Abstract

The goal of this project was to use binary classification models to predict the type of cover type in different wilderness areas with good accuracy, where 1 class in my project mean Lodgepole Pine and 0 class mean not class 1. The data from UCI. I tried three different models (Decision Tree Classifier – Naïve Bayes – Random Forest Classifier) and I printed results of accuracy and confusion matrix.

Design

To classify cover types and reply to the initiating question, where to find Lodgepole Pine trees and how to detect them, the below steps will be followed:

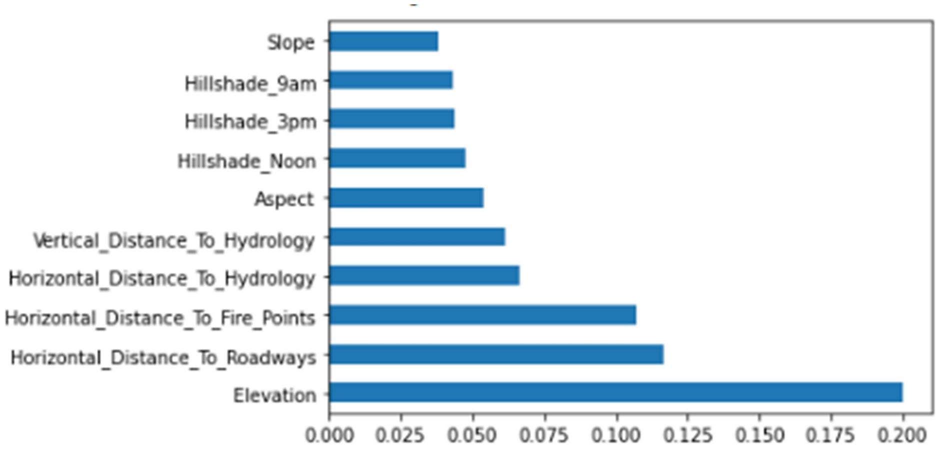
- Understand, Clean and Format Data
- Exploratory Data Analysis
- Feature Engineering & Selection
- Interpret Model Results
- Evaluate the Best Models with Test Data
- Comparison of models performance

Data

The data-set contain 581012 Instances and 54 Attributes, this data-set use for classification project, it is Categorical. It is labeled data-set and has 7 type of labels after chosen most top 10 features this features was the most important features ('Elevation','Horizontal_Distance_To_Roadways','Horizontal_Distance_To_Fire_Points','Horizontal_Distance_To_Hydrology', 'Vertical_Distance_To_Hydrology', 'Aspect', 'Hillshade_Noon', 'Hillshade_3pm', Hillshade_9am', 'Slope').

Algorithms

I converted my multi classification to binary classification I changed classes (1,3,4,5,6,7) to 0 and class 2 to 1 where is this class mean Lodgepole Pine Tree and 0 not this type of tree. I used Extra Trees Classifier to extract the most 10 important features in my data-set and I plot it in chart.



Then I split my data-set into training, validation and testing. 20% test data set and 30 validation data. And here is the results after splitting:

- Number of instances in the training dataset = 325366
- Number of instances in the validation dataset = 139443
- Number of instances in the test dataset = 116203

As I mentioned before I used three models:

- Decision Tree Classifier
- Naïve Bayes
- Random Forest Classifier

In Decision Tree I tried to find best tress size (16) and I got this results:

- Train accuracy   0.8933170644750834
- Validation accuracy   0.8619005615197608
- Test accuracy 0.863901964665284

Here are the results of Confusion Matrix for training:

- True Positives(TP) = 59585
- True Negatives(TN) = 60601
- False Positives(FP) = 12256
- False Negatives(FN) = 7001
- Classification accuracy for validation data set : 0.8619
- Classification error for validation data set : 0.1381
- Precision for validation data set 0.83
- Recall for validation data set 0.89

- F-Measure for validation data set 0.86

Here are the results of Confusion Matrix for testing:

- True Positives(TP) = 49345
- True Negatives(TN) = 51043
- False Positives(FP) = 10009
- False Negatives(FN) = 5806
- Classification accuracy for testing data set : 0.8639
- Classification error for testing data set : 0.1361
- Precision for testing data set 0.83
- Recall for testing data set 0.89
- F-Measure for testing data set 0.86

I tried Naïve Bayes here are the results:

- Validation accuracy 0.6533780828008577
- Test accuracy 0.6555424558746332

Here are the results of Confusion Matrix for validation:

- True Positives(TP) = 43990
- True Negatives(TN) = 47119
- False Positives(FP) = 27851
- False Negatives(FN) = 20483
- Classification accuracy for validation data set : 0.6534
- Classification error for validation data set : 0.3466
- Precision for validation data set 0.61
- Recall for validation data set 0.68
- F-Measure for validation data set 0.65

Here are the results of Confusion Matrix for testing:

- True Positives(TP) = 36504
- True Negatives(TN) = 39672

- False Positives(FP) = 22850
- False Negatives(FN) = 17177
- Classification accuracy for testing data set : 0.6555
- Classification error for testing data set : 0.3445
- Precision for testing data set 0.62
- Recall for testing data set 0.68
- F-Measure for testing data set 0.65

I tried Random Forest here are the results:

- Number of estimators used: 37
- Train accuracy 0.999941604224166
- Validation accuracy 0.9441492222628601
- Test accuracy 0.946266447509961

Here are the results of Confusion Matrix for validation:

- True Positives(TP) = 67599
- True Negatives(TN) = 63356
- False Positives(FP) = 4242
- False Negatives(FN) = 4246
- Classification accuracy for validation data set : 0.9391
- Classification error for validation data set : 0.0609
- Precision for validation data set 0.94
- Recall for validation data set 0.94
- F-Measure for validation data set 0.94

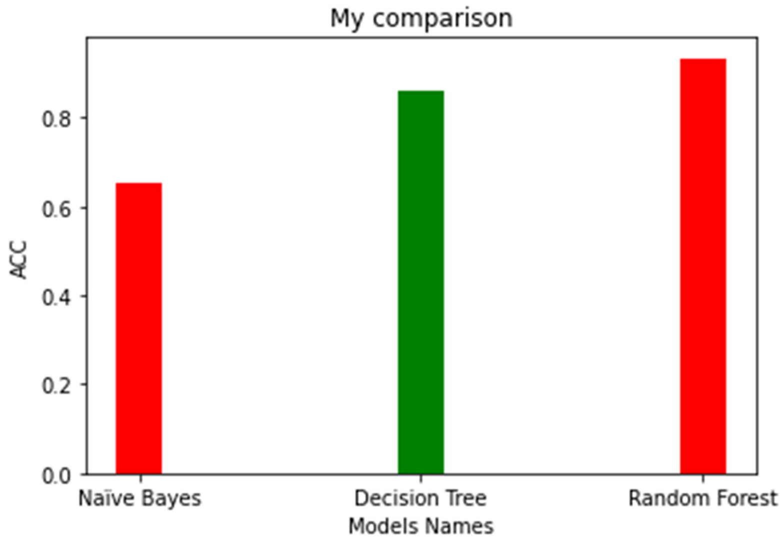
Here are the results of Confusion Matrix for testing:

- True Positives(TP) = 55882
- True Negatives(TN) = 53350
- False Positives(FP) = 3472
- False Negatives(FN) = 3499
- Classification accuracy for testing data set : 0.9400
- Classification error for testing data set : 0.0600
- Precision for testing data set 0.94
- Recall for testing data set 0.94
- F-Measure for testing data set 0.94

Then I applied 10 fold cross validation and here are the Cross-validation results:

- Decision Tree Classifier 0.8589434644696798
- Naïve Bayes 0.6546904025481888
- Random Forest Classifier 0.9334349606946521

Models comparison



Tools

Google Colab-Pandas-matplotlib-sklearn

Communication

I will present my code in presentation and for future work I can make multi classification cover type problem and to build it for android app.