A study with logistic regression and Random Forest Classifier model on the Cleveland Heart Disease data set.

Tianqi Luan

The Cleveland heart disease dataset was created by John Gennari based on Dr. Detrano's database and was modified to be a real MIXED dataset.

The original database contains 76 attributes, but all published experiments refer to using a subset of 14 of them. Attributes: 8 symbolic, 6 numeric:

Age; sex; chest pain type (angina, abnang, notang, asympt); Trestbps (resting blood pres); cholesteral; fasting blood sugar < 120; (true or false); resting ecg (norm, abn, hyper); max heart rate; exercise induced angina (true or false); oldpeak; slope (up, flat, down); number of vessels colored (???); thal (norm, fixed, rever). Finally, the class is either healthy (buff) or with heart-disease (sick).

For pre-processing, we use the original attributes to replace the symbolic attributes: age; sex (1,0); cp (1-4); trestbps; chol; fbs (1,0); restecg (0,1,2); thalach; xang (1,0); oldpeak; slope (1,2,3); ca; thal (3,6,7); class att: 0 is healthy, 1,2,3,4 is sick.

All process above was done on Microsoft Excel, then the original .txt file was saved as .csv file for ease of future analysis in Python.

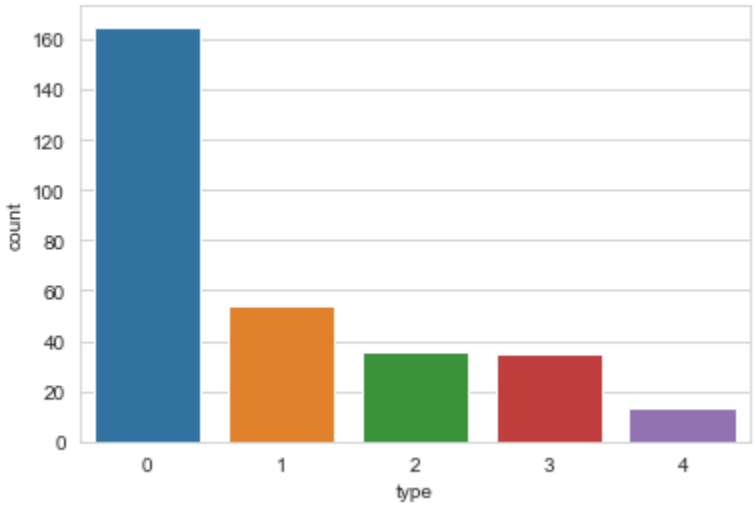
The pre-processed file was uploaded to JupyterNotebook. The .csv file was loaded with the panda extension, graphing work was done mainly by matplotlib.pyplot and seaborn.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

And after the pre-processing above, all attributes are now either int or float as shown below:

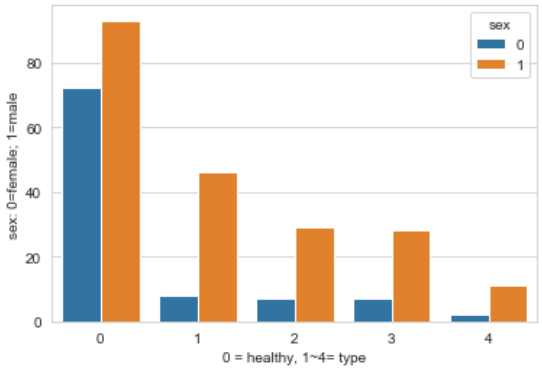
	Age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	num	type
0	63	1	1	145	233	1	2	150	0	2.3	3	0.0	6.0	0	0
1	67	1	4	160	286	0	2	108	1	1.5	2	3.0	3.0	1	2
2	67	1	4	120	229	0	2	129	1	2.6	2	2.0	7.0	1	1
3	37	1	3	130	250	0	0	187	0	3.5	3	0.0	3.0	0	0
4	41	0	2	130	204	0	2	172	0	1.4	1	0.0	3.0	0	0

Now we have 3 float type attributes and 12 int type attributes. Since our goal is classify the last column from 0 (healthy) to 1~4 (sick), here is summary of last column's values.

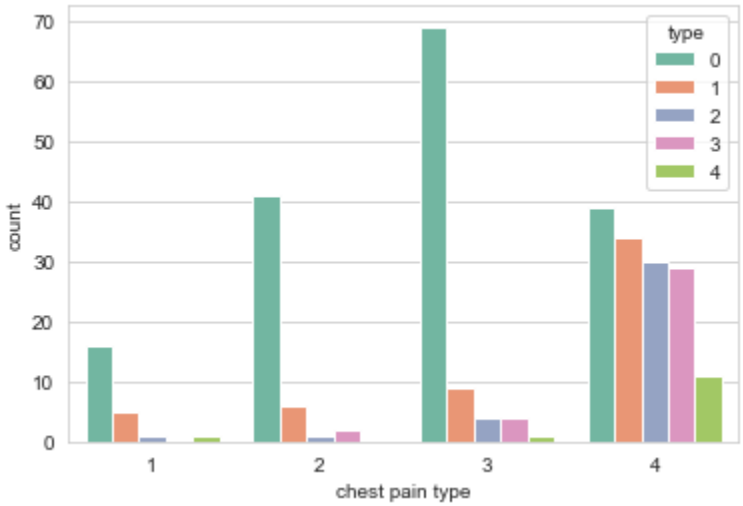


From this graph we can tell the majority of the column has value 0 (healthy), type 1, 2, 3 have similar amount of data while type 4 have only 13 data points.

