ADVANCED DATA SCIENCE CAPSTONE PROJECT: PREDICTION OF CARDIOVASCULAR EVENTS.

PREPARED BY: MAXIM LUKIN

AS OF DATE: 2/10/2020

Use Case:

Cardiovascular diseases (CVDs) are the number 1 cause of death globally, taking an estimated 17.9 million lives each year *.

Data Set:

For current case study Heart Disease Data Set has been chosen **.

Data Set Creators:

- 1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.
- 2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.
- 3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.
- 4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D.

* https://www.who.int/health-topics/cardiovascular-diseases/#tab=tab_1

** https://www.kaggle.com/ronitf/heart-disease-uci

- **Libraries and Versions:**
- Python: 3.6.9 |Anaconda, Inc.| (default, Jul 30 2019, 19:07:31)
- [GCC 7.3.0]
- Pandas: 0.25.3
- Numpy: 1.15.4
- Sklearn: 0.20.3
- Matplotlib: 3.1.3
- Keras: 2.2.4

Data Quality Assessment:

Heart Disease Data Set from Kaggle has been initially available as a cleansed and transformed/adopted data set.

During ETL process dataset has been checked once again for any possible missing values in data frame and inappropriate attributes formats.

- Data Attribute Information:
- age {age}
- sex (1 = male; 0 = female) {sex}
- chest pain type (4 values) {cp}
- resting blood pressure {trestbps}
- serum cholestoral in mg/dl {chol}
- fasting blood sugar > 120 mg/dl (1 = true; 0 = false) {fbs}
- resting electrocardiographic results (values 0,1,2) {restecg}
- maximum heart rate achieved {thalach}
- exercise induced angina {exang}
- oldpeak = ST depression induced by exercise relative to rest {oldpeak}
- the slope of the peak exercise ST segment {slope}
- number of major vessels (0-3) colored by flourosopy {ca}
- thal: 3 = normal; 6 = fixed defect; 7 = reversable defect {thal}

```
# check for missing values
df.isnull().sum()
```

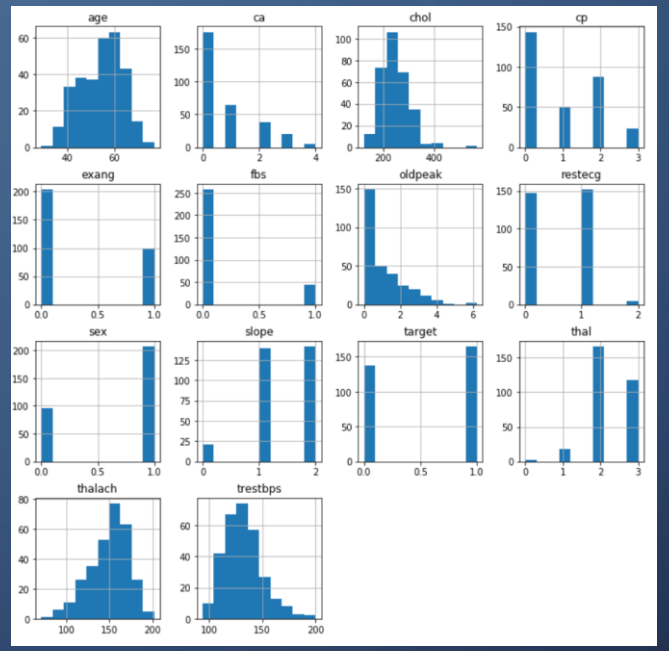
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

