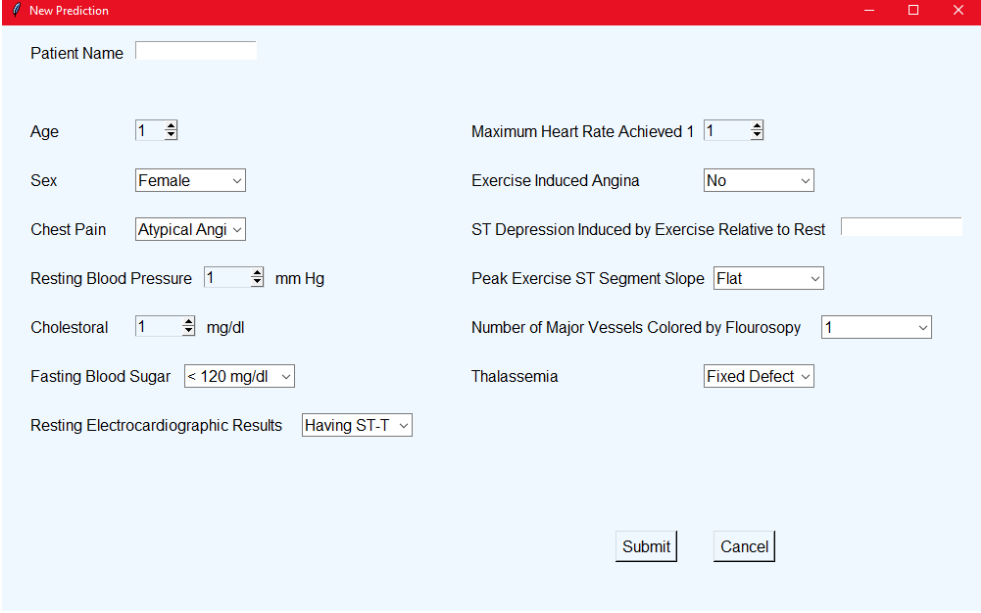
Above are the steps used in creating the basis for the application, the AI model. The patient data will be fed into this model and it will output a prediction. To do this, an application must be created around the model. We developed the application using Pycharm.

