PROJECT

Wine data set

Problem Definition:

the given wine dataset is related to red, white and many other types of wines. the wine da ta set can be viewed as regression or classification tasks.
The given data has the following columns

1) fixed acidity
2) volatile acidity
3) citric acid
4) residual sugar
5) chlorides
6) free sulfur dioxide
7) total sulfur dioxide
8) density
9) pH
10) sulphates
11) alcohol
12) quality

The above columns when mixed in different ratio and proportion gives different type and taste of wine. In these problems, we are going to divide the wine basically into three classes

Let us start with importing the necessary libraries .

In [1]:
```
import warnings
warnings.simplefilter("ignore")
import seaborn as sn
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
import pandas as pd
df=pd.read_csv("winedataset.csv")
print(df)
```

Data Analysis

	Class	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	\
0	1	14.23	1.71	2.43	15.6	127	
1	1	13.20	1.78	2.14	11.2	100	
2	1	13.16	2.36	2.67	18.6	101	
3	1	14.37	1.95	2.50	16.8	113	
4	1	13.24	2.59	2.87	21.0	118	
..	...	...	...	...	...	...	
173	3	13.71	5.65	2.45	20.5	95	
174	3	13.40	3.91	2.48	23.0	102	
175	3	13.27	4.28	2.26	20.0	120	
176	3	13.17	2.59	2.37	20.0	120	
177	3	14.13	4.10	2.74	24.5	96	

	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	\
0	2.80	3.06	0.28	2.29	
1	2.65	2.76	0.26	1.28	
2	2.80	3.24	0.30	2.81	
3	3.85	3.49	0.24	2.18	
4	2.80	2.69	0.39	1.82	
..	...	...	...	...	
173	1.68	0.61	0.52	1.06	
174	1.80	0.75	0.43	1.41	
175	1.59	0.69	0.43	1.35	
176	1.65	0.68	0.53	1.46	
177	2.05	0.76	0.56	1.35	

	Color intensity	Hue	diluted wines	Proline
0	5.64	1.04	3.92	1065
1	4.38	1.05	3.40	1050
2	5.68	1.03	3.17	1185
3	7.80	0.86	3.45	1480
4	4.32	1.04	2.93	735
..	...	...	...	...
173	7.70	0.64	1.74	740
174	7.30	0.70	1.56	750
175	10.20	0.59	1.56	835
176	9.30	0.60	1.62	840

The given data has 178 rows and 14 columns

The columns are:

1) Class
2) Alcohol
3) Malic acid
4) Ash
5) Alcalinity of ash
6) Magnesium
7) Total phenols
8) Flavanoids
9) Nonflavanoid phenols
10) Proanthocyanins
11) Color intensity
12) Hue
13) diluted wines
14) Proline

Datatypes are:

Integer datatype:

1)Class
2)Magnesium
3) Proline

float datatype:

1)Alcohol
2)Malic acid
3)Ash Alcalinity of ash
4)Total phenols
5)Flavanoids
6)Nonflavanoid phenols
7)Proanthocyanins
8)Color intensity
9) Hue diluted

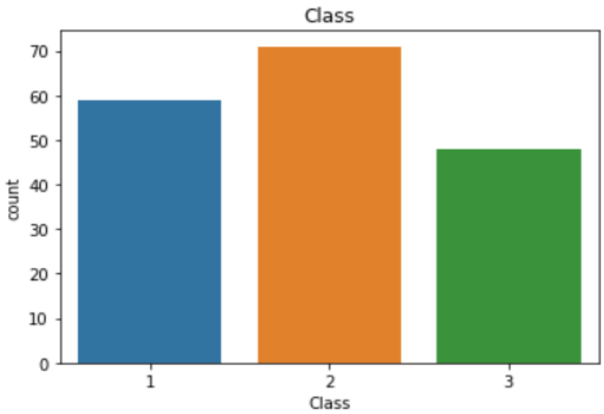
Null values: there are no null values in the given data
Missing values: there are no missing or Nan values in the given datatype
Target variable: our target variable in the given data will be class

EDA:

UNIVARIANT ANALYSIS:

CLASS: there are 3 types of class
Class1, class2 and class3

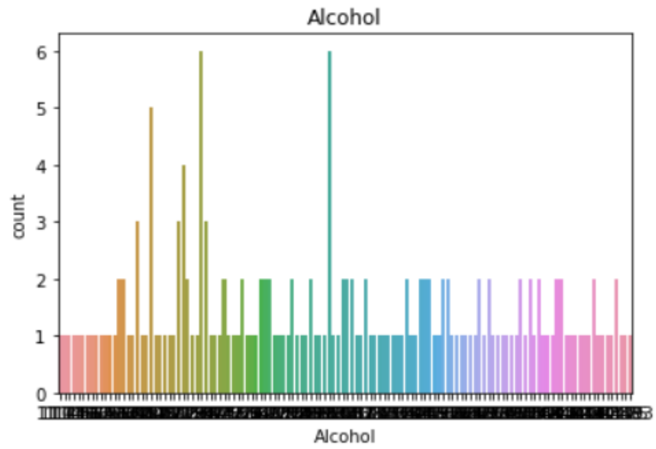
In [52]:
```
sn.countplot(df["Class"])
plt.title("Class")
plt.show()
```



Class 2 have the highest count around 70,
class 3 have the least count around 50 and class 1 have medium count around 60

Alcohol:

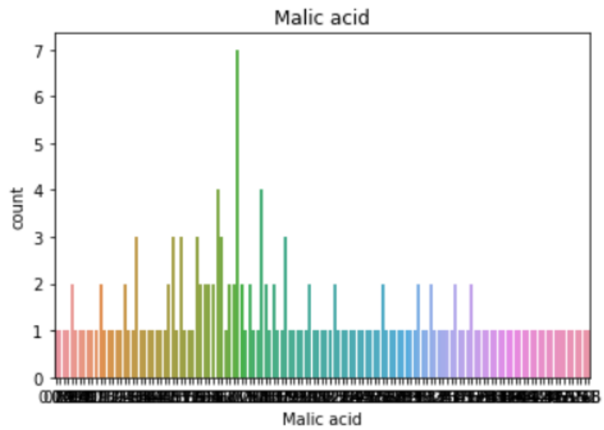
```
[53]: sn.countplot(df["Alcohol"])
      plt.title("Alcohol")
      plt.show()
```



The highest count of alcohol is 6 and the least count of alcohol is 1. Average count is alcohol is 1.

Malic acid :

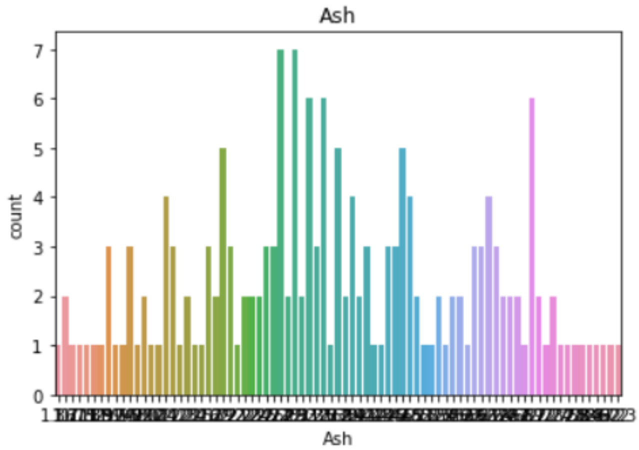
```
In [54]: sn.countplot(df["Malic acid"])
         plt.title("Malic acid")
         plt.show()
```



The highest count of Malic acid is 7 and the least count is 1. the average count is also 1

Ash:

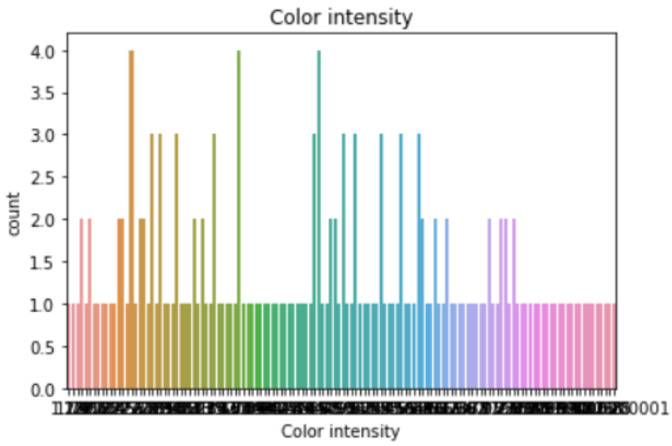
```
In [55]: sn.countplot(df["Ash"])
plt.title("Ash")
plt.show()
```



The highest count of the Ash is 7 and the least count is 1. the average count is also 1

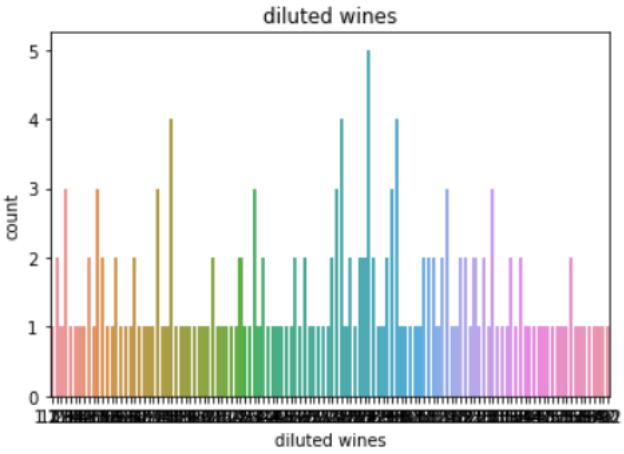
COLOR INTENSITY:

```
In [56]: sn.countplot(df["Color intensity"])
plt.title("Color intensity")
plt.show()
```



The highest count of the Ash is 4 and the least count is 1. the average count is also

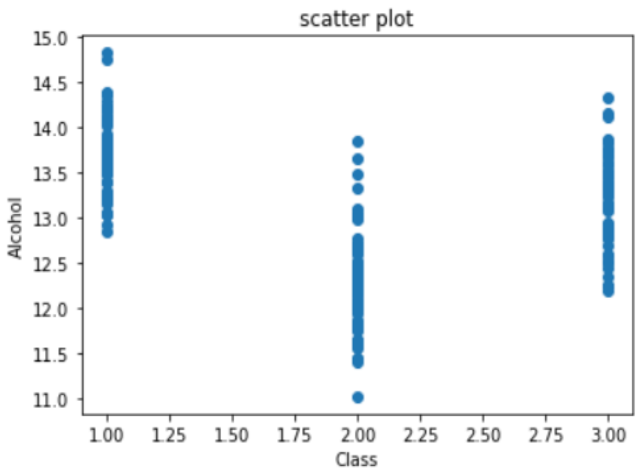
diluted wines:



The highest count of the Ash is 5 and the least count is 1. the average count is also 1

BIVARIANT ANALYSIS:

Between alcohol and class

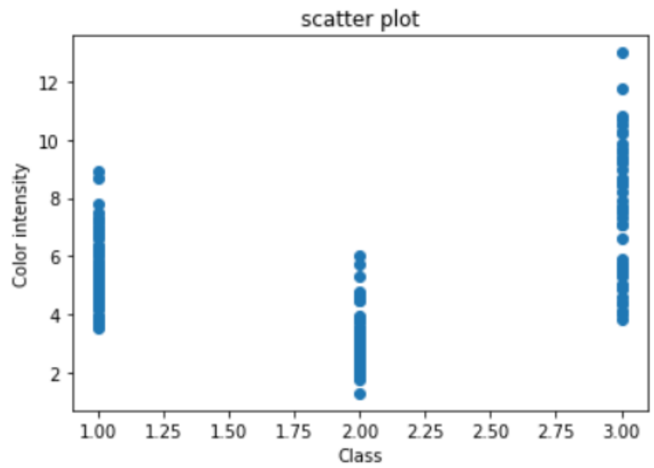


The range of alcohol in class 1 is from 12.5 – 15

The range of alcohol in class 2 is from 11- 14

The range of alcohol in class 3 is from 12- 14.5

Between color intensity and class:

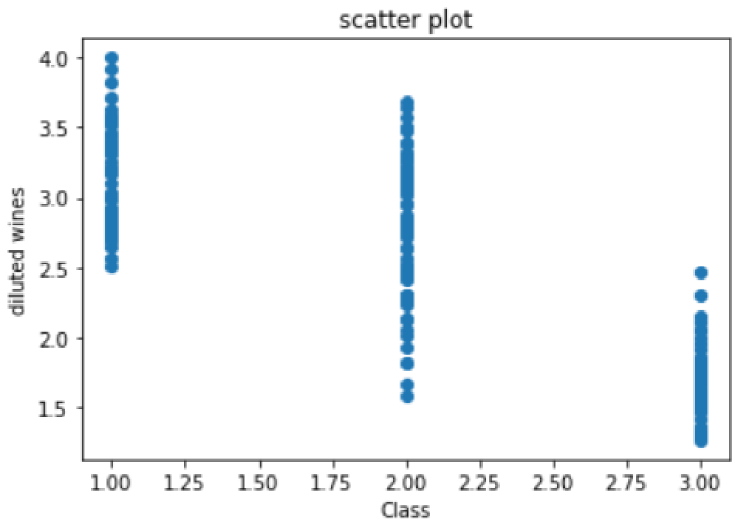


Color intensity in class 1 is range between 5.5 - 9

Color intensity is most in the class 3 wines range between 4 -13

Color intensity is least in class 2 wines range from around 0.5 - 5

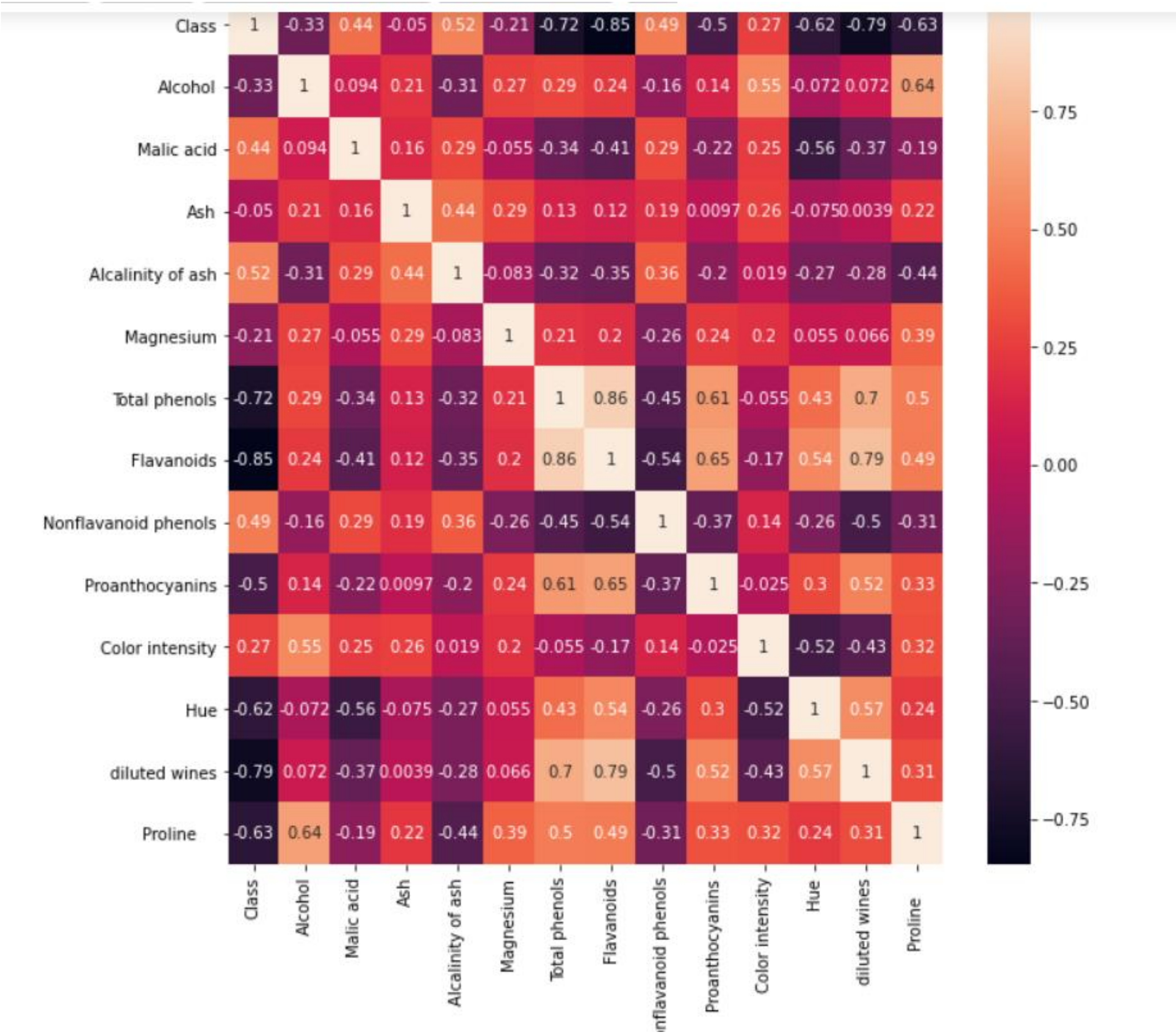
Between diluted wines and class:



We can see that most of the diluted wines belongs to class 2

And least of the diluted wines belongs to the class 3

MULTIVARIANT ANALYSIS:



Negative correlation: Class has negative correlation with alcohol, ash, magnesium, total phenols, flavonoids, proanthocyanins, hue, diluted wines and proline

Positive correlation: Class has positive correlation with malic acid, alkalinity of ash, nonflavanoid phenols and color intensity

Good correlation: class has good correlation with alcohol, magnesium, total phenols, flavon oids, hue, diluted wines and proline, malic acid, alkalinity of ash, nonflavanoid phenols and c olor intensity
No so good correlation: class has not so good correlation with ash and proanthocyanins

PRE-PROSSECING PIPELINE:

```
]: df.describe()
```

]:

	Class	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity	Hu
count	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.00000
mean	1.938202	13.000618	2.336348	2.366517	19.494944	99.741573	2.295112	2.029270	0.361854	1.590899	5.058090	0.9574
std	0.775035	0.811827	1.117146	0.274344	3.339564	14.282484	0.625851	0.998859	0.124453	0.572359	2.318286	0.2285
min	1.000000	11.030000	0.740000	1.360000	10.600000	70.000000	0.980000	0.340000	0.130000	0.410000	1.280000	0.4800
25%	1.000000	12.362500	1.602500	2.210000	17.200000	88.000000	1.742500	1.205000	0.270000	1.250000	3.220000	0.7825
50%	2.000000	13.050000	1.865000	2.360000	19.500000	98.000000	2.355000	2.135000	0.340000	1.555000	4.690000	0.9650
75%	3.000000	13.677500	3.082500	2.557500	21.500000	107.000000	2.800000	2.875000	0.437500	1.950000	6.200000	1.1200
max	3.000000	14.830000	5.800000	3.230000	30.000000	162.000000	3.880000	5.080000	0.660000	3.580000	13.000000	1.7100

]:

ohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity	Hue	diluted wines	Proline
000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000	178.000000
618	2.336348	2.366517	19.494944	99.741573	2.295112	2.029270	0.361854	1.590899	5.058090	0.957449	2.611685	746.893258
827	1.117146	0.274344	3.339564	14.282484	0.625851	0.998859	0.124453	0.572359	2.318286	0.228572	0.709990	314.907474
000	0.740000	1.360000	10.600000	70.000000	0.980000	0.340000	0.130000	0.410000	1.280000	0.480000	1.270000	278.000000
500	1.602500	2.210000	17.200000	88.000000	1.742500	1.205000	0.270000	1.250000	3.220000	0.782500	1.937500	500.500000
000	1.865000	2.360000	19.500000	98.000000	2.355000	2.135000	0.340000	1.555000	4.690000	0.965000	2.780000	673.500000
500	3.082500	2.557500	21.500000	107.000000	2.800000	2.875000	0.437500	1.950000	6.200000	1.120000	3.170000	985.000000
000	5.800000	3.230000	30.000000	162.000000	3.880000	5.080000	0.660000	3.580000	13.000000	1.710000	4.000000	1680.000000

Key Observations

1) the mean is almost equal to median (50 percentile) in all columns.

2) there is large difference between the max and 75 percentiles in Alcalinity of ash, Flavanoids, Color intensity and Proline

3) the above points 1 and 2 suggest that there are extreme outliers present in these four columns.

Removing outliers: as analysis above there are outliers present in the given data, we need to remove the outliers using zscore method.

Converting object values to int: There are no object values to convert to integers.

Removing skewness: it is very important to remove skewness to avoid any kind of over fitting or under fitting, to remove the skewness we will use StandardScaler and fit method.