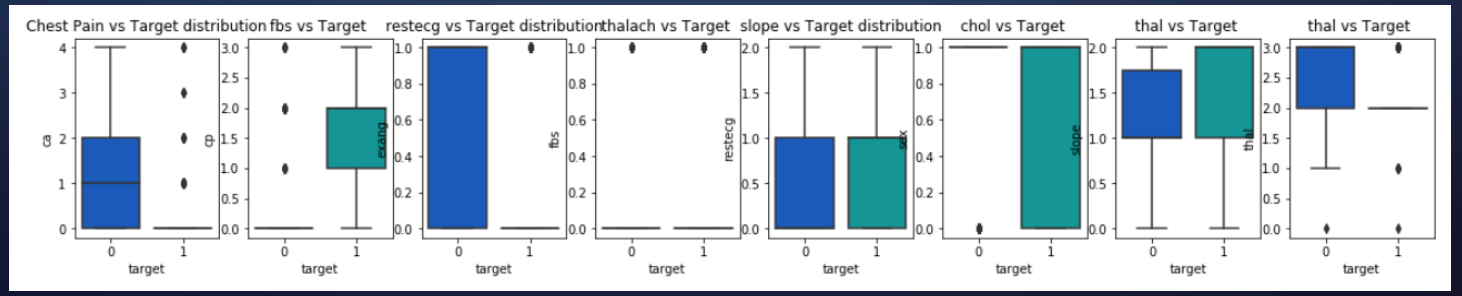
```
# summary of dataset
df.info()
```

```
<class 'pandas.core.frame.DataFrame
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
age         303 non-null int64
sex         303 non-null int64
cp          303 non-null int64
trestbps    303 non-null int64
chol        303 non-null int64
fbs         303 non-null int64
restecg     303 non-null int64
thalach     303 non-null int64
exang       303 non-null int64
oldpeak     303 non-null float64
slope       303 non-null int64
ca          303 non-null int64
thal        303 non-null int64
target      303 non-null int64
dtypes: float64(1), int64(13)
memory usage: 33.2 KB
```

Data Visualization

