


## Application 8

# Supervised Machine Learning

## Wine Predictor using K Nearest Neighbour Algorithm

There is one data set of wine which classify the wines according to its contents into three classes.

Consider below Wine Dataset as



Class	Alcohol	Malic acid	Ash	Alcalinity of ash	Magnesium	Total phenols	Flavanoids	Nonflavanoid phenols	Proanthocyanins	Color intensity	Hue	OD280/OD315 of diluted wines	Proline
1	14.23	1.71	2.43	15.6	127	2.8	3.06	0.28	2.29	5.64	1.04	3.92	1065
1	13.2	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.4	1050
1	13.16	2.36	2.67	18.6	101	2.8	3.24	0.3	2.81	5.68	1.03	3.17	1185
1	14.37	1.95	2.5	16.8	113	3.85	3.49	0.24	2.18	7.8	0.86	3.45	1480
1	13.24	2.59	2.87	21	118	2.8	2.69	0.39	1.82	4.32	1.04	2.93	735
1	14.2	1.76	2.45	15.2	112	3.27	3.39	0.34	1.97	6.75	1.05	2.85	1450
1	14.39	1.87	2.45	14.6	96	2.5	2.52	0.3	1.98	5.25	1.02	3.58	1290
1	14.06	2.15	2.61	17.6	121	2.6	2.51	0.31	1.25	5.05	1.06	3.58	1295
1	14.83	1.64	2.17	14	97	2.8	2.98	0.29	1.98	5.2	1.08	2.85	1045
1	13.86	1.35	2.27	16	98	2.98	3.15	0.22	1.85	7.22	1.01	3.55	1045
1	14.1	2.16	2.3	18	105	2.95	3.32	0.22	2.38	5.75	1.25	3.17	1510
1	14.12	1.48	2.32	16.8	95	2.2	2.43	0.26	1.57	5	1.17	2.82	1280
1	13.75	1.73	2.41	16	89	2.6	2.76	0.29	1.81	5.6	1.15	2.9	1320
1	14.75	1.73	2.39	11.4	91	3.1	3.69	0.43	2.81	5.4	1.25	2.73	1150
1	14.38	1.87	2.38	12	102	3.3	3.64	0.29	2.96	7.5	1.2	3	1547
1	13.63	1.81	2.7	17.2	112	2.85	2.91	0.3	1.46	7.3	1.28	2.88	1310
1	14.3	1.92	2.72	20	120	2.8	3.14	0.33	1.97	6.2	1.07	2.65	1280
1	13.83	1.57	2.62	20	115	2.95	3.4	0.4	1.72	6.6	1.13	2.57	1130
1	14.19	1.59	2.48	16.5	108	3.3	3.93	0.32	1.86	8.7	1.23	2.82	1680
1	13.64	3.1	2.56	15.2	116	2.7	3.03	0.17	1.66	5.1	0.96	3.36	845
1	14.06	1.63	2.28	16	126	3	3.17	0.24	2.1	5.65	1.09	3.71	780
1	12.93	3.8	2.65	18.6	102	2.41	2.41	0.25	1.98	4.5	1.03	3.52	770
1	13.71	1.86	2.36	16.6	101	2.61	2.88	0.27	1.69	3.8	1.11	4	1035
1	12.85	1.6	2.52	17.8	95	2.48	2.37	0.26	1.46	3.93	1.09	3.63	1015
1	13.5	1.81	2.61	20	96	2.53	2.61	0.28	1.66	3.52	1.12	3.82	845

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines.

