**Final-Report**

**Outline: -**

* **Introduction**
* **Description of Dataset**
* **Description of the machine learning network and training algorithm**
* **Experimental setup**
* **Results**
* **Summary and conclusions**
* **References**
* **Appendix**

**Introduction**

Predicting forest cover types is an important task that can provide valuable information for land managers, conservationists, and scientists who are interested in understanding and managing forest ecosystems. By accurately predicting forest cover types, we can identify areas that are vulnerable to wildfires and other natural disasters, predict the impact of climate change on forest ecosystems, and identify areas that are suitable for timber harvesting or recreational use**.**

Accurate prediction of forest cover types can provide valuable information for land managers, conservationists, and scientists who are interested in understanding and managing forest ecosystems. For example, accurate predictions of forest cover types can help identify areas vulnerable to wildfires, predict the impact of climate change on forest ecosystems, and identify areas suitable for timber harvesting or recreational use.

Moreover, the dataset is also useful for evaluating the performance of machine learning algorithms and techniques for classification tasks. The dataset is large and contains a diverse range of features, including both quantitative and categorical variables, making it a challenging but realistic problem for machine learning models to solve.

In addition, the forest cover type prediction dataset has been widely used in research and education, making it a popular benchmark dataset for machine learning and remote sensing applications. Its availability and widespread use have helped to foster innovation and advance the state-of-the-art in machine learning and remote sensing.

**2. Description of Dataset:**

Our goal is to predict the forest cover type using some variables based on environmental conditions and soil types. We get two datasets available for this problem.

* UCI forest cover prediction dataset: This dataset is imbalanced, and it has around 581012 data points.

In this dataset we have,

* 10 Numerical features
* 44 Categorical features

**EDA:**

I worked on some of the numerical features and categorical features(soil features)

**There is no missing information(Null) in this dataset.**

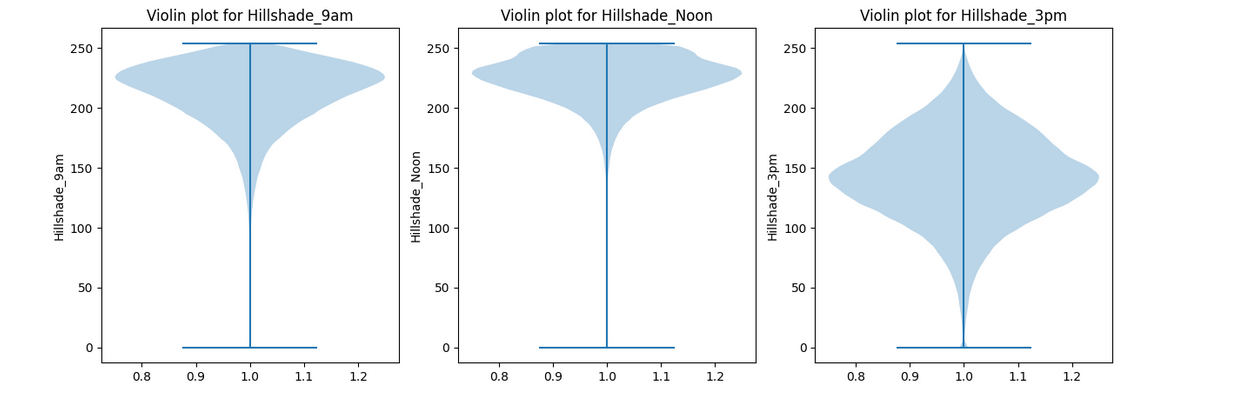
**Some of the features which I covered**

**Hill shade features:**

We have 3 features based on the hill shade values for 3 particular times during the day.

This feature is important as it will give us an idea that how much sunlight is available at that spot.

These values range from 0 to 255, so it is giving us the color shade between black and white. (0 is dark and 255 is white)

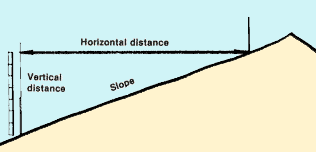


Based on the violin plot, hill shade 3am looks normal and hillshade\_9AM and Hillshade\_Noon are left skewed. As the range is fixed, we can say that there is no outlier present.

