

Wine Data Set - A Case Study using Logistic Regression and PCA

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

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Using chemical analysis determine the origin of wines



Data Set Characteristics:	Multivariate	Number of Instances:	178	Area:	Physical
Attribute Characteristics:	Integer, Real	Number of Attributes:	13	Date Donated	1991-07-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	849522

The attributes are

- 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

In a classification context, this is a well posed problem with "well behaved" class structures. Since we can't visualise the results with 13 attributes, we need to apply dimensionality reduction techniques.

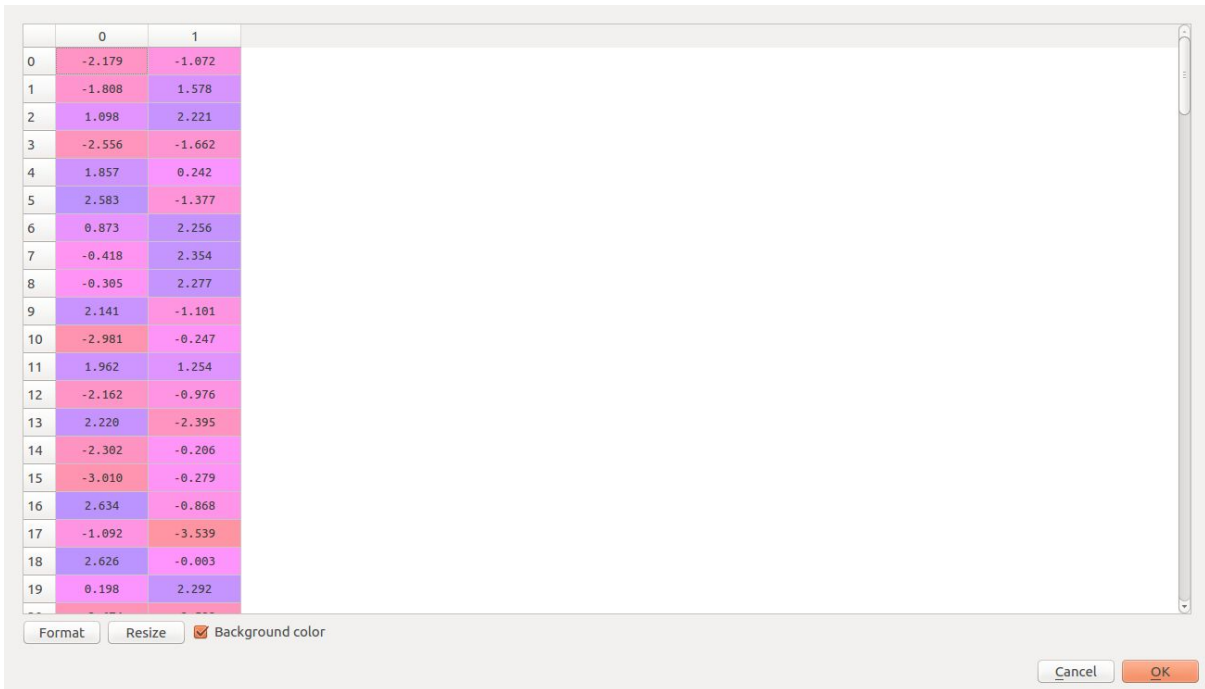
We have used **PCA (Principal Component Analysis)** for dimensionality reduction.