The application launches to a dashboard with 3 graphs of information about the training data:
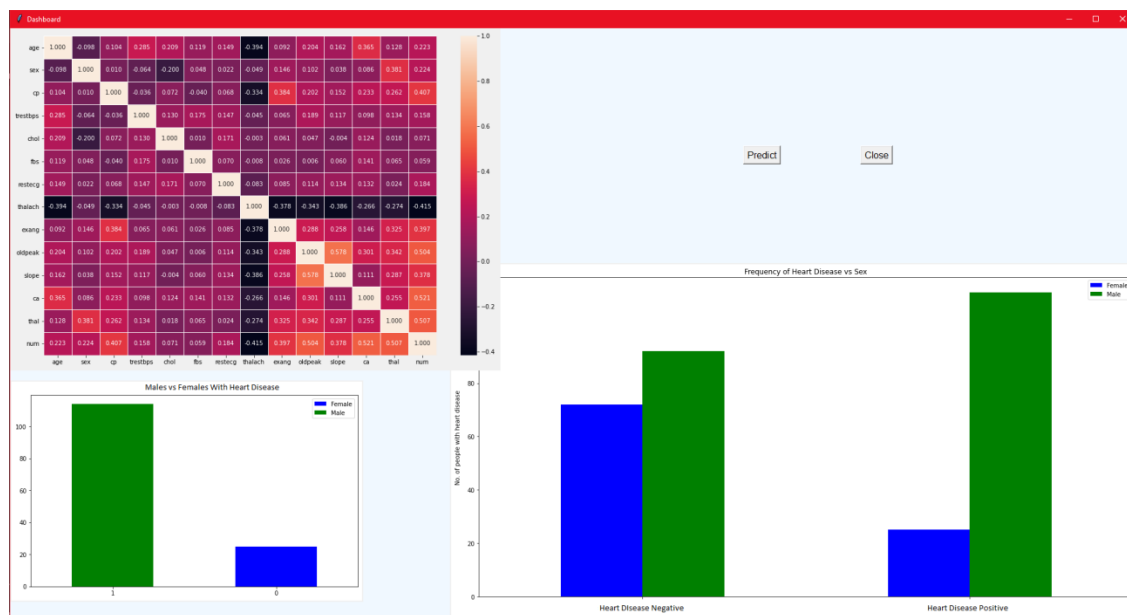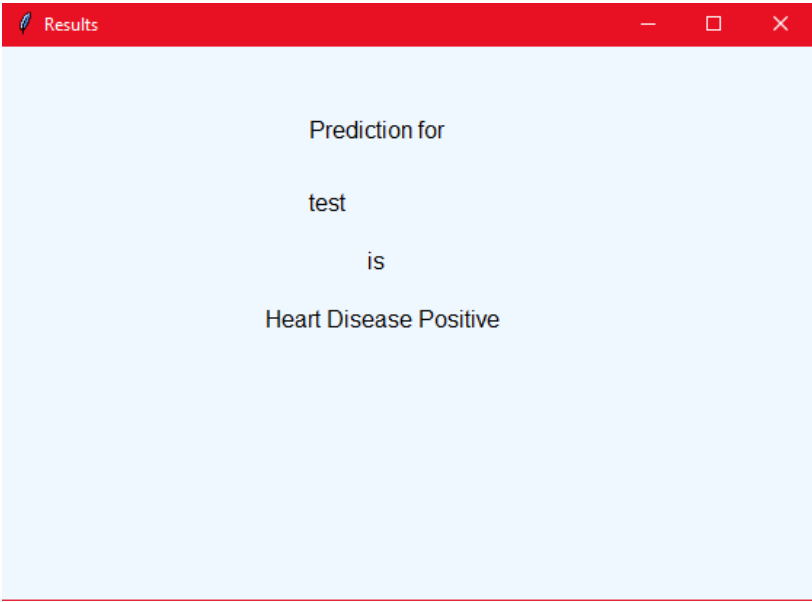
When the predict button is clicked, a second screen opens, allowing the user to enter data for a prediction:

On this screen, nurses and doctors will enter the patients' name and data. Next they will click the submit button. A window will pop up with the models prediction:

This window will contain the patient's name and outcome. This concludes the application.

D.4 Hypothesis Verification

In order for our application to provide an alternative to doctors, it must be accurate and precise. We hypothesized that our model would have a 75% true diagnosis rate. Using scikit learn functions, we can determine the accuracy of our model:

```
LR_model_r2_score=round(r2_score(y_test,LR_model_y_pred)*100,2)
print("Accuracy on Training set: ",round(LR_model.score(X_train_final,y_train)*100,2))
LR_model_score = round(LR_model.score(X_test_final,y_test)*100,2)
print("Accuracy on Testing set: ",LR_model_score)

Accuracy on Training set:  62.4
Accuracy on Testing set:  65.57
```

From these results, we can see our model has a accuracy of 65.57 percent. This is not close enough to our expected 75% accuracy to satisfy our hypothesis. However, this can be improved upon in the future, seeing as this is only a prototype.

D.5 Effective Visualizations and Reporting

To explore the data, visual aids are of great help. We used Jupyter Notebook to help us in understanding the data better.