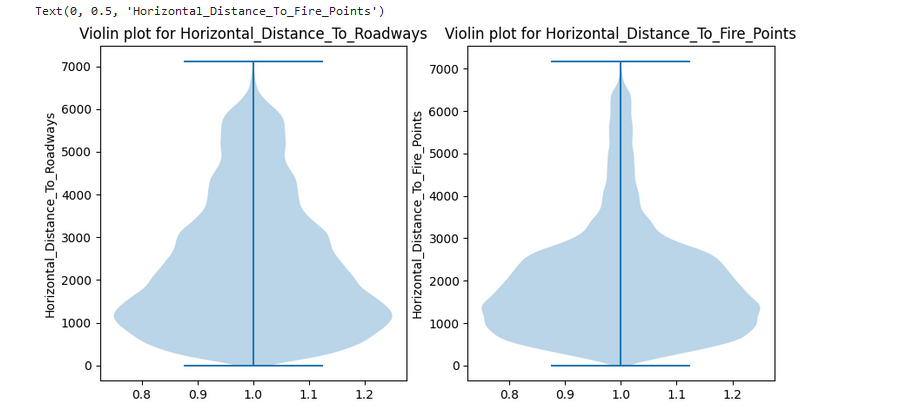
Horizontal distance to roadways and fire points:

Horizontal distance is the distance of the survey spot from the roadways and fire points.

Main impact of this feature is on the forest quality as if the locations are away from the roads and fire points, those spots are more secure and we will get better quality of forest cover as there will be less pollution and the probability of damage from the fire will be very less.



This feature is explaining the horizontal and vertical distance from the survey location to roadways



We can clearly see that these features are heavily right-skewed and the value range we have is from 0 to 7000.

To normalize that, we can apply log on the feature as it will shift the higher range values towards the left and we will get an approximate normal distribution.

**Feature Engineering on Numerical data:**

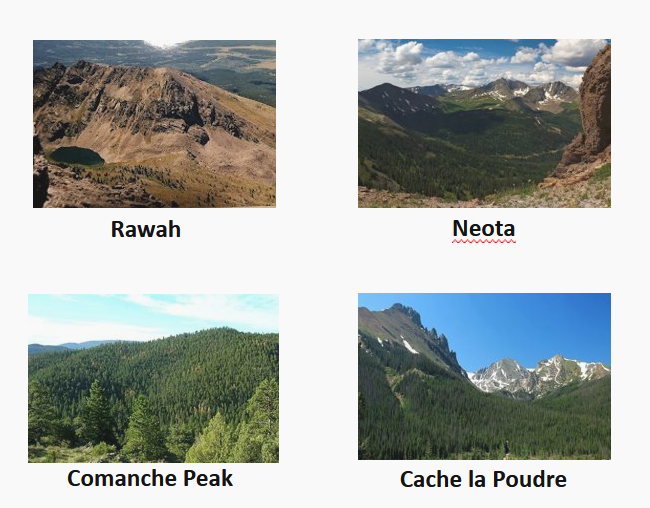
|  |  |
| --- | --- |
|  |  |

So we normalize the feature which was right skewed to approximate normal feature using sqrt transform.

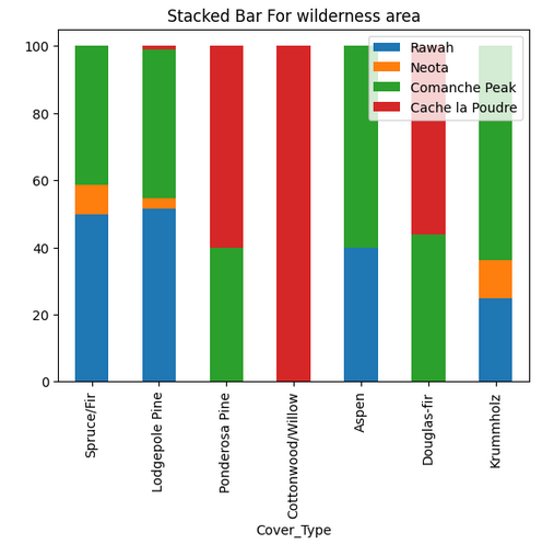
|  |  |
| --- | --- |
|  |  |

**Wilderness area:**

We have 4 features based on wilderness area type.



To find the impact of these features on the cover prediction, we mapped the percentage wise class of each wilderness area for particular class type(cover type).