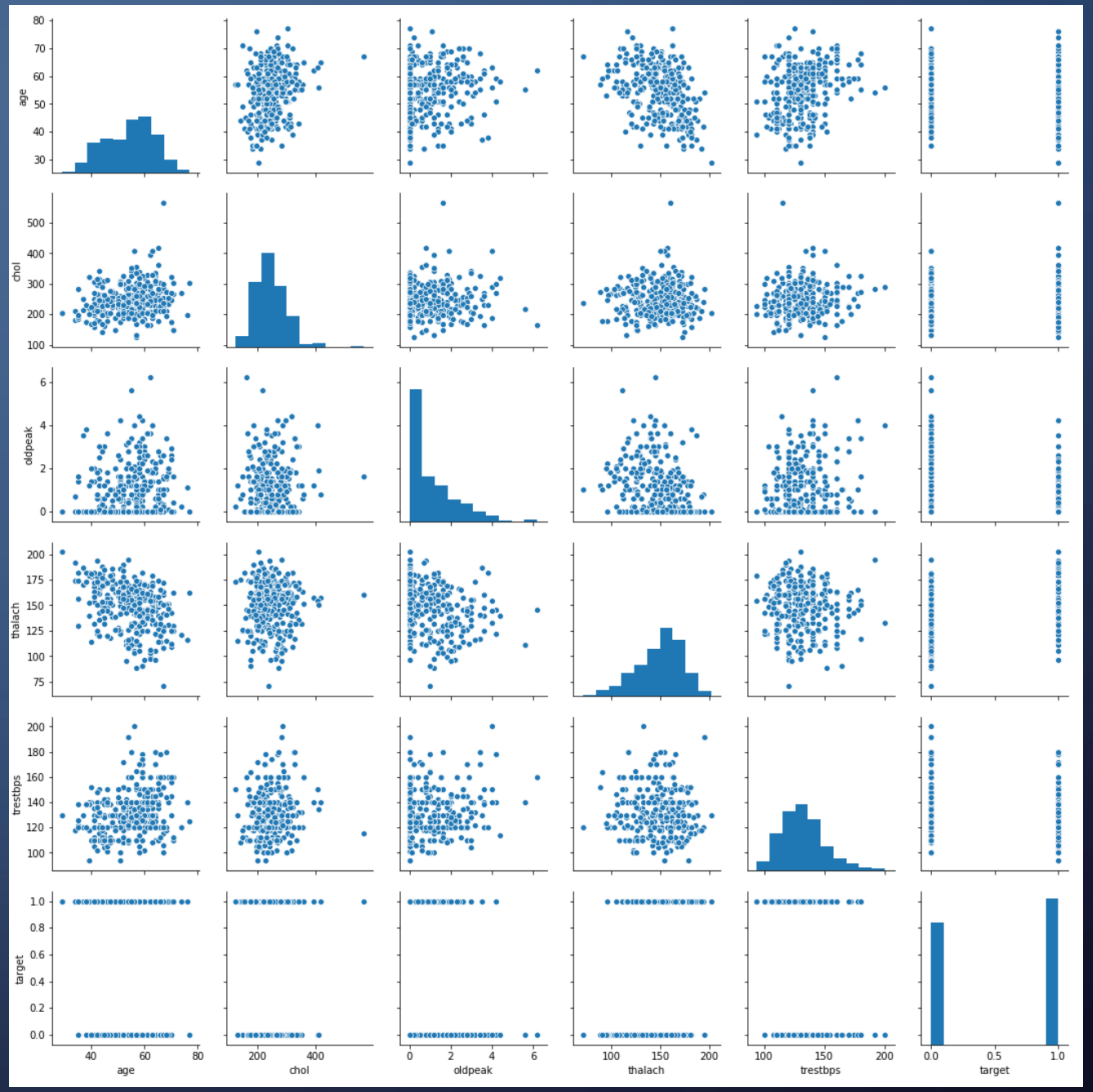
- Data Attribute Information:
- age {age}
- sex (1 = male; 0 = female) {sex}
- chest pain type (4 values) {cp}
- resting blood pressure {trestbps}
- serum cholestoral in mg/dl {chol}
- fasting blood sugar > 120 mg/dl (1 = true; 0 = false) {fbs}
- resting electrocardiographic results (values 0,1,2) {restecg}
- maximum heart rate achieved {thalach}
- exercise induced angina {exang}
- oldpeak = ST depression induced by exercise relative to rest {oldpeak}
- the slope of the peak exercise ST segment {slope}
- number of major vessels (0-3) colored by flourosopy {ca}
- thal: 3 = normal; 6 = fixed defect; 7 = reversable defect {thal}