Using the describe() method, we can see attributes of the data:

```
heart.describe()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	299.000000	301.0
mean	54.438944	0.679868	3.158416	131.689769	246.693069	0.148515	0.990099	149.607261	0.326733	1.039604	1.600660	0.672241	4.7
std	9.038662	0.467299	0.960126	17.599748	51.776918	0.356198	0.994971	22.875003	0.469794	1.161075	0.616226	0.937438	1.9
min	29.000000	0.000000	1.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	1.000000	0.000000	3.0
25%	48.000000	0.000000	3.000000	120.000000	211.000000	0.000000	0.000000	133.500000	0.000000	0.000000	1.000000	0.000000	3.0
50%	56.000000	1.000000	3.000000	130.000000	241.000000	0.000000	1.000000	153.000000	0.000000	0.800000	2.000000	0.000000	3.0
75%	61.000000	1.000000	4.000000	140.000000	275.000000	0.000000	2.000000	166.000000	1.000000	1.600000	2.000000	1.000000	7.0
max	77.000000	1.000000	4.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	3.000000	3.000000	7.0

We can also plot the correlation of each column using matplotlib:
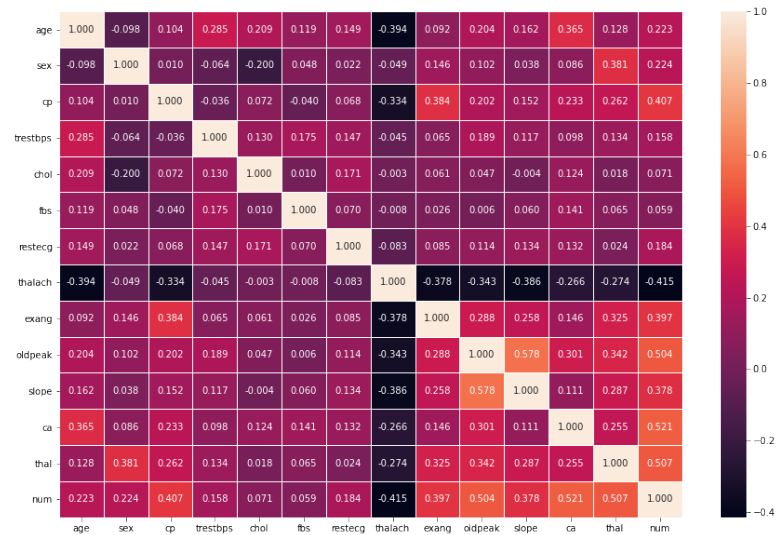
This plot allows us to see each data point in correlation to the next. We also graphed heart disease based on sex:
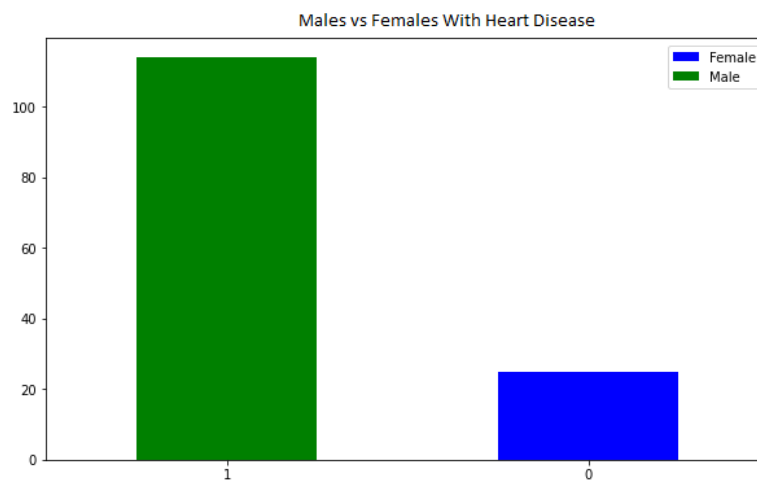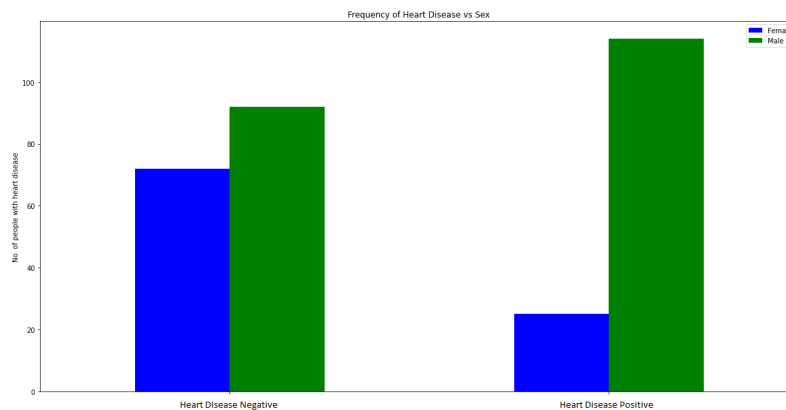
These 2 graphs allow us to examine the number of patients with and without heart disease categorized by their sex. We display these 3 graphs on the dashboard of the application so doctors can have a better understanding of the importance of each data point as well as how common the disease is among sexes.

D.6 Accuracy Analysis

As discussed in the hypothesis section, our initial goal was to have the application at around 75% accuracy. However, from our testing and analysis, we find the model to be accurate only about 65% of the time. While it is a good start, it is not accurate enough to be of use. That is okay, however, seeing as this is just the prototype. The model can and will be tweaked in the future to be more accurate, and there is no doubt that we can reach 75%, possibly even more. This low accuracy model acted as a proof of concept that AI can assist in medical diagnosis.

D.7 Application Testing

Testing is ongoing throughout the development process. During the initial development phase, the data was randomly split up into a training set and a testing set. The training set was used to train the model, and the testing set was used to test the trained model. We are getting 65.57% accuracy on our testing set.