**Based on this graph we can say that, Cache la paudre has mostly cottonwood/Willow cover type.**

**Most of the cover types are observed in wilderness area 3(Comanche peak)**

**Very small percentage of cover type observed in Neota.**

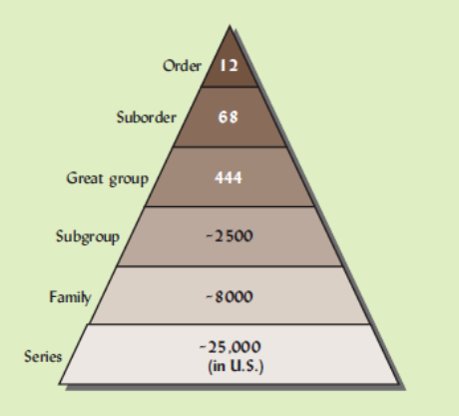
**Cover type 1,2,5 and 7 observed in Rawah.**

**Feature Engineering:**

**Soil Features:**

we have 40 different soil types with the soil texture description.

As we observed that some of the soil description is common, there must be some similarity in these soil types and we can find out the strong relations which are representing the given given types into strong groups.



**In case of soil classification, there are 12 main orders of the soil types.**

**Order** – Twelve soil orders are recognized. The differences among orders reflect the dominant soil forming processes and the degree of soil formation. Each order is identified by a word ending in 'sol.' An example is Alfisols.

**Suborder**- Each order is divided into suborders primarily on the basis of properties that influence soil formation and/or are important to plant growth.

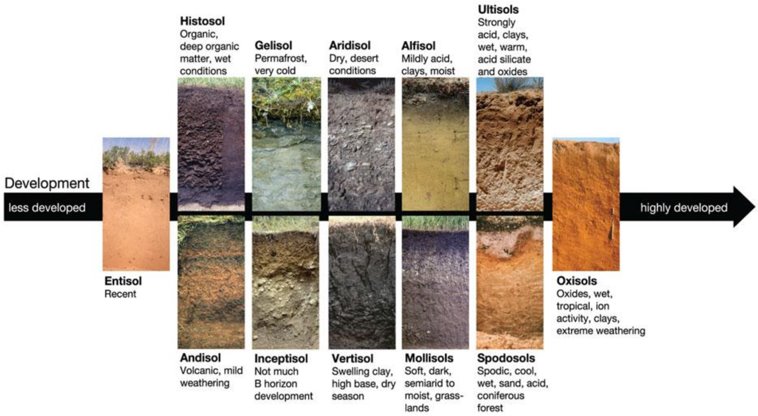
**Great Group** – Each suborder is divided into great groups on the basis of similarities in horizons present, soil moisture or temperature regimes, or other significant soil properties.

**Subgroup** – Each great group has a ‘typic’ (typical) subgroup which is basically defined by the Great Group. Other Subgroups are transitions to other orders, suborders, or great groups due to properties that distinguish it from the great group.

**Family** – Families are established within a subgroup on the basis of physical and chemical properties along with other characteristics that affect management.

**Series** – The series consists of soils within a family that have horizons similar in color, texture, structure, reaction, consistence, mineral and chemical composition, and arrangement in the profile.

The main 12 Soil orders are **Entisols, Inceptisols, Andisols, Mollisols, Alfisols, Spodosols, Ultisols, Oxisols, Gelisols, Histosols, Aridisols, and Vertisols**.



Each order is based on one or two dominants physical, chemical, or biological properties that differentiate it clearly from the other orders. Perhaps the easiest way to understand why certain properties were chosen over others is to consider how the soil (i.e., land) will be used.

As we have 40 different soil types in the dataset, it is really hard to manage this many features in the model and we can reduce the by classifying them into main parent classes.

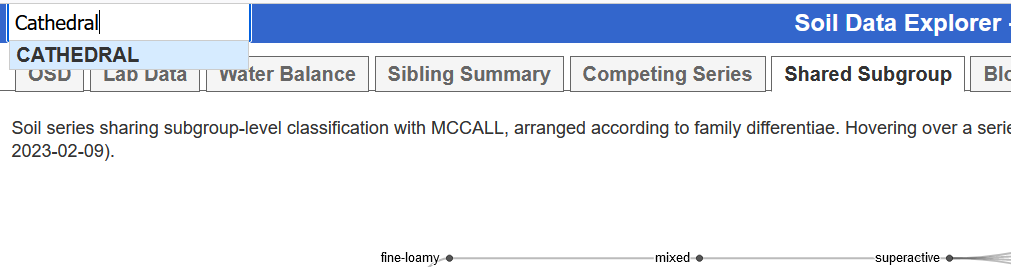
With the help of various survey sites like soilweb.com, we can easily find out the parent type of the given soil family. We can easily find out the order and sub group with the help of survey information.

