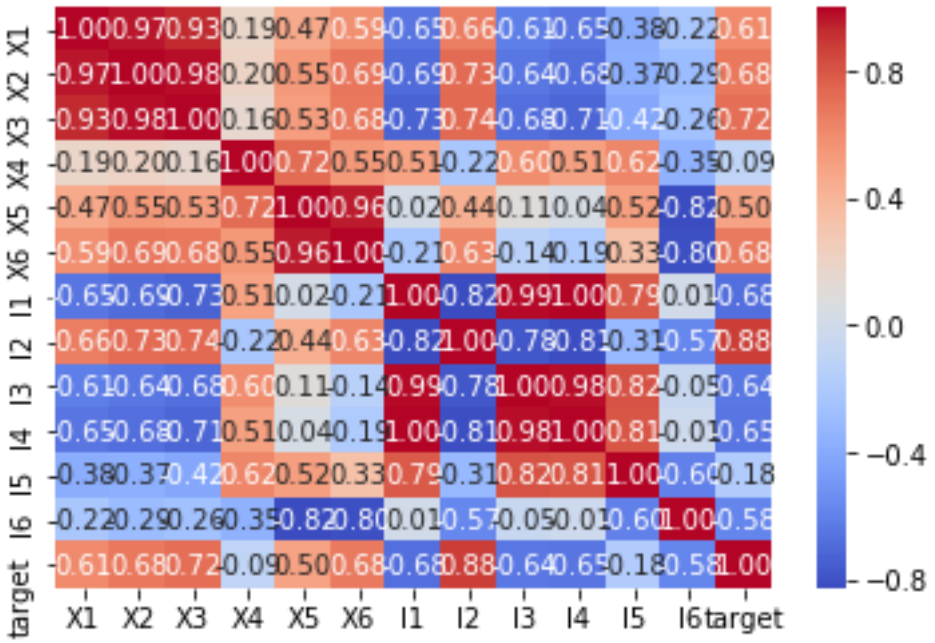
Pulkit garg

Email-id: pulkitgarg33@gmail.com

ML Challenge | Land Classification

The first 3 things you did to understand the data better (including any graphs you plot or summaries you generated)

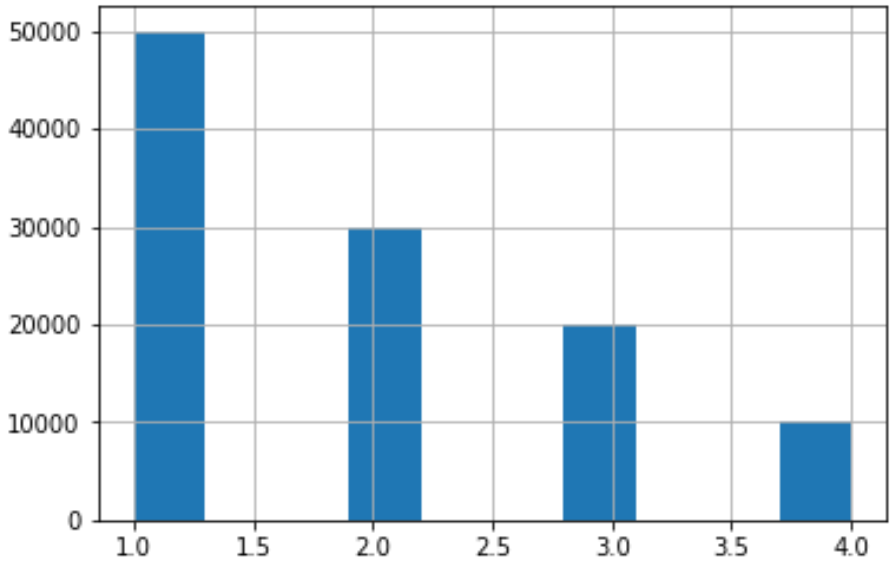
- The first thing was to study the type of variables/ features. All independent variables are of continuous type, while the target variable is of categorical type.
- The target variable has 4 categories :-
    - 1 = Green Land
    - 2 = Water
    - 3 = Barren Land
    - 4 = Built-up
- Features x1 to x6 consist of positive real numbers ranging from 0 to 1000. While features
- Next step was to study the missing values in the data. Luckily data was free from null values.
- Another thing to taken care of was to study the correlation between each feature, which was demonstrated by the given heat map



