Assignment 3 - Part 2

Link to questions - here

Student Details:

- Name : Anjishnu Mukherjee
- Registration Number : B00-5110170220
- Class Roll Number : CS Gy-70
- Exam Roll Number : 510517086
- Email : 5110170220.anjishnu@students.iiests.ac.in

Project Setup

Mount Google Drive

```
Mounted at /content/drive/
```

Load libraries, set seed

```
Environment Information :
-------------------------------------------
OS: Linux-4.19.112+-x86_64-with-Ubuntu-18.04-bionic
Python version: 3.6.9 (default, Jul 17 2020, 12:50:27)  [GCC 8.4.0]
Numpy version: 1.18.5
Pandas version: 1.0.5
Matplotlib version: 3.2.2
Seaborn version: 0.10.1
Scikitlearn version: 0.22.2.post1
-------------------------------------------
Global Seed = 5
-------------------------------------------
```

Forest Cover Type dataset

Link for Data

Read dataset into Pandas dataframe

Let's visualize the first 5 rows of the dataset.

	Elevation	Aspect	Slope	Horizontal_Distance_To_Hydrology	Vertical_Distance_To_Hydrology	Horizontal_Distance_To_Roadways	Hillshade_9am	Hillshade_Noon
0	2596	51	3	258	0	510	221	232
1	2590	56	2	212	-6	390	220	235
2	2804	139	9	268	65	3180	234	238
3	2785	155	18	242	118	3090	238	238
4	2595	45	2	153	-1	391	220	234

```
Data Dimensions: Rows(Records): 581012 Columns(Features): 55
```

Data information:

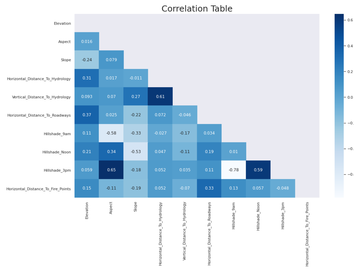
- `Elevation` = Height in meters.
- `Aspect` = Aspect in degrees azimuth.
- `Slope` = Slope in degrees.
- `Horizontal_Distance_To_Hydrology` = Horizontal distance to nearest surface water features.
- `Vertical_Distance_To_Hydrology` = Vertical distance to nearest surface water features.
- `Horizontal_Distance_To_Roadways` = Horizontal distance to nearest roadway.
- `Hillshade_9am` = Hill shade index at 9am, summer equinox. Value out of 255.
- `Hillshade_Noon` = Hill shade index at noon, summer equinox. Value out of 255.
- `Hillshade_3pm` = Hill shade index at 3pm, summer equinox. Value out of 255.
- `Horizontal_Distance_To_Fire_Points` = Horizontal distance to nearest wildfire ignition points.
- `Wilderness_Area1` = Rawah Wilderness Area
- `Wilderness_Area2` = Neota Wilderness Area
- `Wilderness_Area3` = Comanche Peak Wilderness Area
- `Wilderness_Area4` = Cache la Poudre Wilderness Area
- `Soil_Type1 to Soil_Type40` = Type of the soil.
- `Cover_TypeForest` = Cover type, integer value between 1 and 7, with the following key:

1. Spruce/Fir
2. Lodgepole Pine
3. Ponderosa Pine
4. Cottonwood/Willow
5. Aspen
6. Douglas-fir
7. Krummholz

Explore features to decide which ones to drop (EDA)

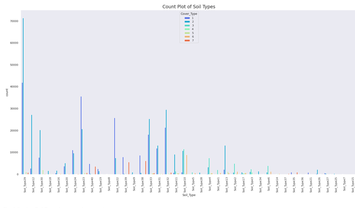
```
features = data.loc[:,'Elevation':'Horizontal_Distance_To_Fire_Points']
wilderness = data.loc[:,'Wilderness_Area1':'Wilderness_Area4']
soiltype = data.loc[:,'Soil_Type1':'Soil_Type40']
```

Correlation map of features



- `Hillshade_3pm` has very small negative correlation with `Horizontal_Distance_To_Fire_Points` . So, if we drop it, it shouldn't affect the model's predicting power by much.
- Also, none of the correlation values are greater than 0.7, so there is no need to drop the other features as they aren't highly correlated.

Count plot of Soil types

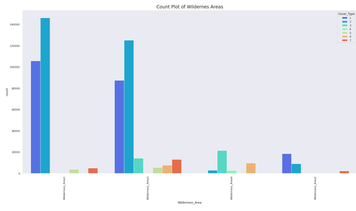


The following Soil Types have very low counts :

- `Soil_Type18`
- `Soil_Type16`
- `Soil_Type23`
- `Soil_Type8`
- `Soil_Type5`
- `Soil_Type28`
- `Soil_Type1`
- `Soil_Type17`
- `Soil_Type3`
- `Soil_Type34`
- `Soil_Type14`
- `Soil_Type37`
- `Soil_Type35`
- `Soil_Type36`
- `Soil_Type21`
- `Soil_Type24`
- `Soil_Type27`
- `Soil_Type25`
- `Soil_Type7`
- `Soil_Type15`

I will be dropping these to reduce computation, as these won't be too significant for the analysis anyway.

Count plot of Wilderness types



`Wilderness_Area2` and `Wilderness_Area4` have very low counts and thus won't be too relevant for the analysis compared to the other types.

Multi-class SVM using Linear Kernel

SVM in sklearn

Reduce the size first, otherwise it will take forever

Feature Selection

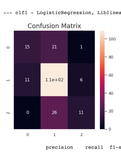
Train:Test split

```
-----------------------------------
The shapes of the different data : (n_samples,n_features)
x train:  (800, 31)
x test:  (200, 31)
y train:  (800,)
y test:  (200,)
-----------------------------------
```

Train the Linear SVM for the multi-class problem

- 7 classes
- 1000 records
- 31 features

```
--- clf1 - LogisticRegression, Liblinear solver, L2 penalty, One Versus Rest ---
```



	precision	recall	f1-score	support
1	0.58	0.41	0.48	37
2	0.70	0.87	0.77	126
3	0.61	0.30	0.40	37
accuracy			0.68	200
macro avg	0.63	0.52	0.55	200
weighted avg	0.66	0.68	0.65	200

```
--------------------------------------------------
Training accuracy : 68.62 [%]
Test accuracy :   67.50 [%]
```

Multi Class Logistic Regression - 2 features, 3 classes

Logistic Regression in sklearn

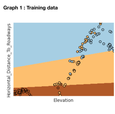
To prevent RAM from being completely filled up, I am choosing 50 rows of data for each of the 3 classes.

	Elevation	Aspect	Slope	Horizontal_Distance_To_Hydrology	Vertical_Distance_To_Hydrology	Horizontal_Distance_To_Roadways	Hillshade_9am	Hillshade_Noor
40	2699	347	3	0	0	2096	213	23
51	2739	323	25	85	43	3118	149	202
52	2696	72	2	30	0	3071	222	23
55	2722	315	24	30	19	3216	148	21
67	2919	13	13	90	6	5321	207	21

```
Data Dimensions: Rows(Records): 150 Columns(Features): 32

-----------------------------------
The shapes of the different data : (n_samples,n_features)
x train:  (375, 2)
x test:  (150, 2)
y train:  (225,)
y test:  (75,)
-----------------------------------
-----------------------------------
```

Graph 1 : Training data



Graph 2 : Test data