Then the relationship between sick type and other attributes was researched. First attribute was sex:



First, male is the majority in this dataset. But this aspect considered, male is still more prone to heart disease as we seen here: in healthy group we have comparable amount of both sex yet in all sick groups there are multiple times males more than female. Next attribute that can be highly relatable to the heart disease is the chest pain type:



From all the patients with type 1/2/3 chest pain, the amount with actual heart disease is quite minor. Yet in all the type 4 chest pain patients. Only less than half of them are healthy. So we can assume that only type 4 chest pain strongly indicates heart disease while the other 3 types don't.

First method used to carry out the actual classification is linear logistic regression, all extensions used listed below:

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler, LabelEncoder
```

Test sized was 0.25 with 100 random states. And the prediction after standard scalar was applied is:

```
array([1, 0, 0, 3, 0, 3, 1, 0, 2, 0, 0, 0, 0, 0, 0, 0, 1, 3, 3, 0, 0, 0,
       1, 3, 2, 1, 1, 0, 3, 3, 3, 0, 0, 0, 3, 0, 0, 1, 0, 0, 1, 3, 0, 1,
       1, 2, 0, 3, 1, 2, 4, 0, 0, 3, 0, 3, 1, 3, 0, 3, 3, 3, 1, 1, 0, 1,
       1, 3, 0, 1, 1, 2, 3, 1, 0, 0], dtype=int64)
```

Apply confusion matrix to test the prediction accuracy:

```
              precision    recall  f1-score   support

           0       0.88      0.74      0.80        38
           1       0.21      0.40      0.28        10
           2       0.00      0.00      0.00        14
           3       0.21      0.40      0.28        10
           4       0.00      0.00      0.00         4

    accuracy                           0.47        76
   macro avg       0.26      0.31      0.27        76
weighted avg       0.49      0.47      0.47        76

Accuracy score: 0.47368421052631576

 confusion_matrix:
 [[28  4  0  5  1]
 [ 4  4  1  1  0]
 [ 0  7  0  7  0]
 [ 0  3  3  4  0]
 [ 0  1  1  2  0]]
```

From simple linear regression, precision of healthy group is pretty fine but prediction to all sick types especially 2 and 4 (lack of support) are both 0 and an overall accuracy of 0.47.

Now we try sklearn.svm for accuracy. The overall accuracy is improved substantially from 0.47 to 0.83 and also see improvement in both healthy group and type 1/2/3 heart disease but still lacking prediction in type 4 heart disease for lacking of support.

```
              precision    recall  f1-score   support

           0       0.95      1.00      0.97        38
           1       1.00      0.60      0.75        10
           2       0.58      1.00      0.74        14
           3       0.83      0.50      0.62        10
           4       0.00      0.00      0.00         4

    accuracy                           0.83        76
   macro avg       0.67      0.62      0.62        76
weighted avg       0.82      0.83      0.80        76

[[38  0  0  0  0]
 [ 2  6  2  0  0]
 [ 0  0 14  0  0]
 [ 0  0  5  5  0]
 [ 0  0  3  1  0]]
Accuracy score: 0.8289473684210527
```

For cross-validation we use Random Forest Classifier to compare the results to linear regression:

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.neural_network import MLPClassifier
```

The number of estimators was set at 200, here is the result:

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        38
           1       0.29      0.90      0.44        10
           2       0.50      0.07      0.12        14
           3       0.20      0.10      0.13        10
           4       0.00      0.00      0.00         4

    accuracy                           0.64        76
   macro avg       0.40      0.41      0.34        76
weighted avg       0.66      0.64      0.60        76

[[38  0  0  0  0]
 [ 0  9  0  1  0]
 [ 0 10  1  3  0]
 [ 0  8  1  1  0]
 [ 0  4  0  0  0]]
Accuracy score: 0.6447368421052632
```

RFC was really good at predicting healthy group but all sick types are not so good. Overall accuracy score 0.64 is better than simple linear regression but still not superior than SVC.

Reference:

1. http://archive.ics.uci.edu/ml/datasets/Heart+Disease
2. https://github.com/LucDemortier/HeartDiseaseStudy
3. https://github.com/kb22/Heart-Disease-Prediction
4. https://towardsdatascience.com/predicting-presence-of-heart-diseases-using-machine-learning-36f00f3edb2c
5. https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html