Now we will split the data into two parts

1) train data
2) test data

the split of our train data and test data will be 25% and 75%

NOW FINDING THE BEST MODEL:

Linear Regression:

we found Accuracy=100.0, cross validation score=100.0 & difference =0.0

Random Forest Regressor:
Accuracy=99.99453551912568, cross validation score =99.97087372216639 & difference =0.023661796959288495

Ada Boost Regressor:
Accuracy=100.0, cross validation score=97.87336443586445 & difference =2.1266355641355545

SGD Regressor:
Accuracy=94.11001904585422, cross validation score =94.95559601455462 & difference =-0.8455769687003993

RESULT:
Linear Regression and Ada Boost Regressor are performing the best with same accuracy and cross validation score. I will choose Linear Regression.

Agriculture dataset
PROBLE DEFINITION:

Int the given agriculture dataset types of crops are given , we need to analyze the crop damage depending on different attributes

The columns are

1) ID

2) Estimated_Insects_Count
3) Crop_Type
4) Soil_Type
5) Pesticide_Use_Category
6) Number_Doses_Week
7) Number_Weeks_Used
8) Number_Weeks_Quit
9) Season
10)Crop_Damage

Let us start with importing the necessary libraries.

In [1]:
```
import warnings
warnings.simplefilter("ignore")
import seaborn as sn
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
import pandas as pd
df=pd.read_csv("winedataset.csv")
print(df)
```

Data Analysis
Problem Definition:

```
              ID  Estimated_Insects_Count  Crop_Type  Soil_Type  \
0      F00000001                      188          1          0
1      F00000003                      209          1          0
2      F00000004                      257          1          0
3      F00000005                      257          1          1
4      F00000006                      342          1          0
...          ...                      ...        ...        ...
88853  F00155935                     3337          1          0
88854  F00155938                     3516          1          0
88855  F00155939                     3516          1          0
88856  F00155942                     3702          1          0
88857  F00155945                     3895          1          0

       Pesticide_Use_Category  Number_Doses_Week  Number_Weeks_Used  \
0                           1                  0                0.0
1                           1                  0                0.0
2                           1                  0                0.0
3                           1                  0                0.0
4                           1                  0                0.0
...                       ...                ...                ...
88853                       2                 10               12.0
88854                       2                 10               20.0
88855                       2                 15               40.0
88856                       2                 10               25.0
88857                       2                 20               37.0

       Number_Weeks_Quit  Season  Crop_Damage
0                      0       1            0
1                      0       2            1
2                      0       2            1
3                      0       2            1
4                      0       2            1
...                  ...     ...          ...
88853                 44       3            0
88854                 38       1            0
88855                  8       2            0
88856                 18       3            0
88857                  7       3            0
```

The given data has 88858 rows and 10 columns

The columns are :
1) ID
2) Estimated_Insects_Count
3) Crop_Type
4) Soil_Type
5) Pesticide_Use_Category
6) Number_Doses_Week
7) Number_Weeks_Used
8) Number_Weeks_Quit
9) Season
10)Crop_Damage

Datatypes:
Integer datatype:
1)Estimated_Insects_Count
2)Crop_Type
3)Soil_Type
4)Pesticide_Use_Category
5)Number_Doses_Week
6)Number_Weeks_Quit
7)Season
8)Crop_Damage

float datatype: ID
object datatype: Number_Weeks_Used

Null values: there are no null values in the given data
Missing values: there are missing values in the given datatype

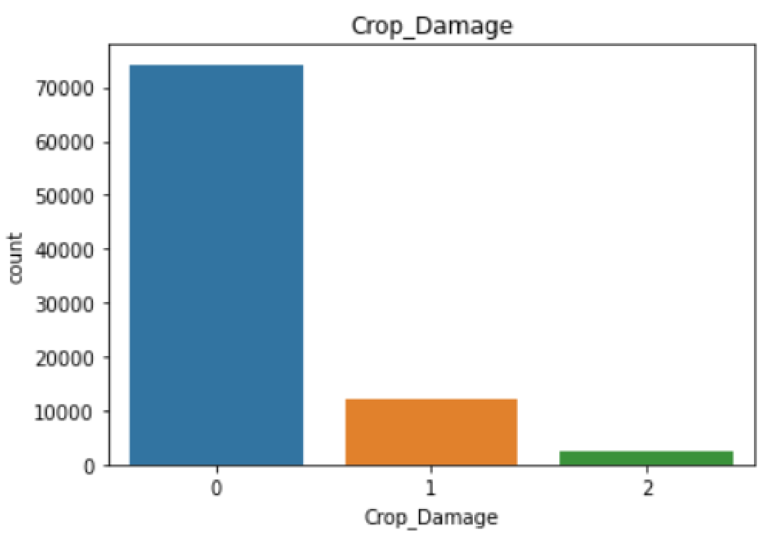
```
In [10]: df=df.replace(np.NaN,df['Number_Weeks_Used'].mean())
```

Using replace method we can fill the Nan values by mean

Target variable: our target variable in the given data will be Average Price

EDA:

UNIVARIANT ANALYSIS:
Crop damage:

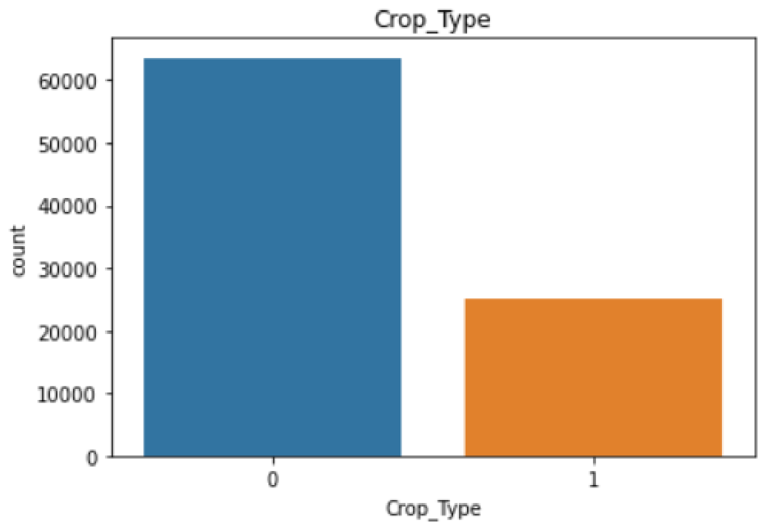


Crop type 0 has highest count of crop damage

Crop type 2 has least count of crop damage

Crop type 1 has crop damage around 10000.

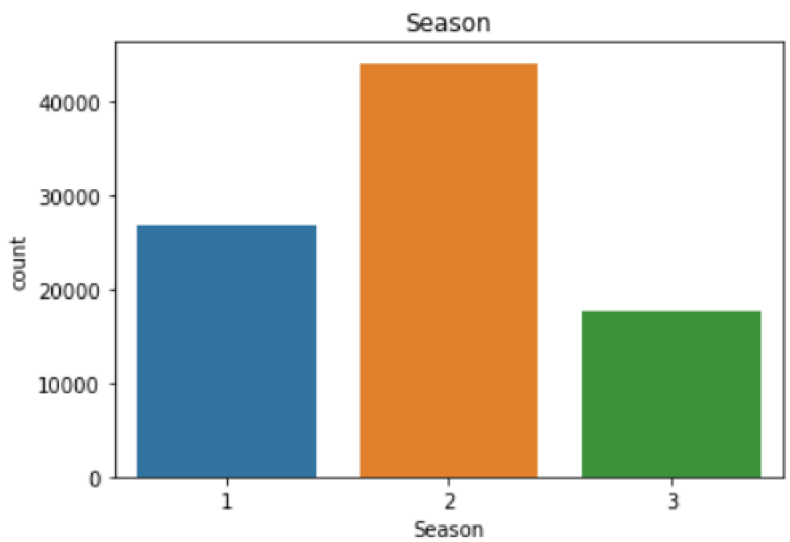
CROP TYPE:



Crop type 0 has the count of 60000

Crop type 1 has the count 25000

SEASON:



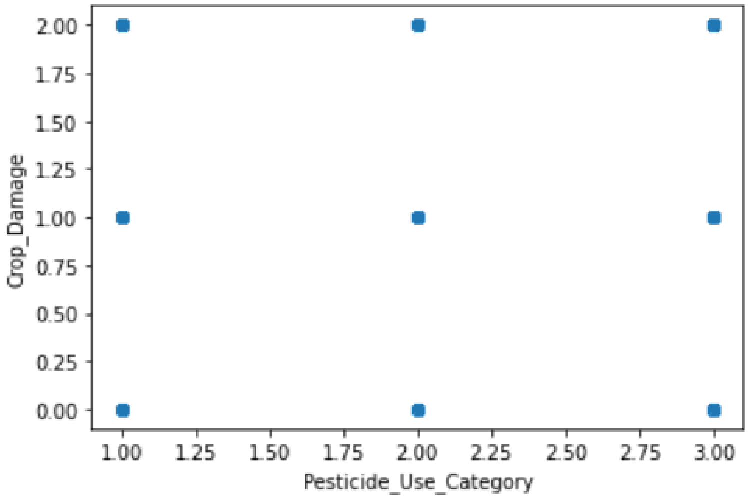
Season 1 has count of around 28000

Season 2 has count of around 45000

Season 3 has count of around 18000

BIVARIANT ANALYSIS:

Between crop damage and pesticide use category

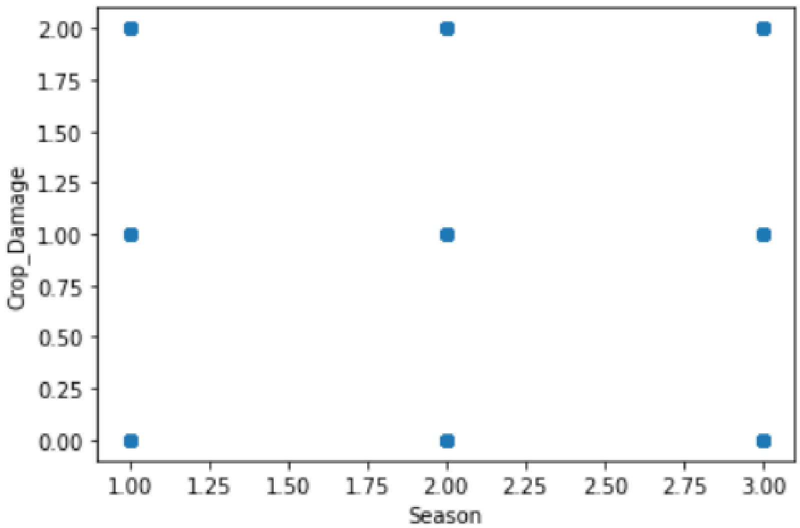


Pesticide category 1 has caused all three type of crop damage

Pesticide category 2 has caused all three type of crop damage

Pesticide category 3 has caused all three type of crop damage

Between crop damage and season

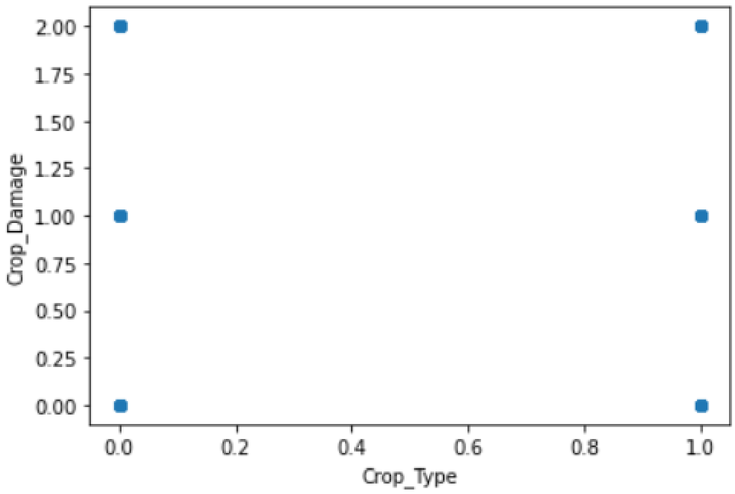


Season 1 has caused all three type of crop damage

Season 2 has caused all three type of crop damage

Season 3 has caused all three type of crop damage

Between crop damage crop type

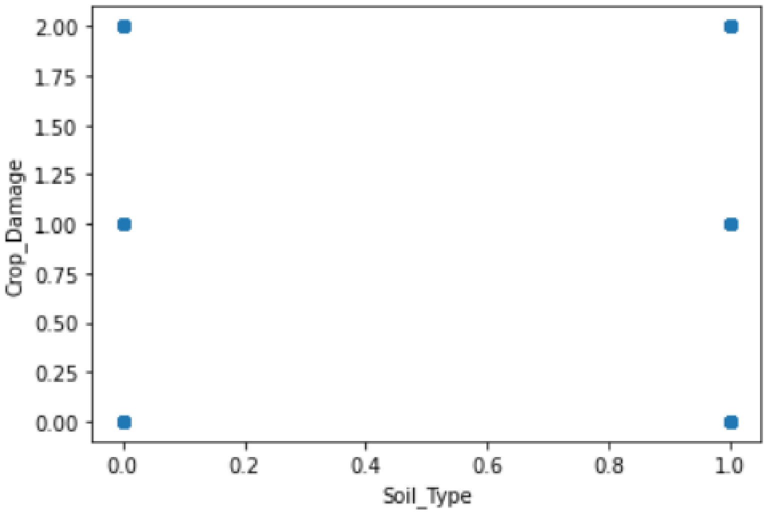


Crop type 1 has caused all three type of crop damage

Crop type 2 has caused all three type of crop damage

Crop type 3 has caused all three type of crop damage

Between crop damage and soil type

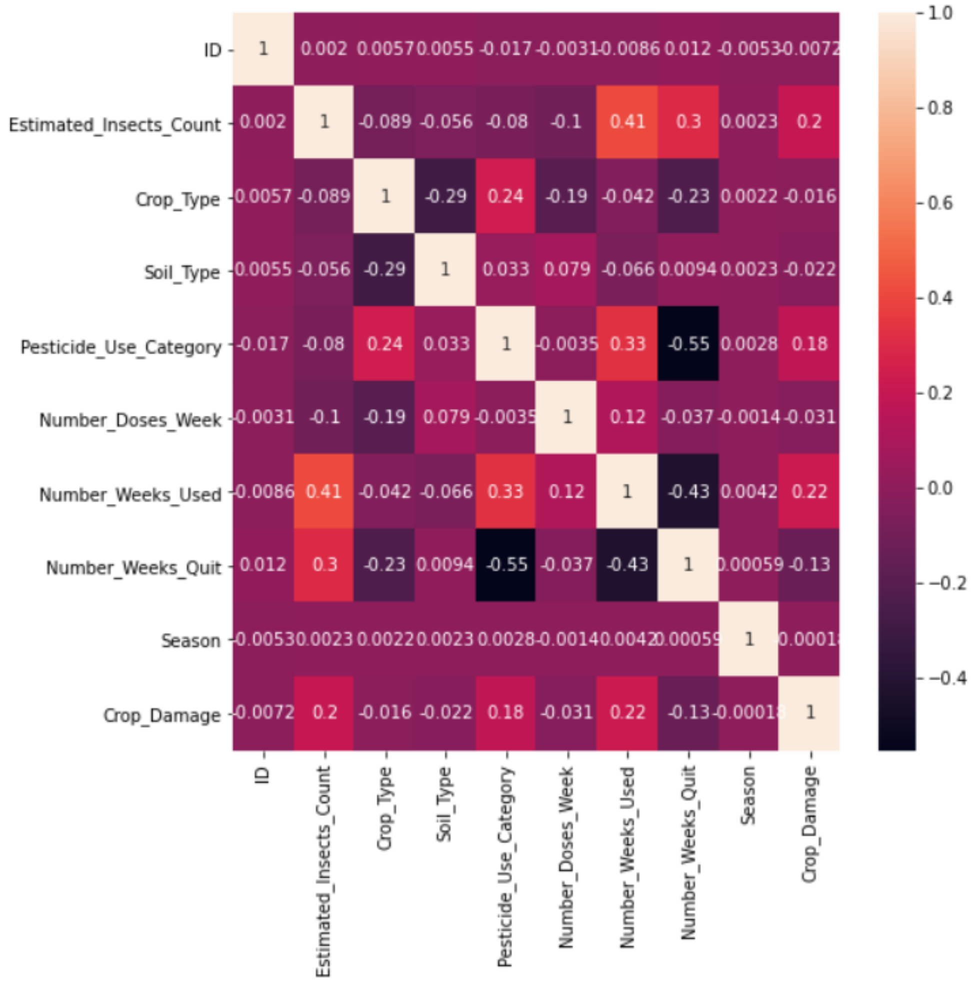


soil type 1 has caused all three type of crop damage

soil type 2 has caused all three type of crop damage

soil type 3 has caused all three type of crop damage

MULTIVARIANT ANALYSIS:



1) ID
2) Estimated_Insects_Count
3) Crop_Type
4) Soil_Type
5) Pesticide_Use_Category
6) Number_Doses_Week
7) Number_Weeks_Used
8) Number_Weeks_Quit
9) Season
10)Crop_Damage

Crop damage has positive correlation with with : ID , Estimated_Insects_Count, Pesticide_Use_Category, Number_Weeks_Used and season

Crop damage has negative correlation with: Crop_Type, Soil_Type, Number_Doses_Week and Number_Weeks_Quit

Crop damage has good correlation with: ID, Estimated_Insects_Count, Pesticide_Use_Category, Number_Weeks_Used and Number_Weeks_Quit

Crop damage has not so good correlation with: Crop_Type, Soil_Type, Number_Doses_Wee k and season

PRE-PROSSECING PIPELINE :
Removing outliers :

]:

	Estimated_Insects_Count	Crop_Type	Soil_Type	Pesticide_Use_Category	Number_Doses_Week	Number_Weeks_Used	Number_Weeks_Quit	
count	88858.000000	88858.000000	88858.000000	88858.000000	88858.000000	79858.000000	88858.000000	8
mean	1399.012210	0.284375	0.458417	2.264186	25.849952	28.623970	9.589986	
std	849.048781	0.451119	0.498271	0.461772	15.554428	12.391881	9.900631	
min	150.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	
25%	731.000000	0.000000	0.000000	2.000000	15.000000	20.000000	0.000000	
50%	1212.000000	0.000000	0.000000	2.000000	20.000000	28.000000	7.000000	
75%	1898.000000	1.000000	1.000000	3.000000	40.000000	37.000000	16.000000	
max	4097.000000	1.000000	1.000000	3.000000	95.000000	67.000000	50.000000	