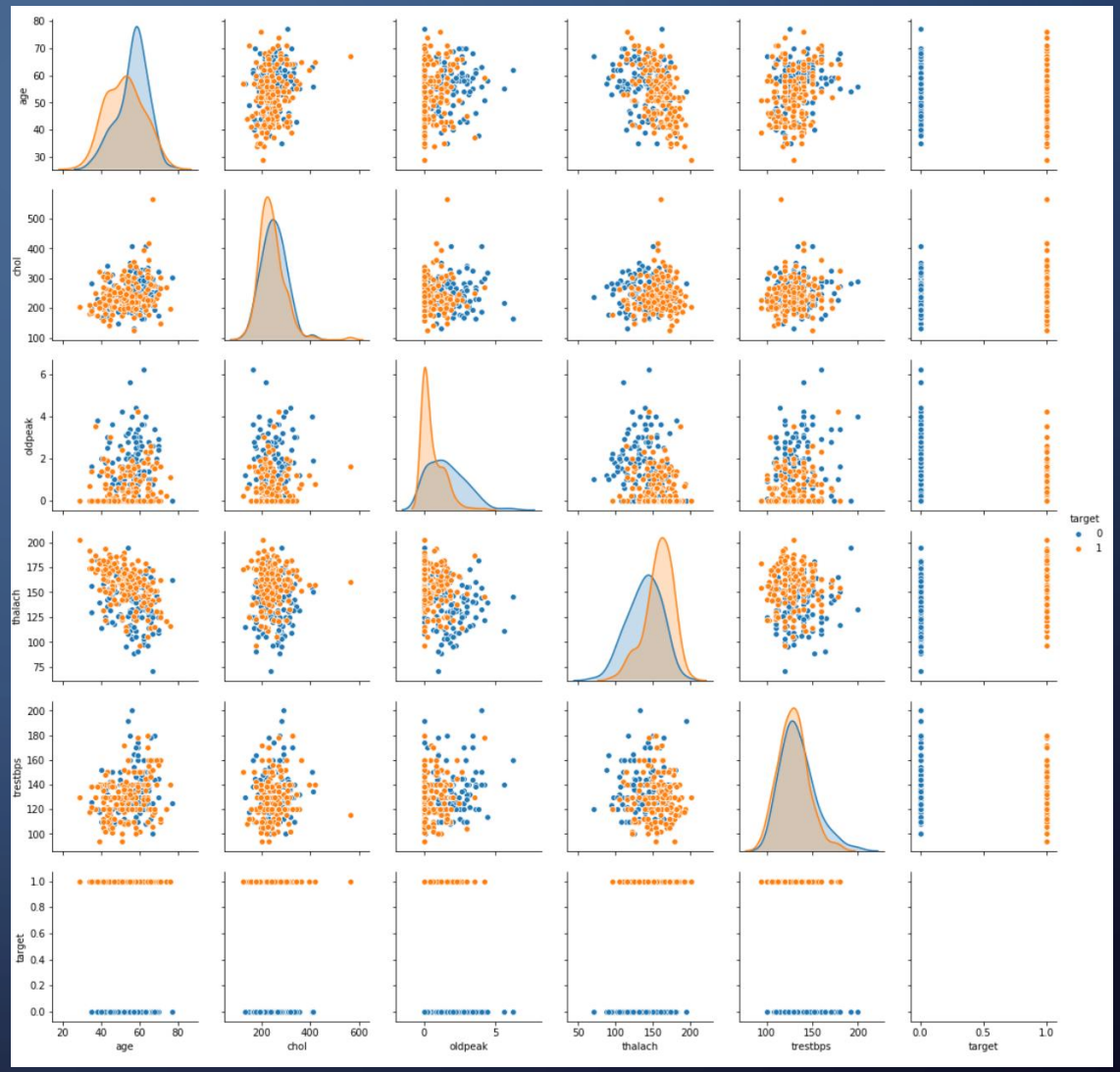
Data Visualization

- Data Attribute Information:
- age {age}
- sex (1 = male; 0 = female) {sex}
- chest pain type (4 values) {cp}
- resting blood pressure {trestbps}
- serum cholestoral in mg/dl {chol}
- fasting blood sugar > 120 mg/dl (1 = true; 0 = false) {fbs}
- resting electrocardiographic results (values 0,1,2) {restecg}
- maximum heart rate achieved {thalach}
- exercise induced angina {exang}
- oldpeak = ST depression induced by exercise relative to rest {oldpeak}
- the slope of the peak exercise ST segment {slope}
- number of major vessels (0-3) colored by flourosopy {ca}
- thal: 3 = normal; 6 = fixed defect; 7 = reversable defect {thal}



Data Visualization

- Data Attribute Information:
- age {age}
- sex (1 = male; 0 = female) {sex}
- chest pain type (4 values) {cp}
- resting blood pressure {trestbps}
- serum cholestoral in mg/dl {chol}
- fasting blood sugar > 120 mg/dl (1 = true; 0 = false) {fbs}
- resting electrocardiographic results (values 0,1,2) {restecg}
- maximum heart rate achieved {thalach}
- exercise induced angina {exang}
- oldpeak = ST depression induced by exercise relative to rest {oldpeak}
- the slope of the peak exercise ST segment {slope}
- number of major vessels (0-3) colored by flourosopy {ca}
- thal: 3 = normal; 6 = fixed defect; 7 = reversable defect {thal}