Example:

Soil\_Type1 is from the Cathedral family.

search the family in

https://soilmap2-1.lawr.ucdavis.edu/sde/?series=mccall#shared-subgroup



Go to the Shared subgroup option for the hierarchical diagram

Diagram

Description automatically generated with medium confidence

In this tab we will get the main parent sub group of the soil family.

Diagram

Description automatically generated with medium confidence

Go to soil website and search for sub-group. We will get to see the main order of the soil.

Graphical user interface, application, Teams

Description automatically generated

Next, we will check if there is any strong relationship between soil order types and forest cover types.

**This is how** we got the mapping of 40 soil types into 6 major soil types. There is one type for which we do not have any information hence we categorize that as unknown.

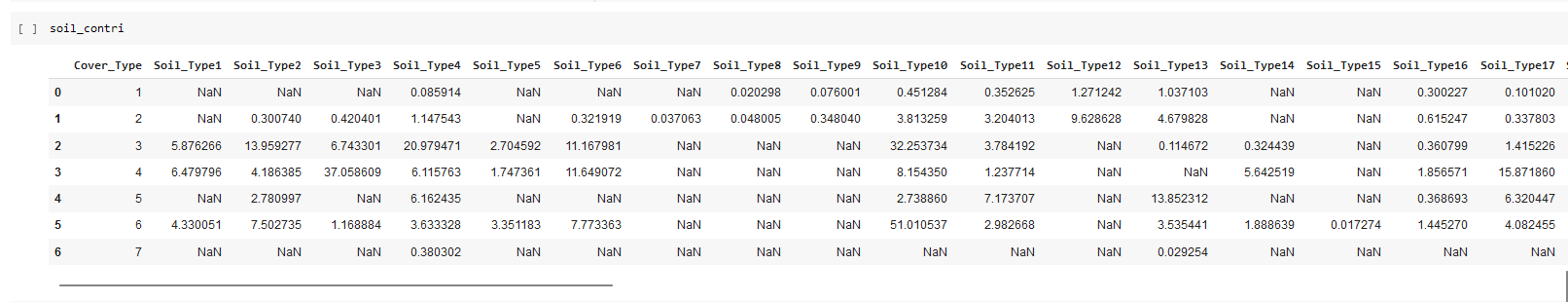


We are using the above method to map the order types to the respective soil type.

**Important soil types based on predicted forest classes:**

As per the soil information, it is observed that we have some soil types which are contributing more in the prediction of a particular soil type.

So, we find out the percentage-wise contribution of each soil type in each class prediction.



There are some soil types, which are helpful in predicting the forest cover type and we will include those features in our model

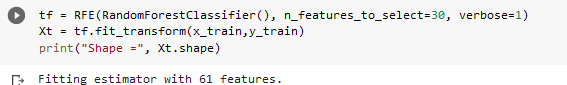
**Based on these observations, we can say that Soil\_Type29,Soil\_Type10,Soil\_Type3,Soil\_Type30,Soil\_Type38 are important in cover prediction.**

**Feature reduction:**

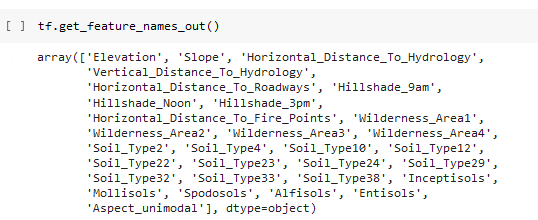
Used RFE to get the feature importance with random forest.

Random forest works really well on imbalanced data and it uses entropy or gini to evaluate the node and split it based on information gain.

So if the feature is used multiple times to split the nodes, that feature will the important feature.

