Report on Abalone

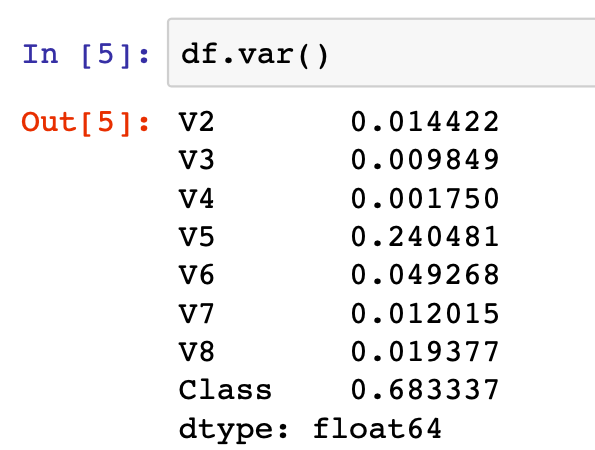
**Step1: Performing initial EDA:**

Abalone are is a common name for any of a group of small to very large sea snails, marine gastropod molluscs in the family Haliotidae. Other common names are ear shells, sea ears, and muttonfish or mutton shells in Australia, ormer in the UK, abalone in South Africa, and paua in New Zealand. Abalone are marine snails.



* There are 7 numerical & 2categorical columns are there in data set
* The variance of the numerical columns are:

I used the code:



Hence the variance is high in the V8 column

* Normalize the numerical columns in the range of (0-1):

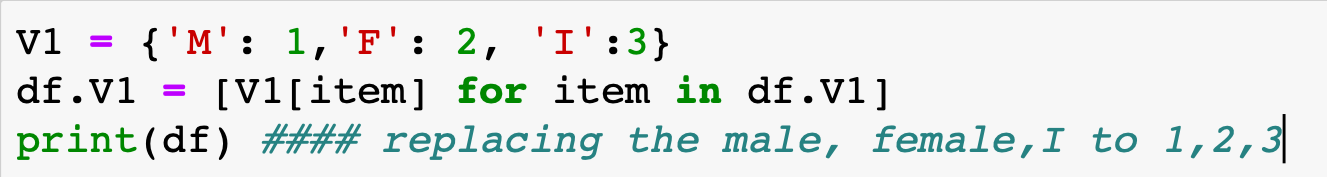
I used the code :

I have taken only the numerical columns to normalize between 0-1

1. df1=df[['V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8']] – creating a new database all numerical columns to one column
2. import sklearn
3. from sklearn import preprocessing – these 2nd and 3rd step
4. df1.columns – checking the columns
5. After normalizing the numerical columns it has been converted in between 0-1 by code: min\_max\_scaler = preprocessing.MinMaxScaler(), x\_scaled = min\_max\_scaler.fit\_transform(x), df1\_normalized = pd.DataFrame(x\_scaled), df1\_normalized.head() #normalized dataframe – by using this we get normalized columns and shows normalized data

* **Convert all categorical columns as one-hot encoding:**

Since I have two categorical variables one categorical variable is in words and one categorical variable in the numeric so I converted that M,f,I in to the 1,2,3 by code



**Step2: Perform classification:**

* Performed a logistic regression
* Data is splitted into training and test data.
* Now the trained data is done on the test data
* After doing that I got the as 61% as accuracy
* After that I done confusion matrix
* Then confusion matrix is plotted and where false positive is 1.6% and false negative is 76%
* Performed a classification using multi-layer neural network model by code:(from sklearn.neural\_network import MLPClassifier, from sklearn.metrics import accuracy\_score) and taken hidden layers size is 150,99 respectively and I got the accuracy of 66% accuracy for this model after this confusion matrix is plotted and the confusion matrix is plotted and where false positive is 9.09% and false negative is 88%
* taken hidden layers size is 1200,1100 respectively and I got the accuracy of 67% accuracy for this model after this confusion matrix is plotted and the confusion matrix is plotted and where false positive is 10.44% and false negative is 90%

Here my observation is that when we increase size of hidden layers then the percentage of false positive and false negative is increases

**Steps3: Try different kind of neural network models:**

* In this by taking hidden layers size is 10,30,50,60 respectively and I got the accuracy of 65% accuracy for this model after this confusion matrix is plotted and the confusion matrix is plotted and where false positive is 7.6% and false negative is 85%
* In this by taking hidden layers size is 10,30,50,60,80,60 respectively and I got the accuracy of 66% accuracy for this model after this confusion matrix is plotted and the confusion matrix is plotted and where false positive is 9.09% and false negative is 88%

Here my observation is that when we increase the more number of hidden layers with the more value then the accuracy changes I have tried the changing with low numbers and increase the values in hidden layers then I see changing in accuracy.