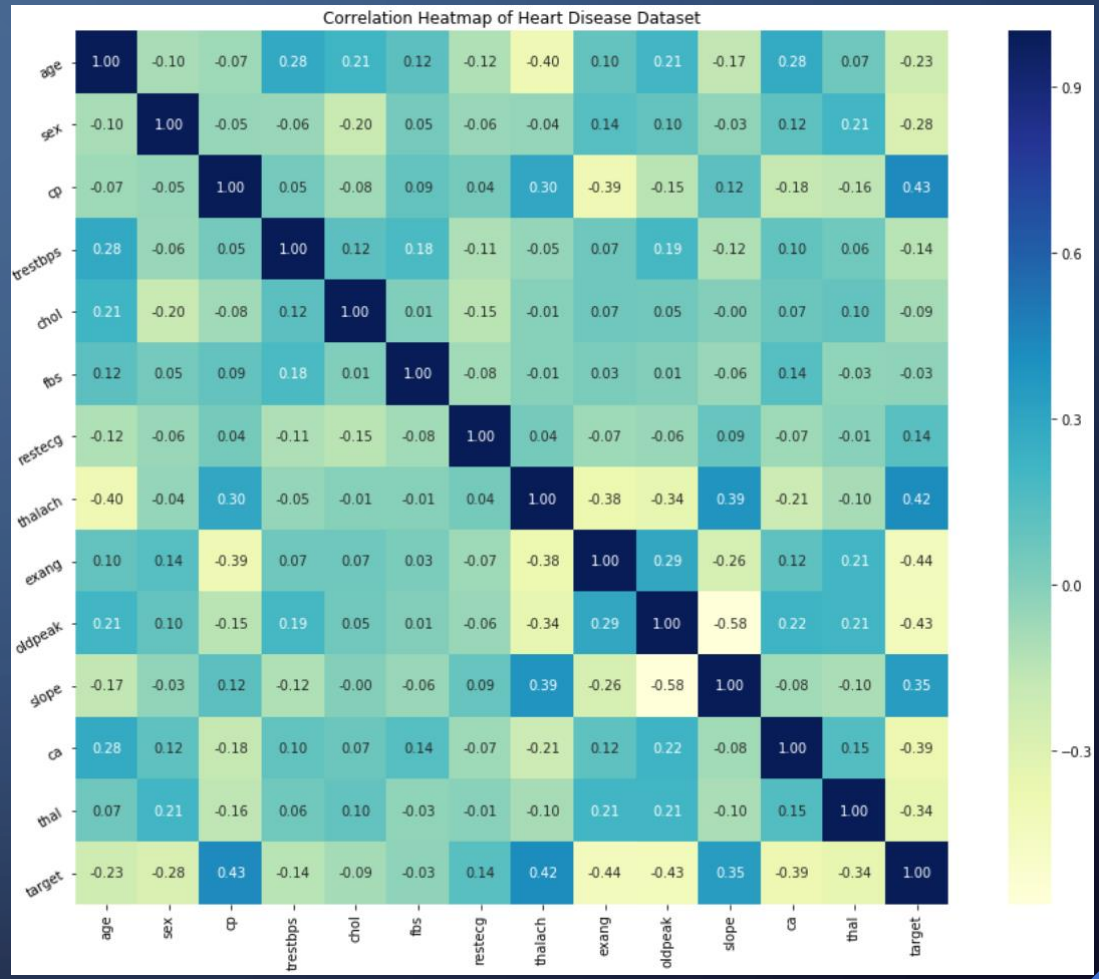


Data Visualization

- Data Attribute Information:
- age {age}
- sex (1 = male; 0 = female) {sex}
- chest pain type (4 values) {cp}
- resting blood pressure {trestbps}
- serum cholestoral in mg/dl {chol}
- fasting blood sugar > 120 mg/dl (1 = true; 0 = false) {fbs}
- resting electrocardiographic results (values 0,1,2) {restecg}
- maximum heart rate achieved {thalach}
- exercise induced angina {exang}
- oldpeak = ST depression induced by exercise relative to rest {oldpeak}
- the slope of the peak exercise ST segment {slope}
- number of major vessels (0-3) colored by flourosopy {ca}
- thal: 3 = normal; 6 = fixed defect; 7 = reversable defect {thal}

```
# Checking correlation
correlation = df.corr()
correlation['target'].s
```

```
target      1.000000
cp          0.433798
thalach     0.421741
slope       0.345877
restecg     0.137230
fbs        -0.028046
chol       -0.085239
trestbps   -0.144931
age        -0.225439
sex        -0.280937
thal       -0.344029
ca         -0.391724
oldpeak    -0.430696
exang      -0.436757
Name: target, dtype: fl
```



Training and testing models and algorithms

Based on the capstone project requirements and for educational purposes the following algorithms have been used:

- Logistic Regression,
- Support Vector Machines,
- Linear Support Vector Machines (SVC),
- k-Nearest Neighbors algorithm (KNN),
- Gaussian Naive Bayes,
- Perceptron,
- Stochastic Gradient Descent,
- Decision Tree Classifier,
- Random Forest,
- Ridge Classifier.