ects_Count	Crop_Type	Soil_Type	Pesticide_Use_Category	Number_Doses_Week	Number_Weeks_Used	Number_Weeks_Quit	Season	Crop_Damage
858.000000	88858.000000	88858.000000	88858.000000	88858.000000	79858.000000	88858.000000	88858.000000	88858.000000
399.012210	0.284375	0.458417	2.264186	25.849952	28.623970	9.589986	1.896959	0.190562
849.048781	0.451119	0.498271	0.461772	15.554428	12.391881	9.900631	0.701322	0.454215
150.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	1.000000	0.000000
731.000000	0.000000	0.000000	2.000000	15.000000	20.000000	0.000000	1.000000	0.000000
212.000000	0.000000	0.000000	2.000000	20.000000	28.000000	7.000000	2.000000	0.000000
898.000000	1.000000	1.000000	3.000000	40.000000	37.000000	16.000000	2.000000	0.000000
097.000000	1.000000	1.000000	3.000000	95.000000	67.000000	50.000000	3.000000	2.000000

Key Observations

1) the mean is greater than standard deviation in all columns.

2)there is large difference between the max and 75 percentiles in Estimated_Insects_Cou nt, Number_Doses_Week, Number_Weeks_Used and Number_Weeks_Quit

3) the above points 1 and 2 suggest that there are extreme outliers present in these four columns.

Removing outliers: as analyzed above there are outliers present in the given data, we need to remove the outliers using Zscore method.

Converting object values to int: There are object values present which need to be converted into integers.

```
df=df.replace(np.NaN,df['Number_Weeks_Used'].mean())
```

Using replace method we have convert object values into integer values .

Removing skewness: it is very important to remove skewness to avoid any kind of over fitting or under fitting, to remove the skewness we will use StandardScaler and fit method.

Now we will split the data into two parts

3) train data
4) test data

the split of our train data and test data will be 55% and 45%

Target variable: our target variable will be crop damage

NOW FINDING THE BEST MODEL:

Linear Regression:
we found Accuracy=100.0, cross validation score=100.0 & difference =0.0

Random Forest Regressor:
Accuracy=100.0, cross validation score =100.0 & difference =0.0

Ada Boost Regressor:
Accuracy=100.0, cross validation score =100.0 & difference =0.0

SGD Regressor:
Accuracy=99.99999890544973, cross validation score =99.99999885227527 & difference =5.317446039043716e-08

RESULT:
Linear Regression, Random Forest Regressor and Ada Boost Regressor are performing the best with same accuracy and cross validation score. I will choose Linear Regression.