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0.877	0.798	0.644	0.130	0.489	-0.703	-1.428	1.072	-1.368	0.352	0.029	-1.064	-0.206
1	-0.367	-0.758	-0.398	0.334	-1.413	-1.442	-0.503	1.701	0.024	-0.841	0.029	-0.731	-0.817
2	-1.697	-0.344	-0.323	-0.453	-0.145	1.249	0.320	-1.521	-0.435	-0.757	0.902	0.519	-1.313
3	0.516	1.383	0.421	1.004	0.136	-0.752	-1.233	0.522	-0.333	0.951	-1.106	-1.425	0.025
4	0.640	-0.506	0.905	0.130	-0.286	0.445	0.680	-0.656	0.092	-0.643	0.727	1.713	0.339
5	0.926	-0.785	1.240	0.859	0.066	1.216	1.338	-0.578	1.314	0.276	1.033	0.172	1.793
6	-0.864	0.412	-0.547	-0.453	-0.850	0.314	0.309	-0.892	0.652	-1.221	0.858	0.991	-1.494
7	-0.478	-0.929	-1.737	-0.308	-0.850	-1.327	-0.606	-0.578	-0.435	-1.094	0.378	0.255	-0.596
8	-1.958	-1.469	0.495	0.421	-0.850	0.363	0.063	0.444	-0.282	-0.828	0.640	-0.384	-1.019
9	0.815	0.654	0.719	-1.270	1.122	0.724	1.111	-1.521	0.092	0.023	0.029	1.074	0.339
10	-0.478	0.079	-0.621	-0.308	-0.427	-1.048	-1.326	2.094	-1.131	0.866	-0.975	-1.397	-0.156
11	-1.274	-1.154	-0.249	0.421	0.066	1.840	0.196	-1.835	0.075	-0.774	0.160	0.755	0.475
12	-0.914	1.356	-0.621	-0.308	0.041	-1.442	-1.202	-0.578	-0.791	1.334	-1.324	-0.814	0.372
13	1.635	-0.407	1.314	0.130	1.404	0.888	1.225	-0.263	0.618	0.407	0.509	0.089	1.776
14	-0.130	0.555	0.123	0.130	0.277	-1.573	-0.750	-0.971	-1.317	0.150	-0.931	-1.620	-0.701
15	0.628	1.095	-0.658	-0.016	-0.850	-1.048	-1.511	1.701	-1.232	0.276	-0.626	-1.064	-0.536
16	0.715	-0.596	-0.212	-0.978	1.193	1.462	1.379	-0.185	1.246	0.457	-0.015	1.102	0.174
17	1.685	-0.623	1.240	1.587	-0.145	0.888	-0.657	1.308	1.857	3.354	-1.673	-0.870	-0.272
18	0.902	-0.461	-0.026	-0.061	0.066	0.576	0.957	-0.735	0.142	-0.525	0.684	1.963	0.967
19	-0.951	-0.974	-1.589	-0.162	-0.568	0.166	0.093	0.208	0.004	-0.989	-0.407	0.602	-1.422

When we apply PCA, we get two principal components with maximum variance. This is understood by observing the **explained variance matrix**.

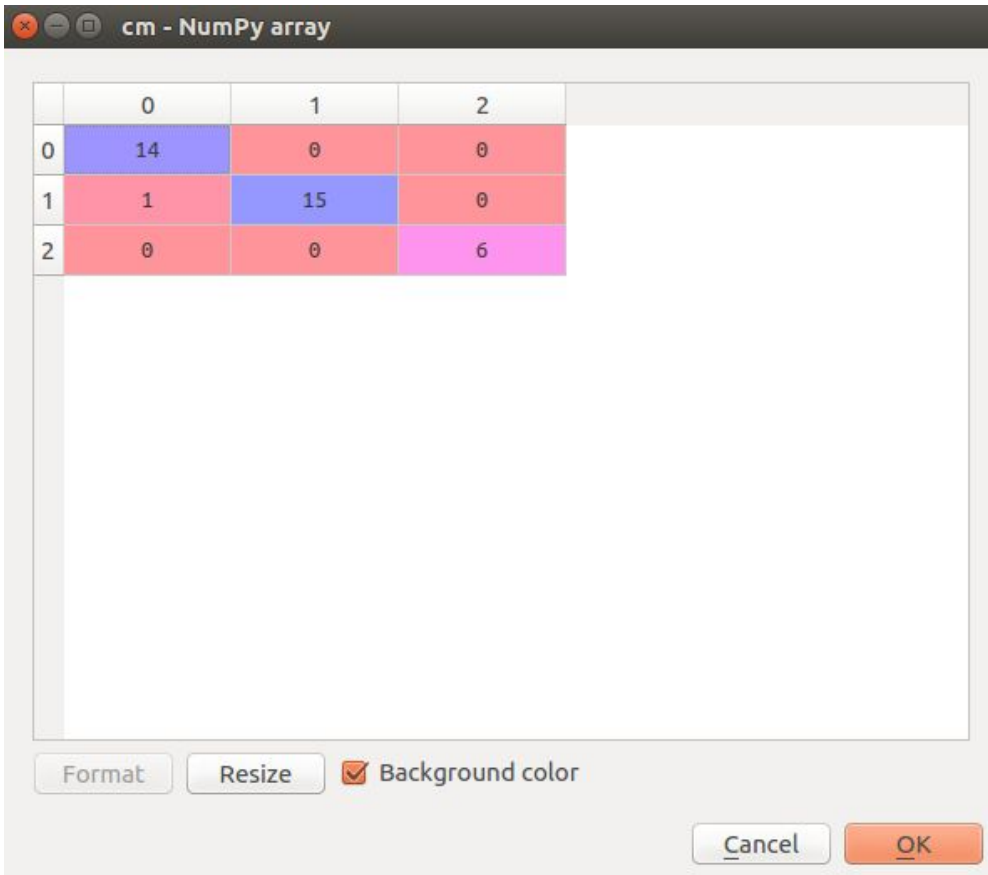
	0
0	0.369
1	0.193

The **PCA algorithm** reduces the 13 attribute dataset into 2 principal components as shown below.