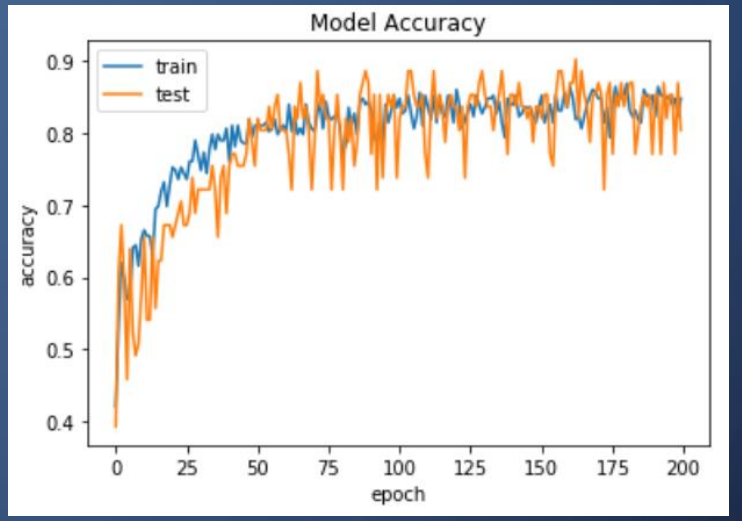
```
models.sort_values(by=['Score_test', 'Score_train'], ascending=False)
```

	Model	Score_train	Score_test
0	Logistic Regression	84.71	85.25
4	Naive Bayes	83.47	85.25
8	Random Forest	100.00	83.61
2	Linear SVC	84.30	83.61
9	RidgeClassifier	83.47	83.61
7	Decision Tree Classifier	100.00	77.05
5	Perceptron	67.77	70.49
6	Stochastic Gradient Decent	63.22	70.49
3	k-Nearest Neighbors	78.10	63.93
1	Support Vector Machines	100.00	59.02

https://eu-gb.dataplatform.cloud.ibm.com/analytics/notebooks/v2/4c28ac5d-30c8-4029-85c6-bebfc22418ec/view?projectid=535fa780-2008-4612-9dad-6472967021c4&context=analytics

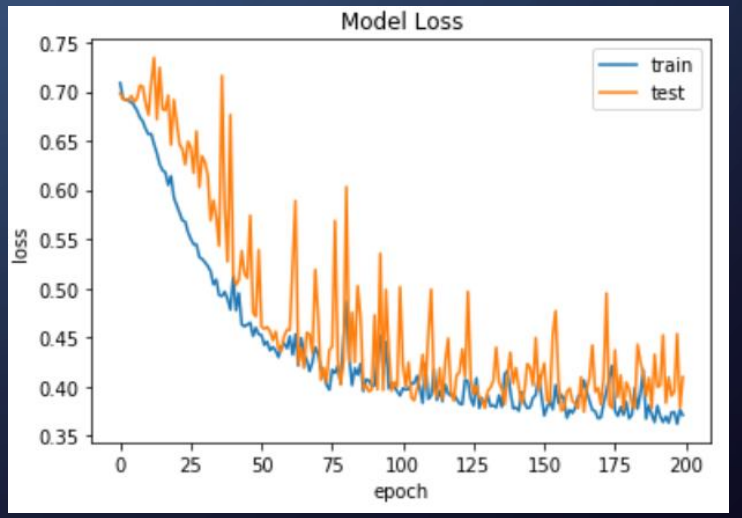
Keras Neural Network

```python
# define a function to build the keras model
def create_model():
    # create model
    model = Sequential()
    model.add(Dense(16, input_dim=13, kernel_initializer='normal', activation='relu'))
    model.add(Dense(8, kernel_initializer='normal', activation='relu'))
    model.add(Dense(2, activation='softmax'))

    # compile model
    adam = Adam(lr=0.001)
    model.compile(loss='categorical_crossentropy', optimizer=adam, metrics=['accuracy'])
    return model

model = create_model()
```



Results for Categorical Model
0.8032786885245902

	precision	recall	f1-score	support
0	0.70	0.88	0.78	24
1	0.90	0.76	0.82	37
micro avg	0.80	0.80	0.80	61
macro avg	0.80	0.82	0.80	61
weighted avg	0.82	0.80	0.81	61