```
LR_model_r2_score=round(r2_score(y_test,LR_model_y_pred)*100,2)
print("Accuracy on Training set: ",round(LR_model.score(X_train_final,y_train)*100,2))
LR_model_score = round(LR_model.score(X_test_final,y_test)*100,2)
print("Accuracy on Testing set: ",LR_model_score)

Accuracy on Training set:  62.4
Accuracy on Testing set:  65.57
```

Our goal will be to improve on this accuracy for the next sprint.

D.8 Application Files

correlationmap.png – image of data column correlation used on the application dashboard

frequency.png – image of frequency of heart disease between males and females used on the application dashboard

malesvsfemales.png – image of heart disease positive cases between males and females used on the dashboard

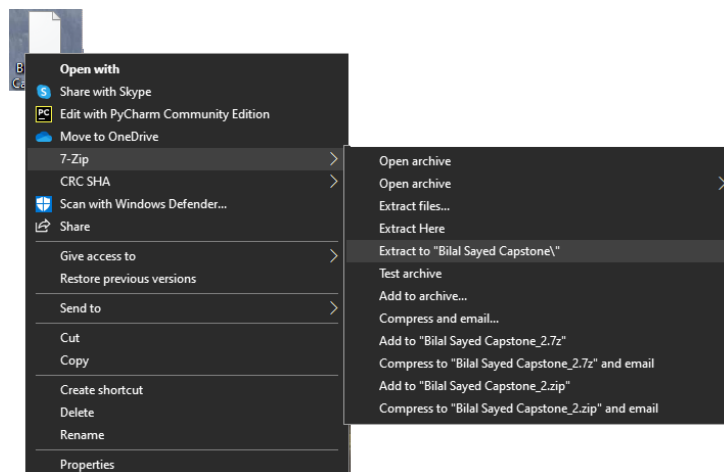**model.joblib** – joblib file of the trained model