	0	1
0	-2.179	-1.072
1	-1.808	1.578
2	1.098	2.221
3	-2.556	-1.662
4	1.857	0.242
5	2.583	-1.377
6	0.873	2.256
7	-0.418	2.354
8	-0.305	2.277
9	2.141	-1.101
10	-2.981	-0.247
11	1.962	1.254
12	-2.162	-0.976
13	2.220	-2.395
14	-2.302	-0.206
15	-3.010	-0.279
16	2.634	-0.868
17	-1.092	-3.539
18	2.626	-0.003
19	0.198	2.292

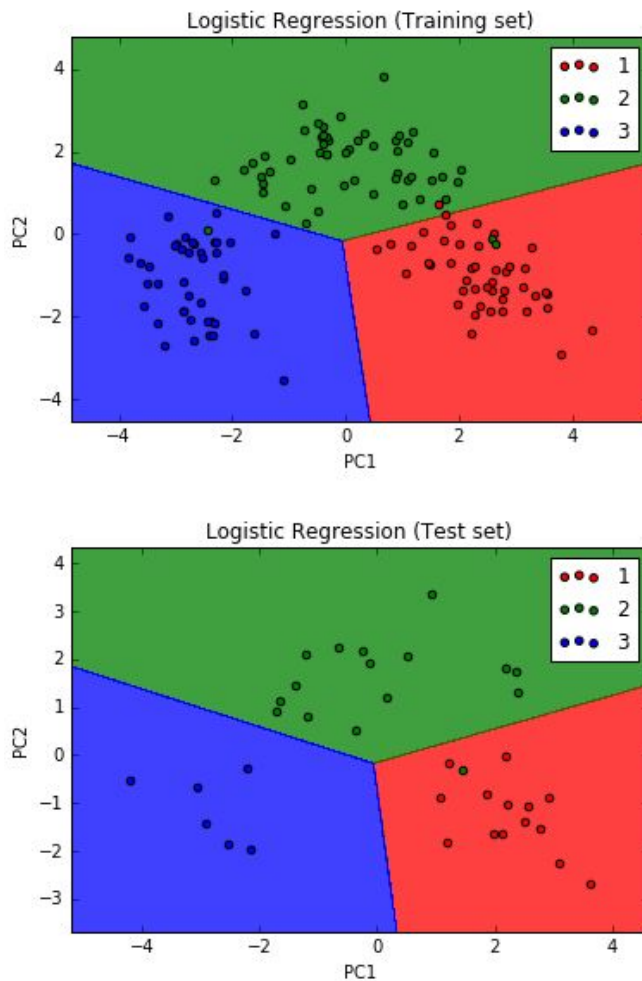
The **confusion matrix** is constructed to find out the accuracy of prediction. The diagonal elements are the correctly predicted outputs. The other cell elements are the incorrectly predicted outputs.



	0	1	2
0	14	0	0
1	1	15	0
2	0	0	6

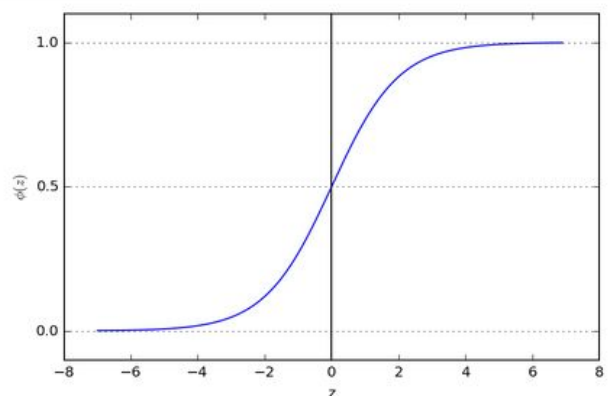
We are using **Matplotlib.pyplot** python library to visualise the results. We are using a mesh grid to plot the results. The training set and test set data is visualised. There are 3 prediction regions.

The green dot in blue region is an **outlier**.



Logistic Regression is a classification algorithm, which uses sigmoid function.

$$P(y = 1 | x) = \phi(z) = \frac{1}{1 + e^{-z}}$$



The model predicts with 97.22 % accuracy.