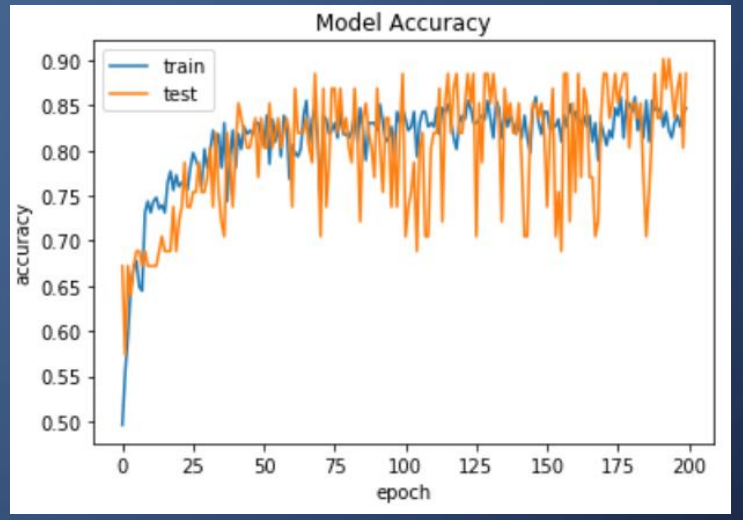


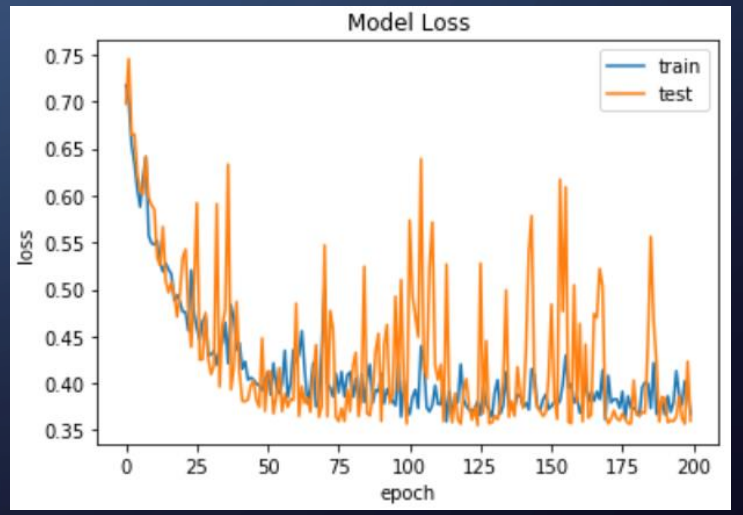
Keras Neural Network (one more iteration):

Loss: categorical => binary

Output layer: softmax => sigmoid

```
def create_binary_model():
    # create model
    model = Sequential()
    model.add(Dense(16, input_dim=13, kernel_initializer='normal', activation='relu'))
    model.add(Dense(8, kernel_initializer='normal', activation='relu'))
    model.add(Dense(1, activation='sigmoid'))

    # Compile model
    adam = Adam(lr=0.001)
    model.compile(loss='binary_crossentropy', optimizer=adam, metrics=['accuracy'])
    return model

binary_model = create_binary_model()
```






```
Results for Binary Model
0.8852459016393442
              precision    recall  f1-score   support

           0       0.87      0.83      0.85        24
           1       0.89      0.92      0.91        37

   micro avg       0.89      0.89      0.89        61
   macro avg       0.88      0.88      0.88        61
weighted avg       0.88      0.89      0.88        61
```

THANK YOU FOR YOUR TIME ! ! !

Postscript:

That paper has been created based on the studies of the following contributors and data scientists:

https://www.kaggle.com/prashant111/extensive-eda-visualization-with-python
https://www.kaggle.com/faressayah/heart-disease-eda-9-ml-algorithms-90
https://www.kaggle.com/tentotheminus9/what-causes-heart-disease-explaining-the-model#The-Model
https://www.kaggle.com/cdabakoglu/heart-disease-classifications-machine-learning
https://www.kaggle.com/vbmokin/heart-disease-comparison-of-20-models
https://www.kaggle.com/rahul197/heart-disease-classification-machine-learning
https://www.kaggle.com/mytymohan/heart-disease-eda-lr-dt-rf-gb-svm-dl
https://towardsdatascience.com/machine-learning-for-diabetes-562dd7df4d42
https://towardsdatascience.com/machine-learning-for-diabetes-562dd7df4d42

Thanks to everyone! Your work was very helpful for me!