Wine data set contains 13 features as

- 1) Alcohol
- 2) Malic acid
- 3) Ash
- 4) Alcalinity of ash
- 5) Magnesium
- 6) Total phenols
- 7) Flavanoids
- 8) Nonflavanoid phenols
- 9) Proanthocyanins
- 10) Color intensity
- 11) Hue
- 12) OD280/OD315 of diluted wines
- 13) Proline

According to the above features wine can be classified as

- Class 1
- Class 2
- Class 3

**Consider below characteristics of Machine Learning Application :**

<b>Classifier :</b>	<b>K Nearest Neighbour</b>
<b>DataSet :</b>	<b>Winne Predictor Dataset</b>
<b>Features :</b>	<b>Alcohol, Malic acid , Ash, Alcalinity of ash , Magnesium , Total phenols , Flavanoids , Nonflavanoid phenols , Proanthocyanins , Color intensity, Hue , OD280/OD315 of diluted wines , Proline</b>
<b>Labels :</b>	<b>Class 1, Class 2, Class 3</b>
<b>Training Dataset :</b>	<b>70% of 178 Entries</b>
<b>Testing Dataset :</b>	<b>30% of 178 Entries</b>

## Consider below Machine Learning Application

```
1 from sklearn import metrics
2 from sklearn import datasets
3 from sklearn.neighbors import KNeighborsClassifier
4 from sklearn.model_selection import train_test_split
5
6 def WinePredictor():
7     #Load dataset
8     wine = datasets.load_wine()
9
10    # print the names of the features
11    print(wine.feature_names)
12
13    # print the label species(class_0, class_1, class_2)
14    print(wine.target_names)
15
16    # print the wine data (top 5 records)
17    print(wine.data[0:5])
18
19    # print the wine labels (0:Class_0, 1:Class_1, 2:Class_3)
20    print(wine.target)
21
22    # Split dataset into training set and test set
23    X_train, X_test, y_train, y_test = train_test_split(wine.data, wine.target, test_size=0.3) # 70% training and
        30% test
24
25    #Create KNN Classifier
26    knn = KNeighborsClassifier(n_neighbors=3)
27
28    #Train the model using the training sets
29    knn.fit(X_train, y_train)
30
31    #Predict the response for test dataset
32    y_pred = knn.predict(X_test)
33
34    # Model Accuracy, how often is the classifier correct?
35    print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
36
37 def main():
38     print("---- Marvellous Infosystems by Piyush Khairnar-----")
39
40     print("Machine Learning Application")
41
42     print("Wine predictor application using K Nearest Neighbor algorithm")
43
44     WinePredictor()
45
46 if __name__ == "__main__":
47     main()
48
```

**Output of above Application :**

```

(base) MacBook-Pro-de-MARVELLOUS: Wine_Classifier_KNN -- bash -- 51x20
N marvellous$ python3 WinePredictor.py
---- Marvellous Infosystems by Piyush Khairnar ----
Machine Learning Application
Wine predictor application using K Nearest Neighbor
algorithm
['alcohol', 'malic_acid', 'ash', 'alcalinity_of_ash',
'magnesium', 'total_phenols', 'flavonoids', 'non
flavonoid_phenols', 'proanthocyanins', 'color_inten
sity', 'hue', 'od280/od315_of_diluted_wines', 'prol
ine']
['class_0' 'class_1' 'class_2']
[[1.423e+01 1.710e+00 2.430e+00 1.560e+01 1.270e+02
2.800e+00 3.060e+00
2.800e-01 2.290e+00 5.640e+00 1.040e+00 3.920e+00
1.065e+03]
[1.320e+01 1.780e+00 2.140e+00 1.120e+01 1.000e+02
2.650e+00 2.760e+00
2.600e-01 1.280e+00 4.380e+00 1.050e+00 3.400e+00
1.050e+03]
2.400e-01 2.180e+00 7.800e+00 8.600e-01 3.450e+00
1.480e+03]
[1.324e+01 2.590e+00 2.870e+00 2.100e+01 1.180e+02
2.800e+00 2.690e+00
3.900e-01 1.820e+00 4.320e+00 1.040e+00 2.930e+00
7.350e+02]]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 2 2]
Accuracy: 0.6666666666666666
(base) MacBook-Pro-de-MARVELLOUS: Wine_Classifier_KNN
N marvellous$

```