main.py – python file of the application

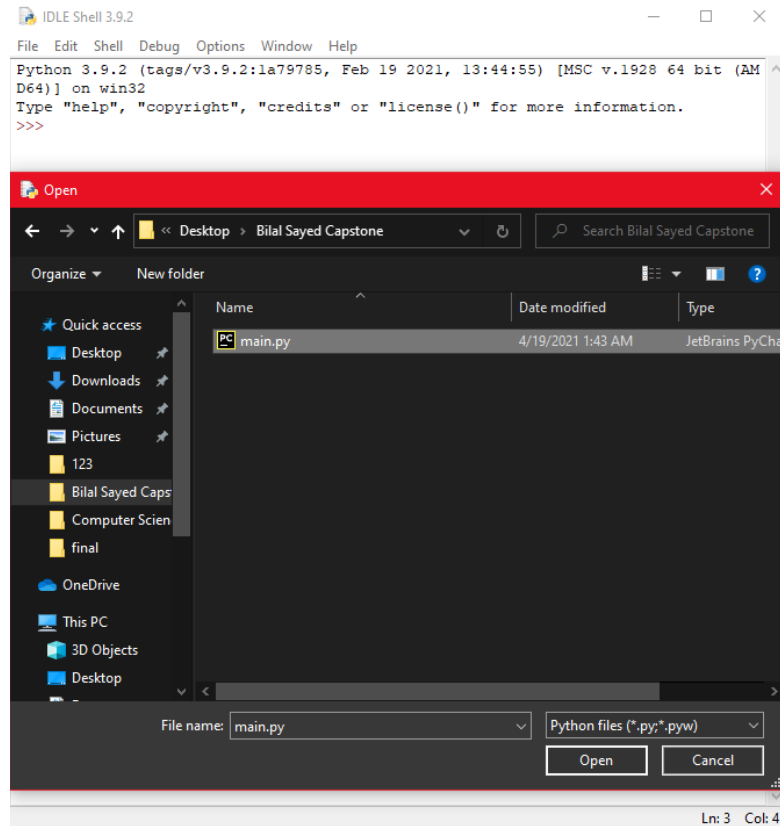**processed.cleveland.csv** – csv file of the patient data

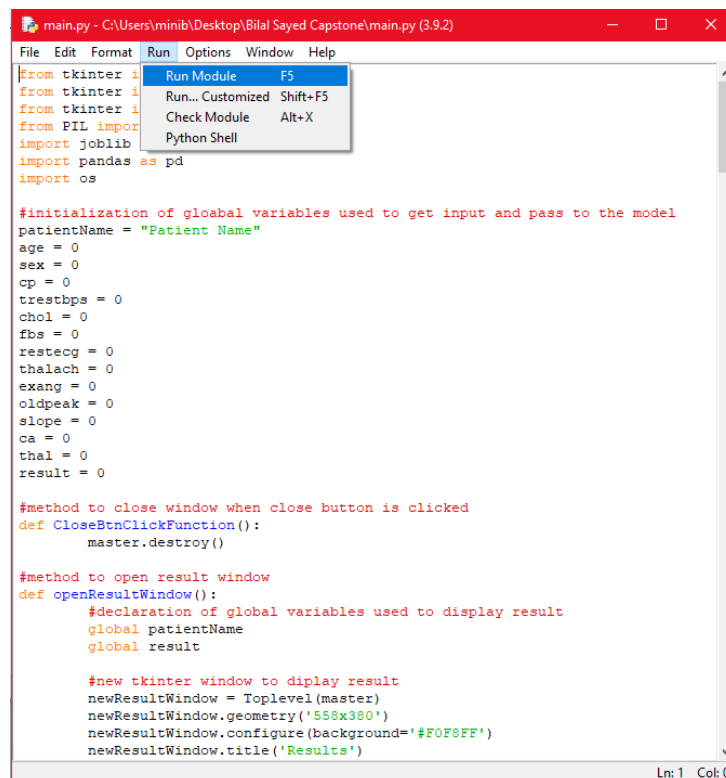**final model.ipynb** – Jupyter Notebook file of model

D.9 User's Guide

First download the zip folder included with this document. Next unzip the contents of the file:

Next, open IDLE and click File>Open. Then navigate to the unzipped folder:

Next click File>Run Module(or press F5):

That's it! You are at the dashboard now. Take a moment to view the graphs provided. Next click the predict button to open the prediction window:
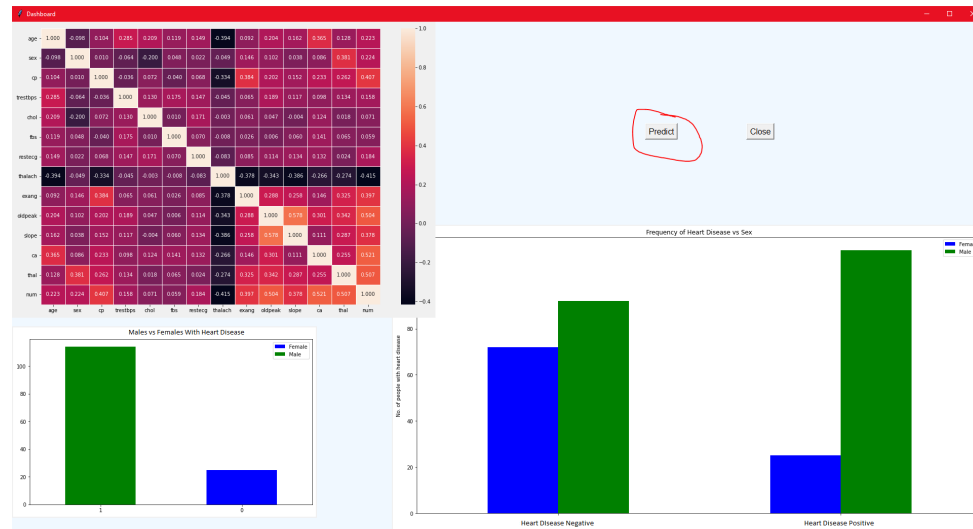
Now you are in the prediction window. Enter the patient name, and all their data:
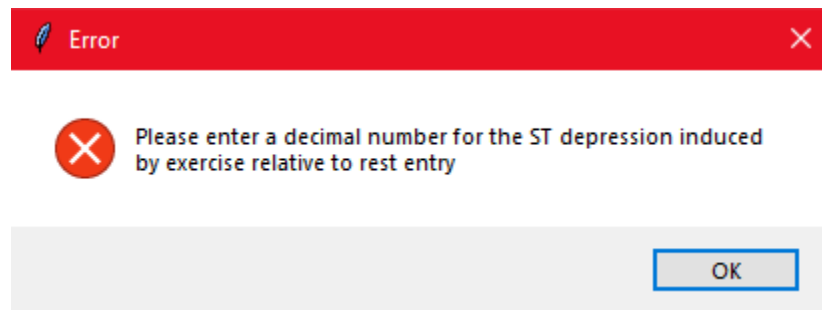