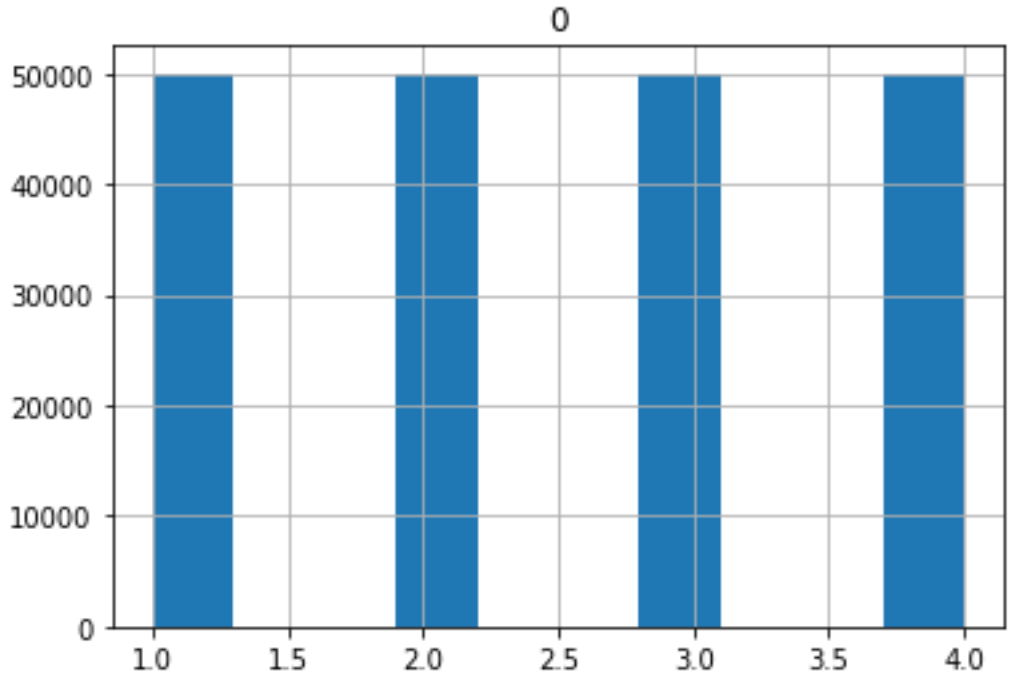
Pre-processing involved

- The data was not shuffled, hence for efficient training we have to shuffle the data.
- No ordinal or nominal features are there so no encoding is required.
- The main issue with the training data was the imbalance in data. As illustrated by the histogram below. Number of samples for each category of target variable were not similar. To solve the problem we have to perform over sampling. In over sampling the observation of the minority classes are randomly replicated until the data is balanced.

  Original dataset



  Dataset after over sampling



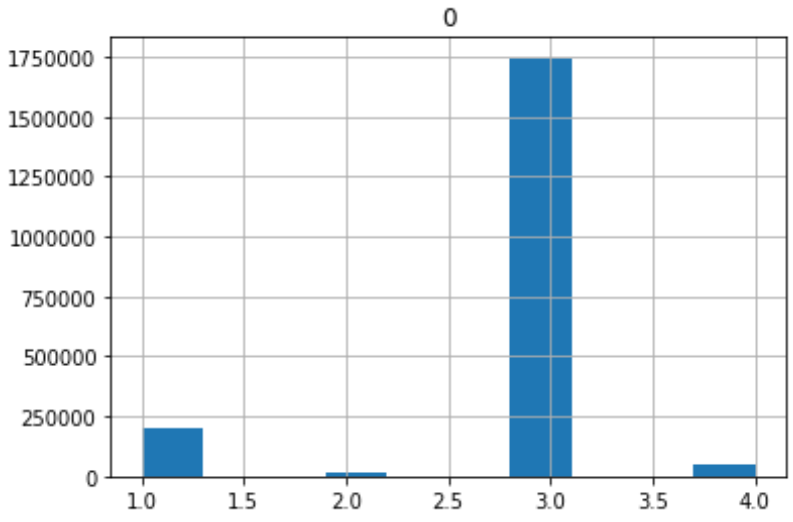
- Then the next step was to check whether there was scope for any dimensionality reduction. To check that implementing PCA algorithm. By that I was able to come to a conclusion that feature X5, X6, I5 & I6 were not so significant. Hence they can be dropped.
- Then we have to normalize the data, to make ideal to be fit into a model.

The different models you tried - what worked and what did not

- As our dataset had only continuous values, and it was a supervised learning model hence I thought ANN or support vector classifier could be applied.
- But for the support vector classifier both the training and validation accuracy were lesser than what was achieved through the ANN.
- With lesser computation time and overfitting threats in mind, I decided to use only a three layers ANN. Through ANN the results were quite convincing as
  Training accuracy = 95 - 96%
  Validation accuracy = 86-87 %
  Which were quite descent according to a three layered network model?

Explain the basics of the model you used and how you measured the error

- To check my model performance and its ability to generalise, I have used the hold out validation technique. In which I have let 10% of the training data to be used as validation data.
- After trying different combinations of no. of hidden nodes in the neural network model, I was able to get best results with an 8-6-4 NN.
- The trickiest part when dealing NN is to choose the number of epochs and batch size such that the model generalises rather than memorizing.
  <u>Predicted output for test set</u>



How further could this have been improved and the code optimized for speed

- The training time can be further reduced by doing some more analysis on the no. of hidden nodes and layers in the neural network. Further the no. of epochs and batch size could also be altered further.
- In terms of prediction time, neural networks provide fast computation by their parallel processing properties. Hence may not much can be improved in that.
- For the sake of speed, under sampling could be performed instead of over sampling. But that would reduce the training data by a significant amount. Hence such an action might not be good for training and generalisation.

----------END----------