Assignment no. 3

Title: Perform the categorization of dataset

Theory:-

classification is a large domain in field of statistics and ML. Generally, classification can be broken down into two areas.

* Binary classification: Where we wish to group an outcome into one of two groups.

* Multi-class classification: where we wish to group an outcome into one of multiple groups.

In this post, our focus will be on using a variety of classification algorithm across both of these domains but emphasis will be explained on theory behind them.

We can use libraries in python such as Scikit-learn for ML models and pandas to import data as data frames these can easily be installed and imported into python with pip.

```
$ Python 3 - m pip install sklearn
$ Python 3 - m pip install pandas
```

```
import sklearn as SK
import Pandas as pd
```

Binary classification:

For binary classification, we are interested in classifying data into one of two binary groups - these are usually represented as 0's and 1's in our data.

We look at data regarding coronary heart disease (CHD) in South Africa. the goal is to use different variables such as tobacco usage, family history, cholesterol levels, alcohol usage obesity and race.

A full description of this dataset is available in data sections of elements of statistical learning website

A code below read data into a pandas frame and then separates dataframe into a y vector of responses and the an matrix of explantory variables.

```
import pandas as pd

os.chdir('/users/steven.hurwitt/document/Blog/classification')
heart = pd.read_csv('SAHeart.csv', sep=',', header=0)
heart.head()
y = heart.iloc[:9]
x = heart.iloc[:,:9]
```

When running this code, just be sure to change file system path and name to suit your setup.

sbp	tobacco	ldl	adiposity	famhist	typea	obesity	alcohol	
0	160	12.00	5.73	23.11	1	49	25.30	97.20 52 1
1	144	6.01	4.44	28.61	0	55	28.87	2.66 63 1
2	118	0.08	3.48	32.28	1	52	29.14	3.81 46 0
3	170	7.50	6.41	38.03	1	51	31.99	24.26 58 1
4	134	13.60	3.50	27.78	1	60	25.99	34.49 1

Logistic Regression :-

Logistic regression is type of generalized linear model GLM that uses a logistic function to model a binary variable based on any kind of independent variables

To fit a binary logistic regression with sklearn. We use the logistic regression module with multi-class set to "ovr" and fit x and y

We save then use predict method to predict probabilities of new data as well as score method to get mean prediction accuracy.

Support vector machines :-

Support vector machines are a type of classification algorithm that are more flexible - they can do linear classification, but can use non-linear kernel functions the following examples uses a linear classifier to fit a hyperplane that separates data into two classes.

Random forests :-

Random forests are an ensemble learning method that multiple decision trees on subset of data and average

the results. We can explain if there using sklearn and use them to predict outcomes as well as get mean prediction accuracy.

Neural network:

Neural networks are ML algorithms that involves fitting many hidden layers used to represent neurons that are connected with synaptic activation functions. These essentially use a very simplified models of brain to model and predict data.

We use sklearn for consistency in this post, however, libraries such as Tensorflow and Keras are more suited to fitting and customizing neural networks of which there are few variates used for different purposes

Multi class classification:

While binary classification alone is incredibly useful, there are times when we would like to model and predict data that has more than two classes. many of same algorithms can be used with slight modifications.

Additionally, it is common to split data into training and test sets. This means we use a certain portion of data to fit model and save remaining portion of it is to evaluate the prediction accuracy of fitted model.

There's no official rule to follow when deciding on a split proportion though in most cases you'd ~~that~~ want about 70% to be dedicated for training set &

for training set and around 30% for test set.

To explore both multi-class classifiers as well as training / test data, we will look at another dataset from elements of statistical learning website. This is data used to determine which one of eleven vowel sounds were spoken

Although implementations of these models were rather naive we can still compare predictive accuracy across models. This will tell us which one is most accurate for this specific training & test dataset:

Model	predictive accuracy
Logistic Regression	46%
Support vector machine	64.07%
Random forest	57.58%
Neural network	54.55%

This show us that for vowel data, SVM using default radial basis function was most accurate.

conclusion:

To summarize this post, we began by exploring

Simplest form of classification : Binary This helped us to model data where our response could take one of two states.

We then moved further into multi - class classification when response variable can take any number of states

We also saw how to fit and evaluate models with training and test sets. Furthermore, we could explore additional way to define model fitting among various algorithms.