Next click submit to submit the data and receive a result:
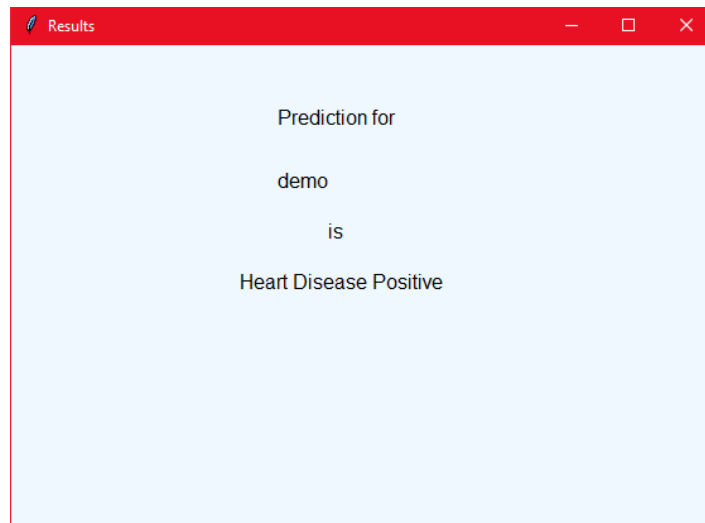
That is how to use the Heart Disease Prediction Application.

Troubleshooting:

If you receive an error screen:

Simply click ok, then enter a decimal value in the ST depression induced by exercise relative to rest field. Then click submit.

E. Sources

Jackson, D. (n.d.). Software development price guide & hourly rate comparison. Retrieved April 19, 2021, from https://www.fullstacklabs.co/blog/software-development-price-guide-hourly-rate-comparison

Windsor, G. (2020, February 28). 5 stages of the Agile system development life cycle - BRIGHTWORK.COM. Retrieved April 19, 2021, from https://medium.com/brightwork-collaborative-project-management-blog/5-stages-of-the-agile-system-development-life-cycle-brightwork-com-a207bdf61696