****

**We got top 30 features using get\_feature\_names\_out() method.**

****

**3. Description of the machine learning network and training algorithm**

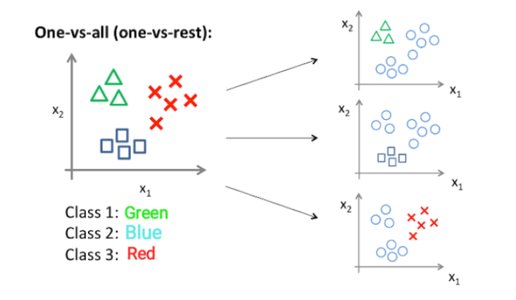
**Model Building:**

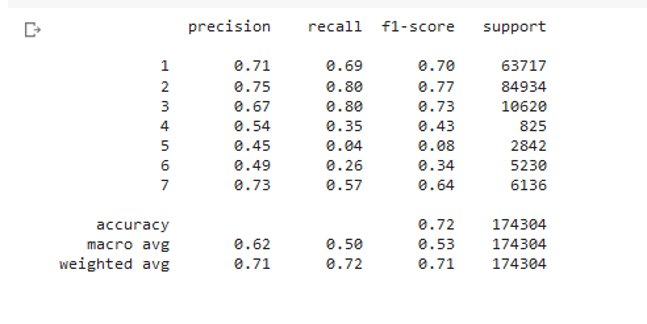
**Base model:**

**Logistic Regression for Multiclass:**

**Used the logistic regression using one vs all approach for multi-class.**

**It will create sub-models for each class and provides the output based on the higher probability of the class.**

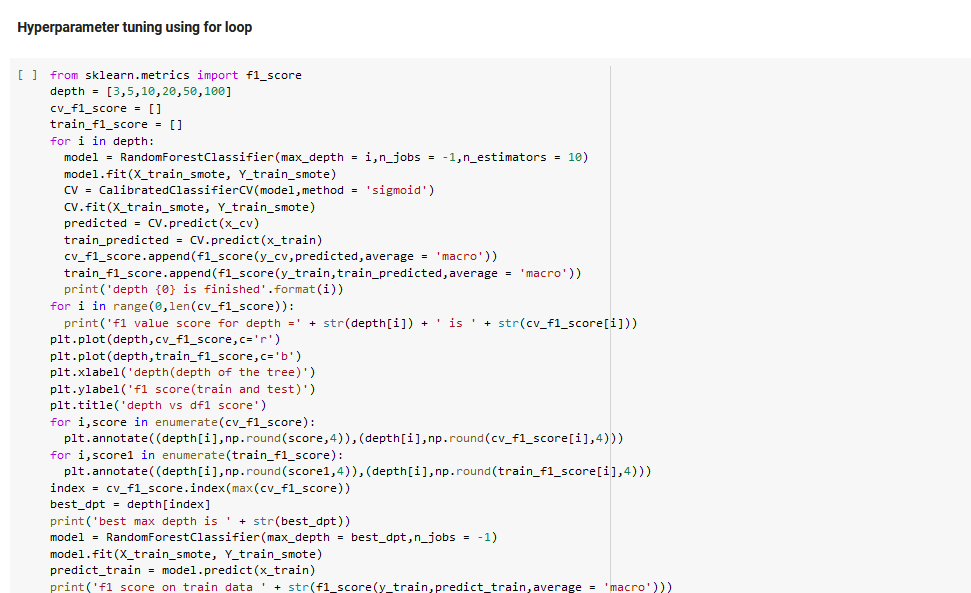
****

****

**Random forest with hyperparameter tuning:**

**As grid search and randomized cv search were taking too long, we decided to perform the hyperparameter tuning using for loop.**

**We passed the depth and number of estimators in for loop and check the F1 score for each model using CV and Test data.**

****

**We got F1 score around .92 which is a stable model.**

**MLP classifier**

We used a Grid Search method on the parameters mentioned below for the MLP Classifier

**Text

Description automatically generated**

After using the Grid search method, which was computationally expensive, we found the best parameters:

Text, letter

Description automatically generated

The below picture shows the classification report for the MLP classifier: -

Table

Description automatically generated

**XGBoost classifier**

We performed the hyperparameter tuning for the XGBoost classifier like what was done for the Random Forest classifier for the parameters like the number of estimators and depth. Based on many iterations, we found the best values of the number of estimators and depth values to be 100 which gave us the best F1 score of 0.93.

Text, timeline

Description automatically generated

A picture containing chart

Description automatically generated