Linear Discriminant Analysis

Definition

The goal of an LDA is to project a feature space (a dataset n-dimensional samples) onto a smaller subspace k (where $k \leq n-1$ $k \leq n-1$) while maintaining the class-discriminatory information.
In general, dimensionality reduction does not only help reducing computational costs for a given classification task, but it can also be helpful to avoid overfitting by minimizing the error in parameter estimation (“curse of dimensionality”).

What to learn ?

a) **Covariance matrix**
b) **Eigen value and vector**
c) **Discriminant function**
d) **Mean**
e) **Matrix multiplication**
f) **Matrix inverse**

Algorithm

1) Pick a dataset

ring curvature	ring diameter	Result
2.95	6.63	Pass
2.53	7.79	Pass
3.57	5.65	Pass
3.16	5.47	Pass
2.58	4.46	not Pass
2.16	6.62	not Pass
3.27	3.52	not Pass

2) **Create independent variables from features and dependent variables from class.**

In above example independent variable x=

2.95	6.63
2.53	7.79
3.57	5.65
3.16	5.47
2.58	4.46
2.16	6.62
3.27	3.52

Dependent variable y =

1
1
1
1
2
2
2

3) **Calculate number of groups g. here it is 2**
4) **Calculate features data for each group x_i**

$$\mathbf{x}_1 = \begin{bmatrix} 2.95 & 6.63 \\ 2.53 & 7.79 \\ 3.57 & 5.65 \\ 3.16 & 5.47 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 2.58 & 4.46 \\ 2.16 & 6.22 \\ 3.27 & 3.52 \end{bmatrix}$$

5) **Calculate mean of features of each group.**

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 3.05 & 6.38 \end{bmatrix}, \quad \boldsymbol{\mu}_2 = \begin{bmatrix} 2.67 & 4.73 \end{bmatrix}$$

6) **Calculate global mean which is average of all x**

$$\boldsymbol{\mu} = \begin{bmatrix} 2.88 & 5.676 \end{bmatrix}$$

7) **Calculate mean corrected data. For each group x_i, minus the global mean from each matrix cell value.**
   For example , 2.95 – 2.88=.07 approx. Please note that the below values are for references only.

$$\mathbf{x}_1^o = \begin{bmatrix} 0.060 & 0.951 \\ -0.357 & 2.109 \\ 0.679 & -0.025 \\ 0.269 & -0.209 \end{bmatrix}, \quad \mathbf{x}_2^o = \begin{bmatrix} -0.305 & -1.218 \\ -0.732 & 0.547 \\ 0.386 & -2.155 \end{bmatrix}$$

8) **Calculate covariance matrix for each group i**

$$\mathbf{c}_i = \frac{\left(\mathbf{x}_i^o\right)^T \mathbf{x}_i^o}{n_i}$$

$$\mathbf{c}_1 = \begin{bmatrix} 0.166 & -0.192 \\ -0.192 & 1.349 \end{bmatrix}, \quad \mathbf{c}_2 = \begin{bmatrix} 0.259 & -0.286 \\ -0.286 & 2.142 \end{bmatrix}$$

9) **Calculate pooled covariance matrix from the above created matrices.**

$$C(r,s) = \frac{1}{n}\sum_{i=1}^{g} n_i \cdot c_i(r,s)$$

$$\tfrac{4}{7} \cdot 0.166 + \tfrac{3}{7} \cdot 0.259 = 0.206 \quad \tfrac{4}{7}(-0.192) + \tfrac{3}{7}(-0.286) = -0.233$$

$$\mathbf{C} = \begin{bmatrix} 0.206 & -0.233 \\ -0.233 & 1.689 \end{bmatrix}$$

10) **Find inverse of above matrix**

$$\mathbf{C}^{-1} = \begin{bmatrix} 5.745 & 0.791 \\ 0.791 & 0.701 \end{bmatrix}$$

11) **Find prior probability vector (in case nothing is given than we assume it using below formula)**

$$p_i = \frac{n_i}{N} \qquad \mathbf{p} = \begin{bmatrix} 0.571 \\ 0.429 \end{bmatrix} = \begin{bmatrix} 4/7 \\ 3/7 \end{bmatrix}$$

12) **Calculate discriminant function (this is the main function)**

$$f_i = \boldsymbol{\mu}_i \mathbf{C}^{-1} \mathbf{x}_k^T - \tfrac{1}{2} \boldsymbol{\mu}_i \mathbf{C}^{-1} \boldsymbol{\mu}_i^T + \ln(p_i)$$

This is matrix multiplication.

13) **Classify the existing data.**

14) **Similarly predict it for new values. Classify to class for which discriminant function is the maximum.**

Like for values [2.81,5.46] class predicted is 2

Excel calculations:

	A	B	C	D	E	F	G	H	I	J	K
1	**Linear Discriminant Analysis: 2 Category 2 Dimensional Data**										
2											
3	**Training Data, D**				**Mean Corrected Data**			Discriminant function			**Results**
4	class	X1	X2		X1o	X2o		f1	f2		**Classificati**
5	1	2.95	6.63		0.060	0.951		55.220	53.071		1
6	1	2.53	7.79		-0.357	2.109		53.774	51.394		1
7	1	3.57	5.65		0.679	-0.025		62.476	59.589		1
8	1	3.16	5.47		0.269	-0.209		51.953	50.764		1
9	2	2.58	4.46		-0.305	-1.218		32.028	34.313		2
10	2	2.16	6.22		-0.732	0.547		34.554	35.757		2
11	2	3.27	3.52		0.386	-2.155		41.174	42.414		2
12	**prediction**	2.81	5.46		-0.078	-0.219		44.049	44.085		2
13											
14	average	2.888	5.676								
15	std dev	0.454	1.300								

Sample program for above scenario using sklearn

Input dataset inputdata.csv

```
, ,
1,2.95,6.63
1,2.53,7.79
1,3.57,5.65
1,3.16,5.47
2,2.58,4.46
2,2.16,6.62
2,3.27,3.52
```

Program

```python
import numpy as np
import pandas as pd
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis

clf = LinearDiscriminantAnalysis()
```

```
classes=np.array([1,1,1,1,2,2,2])

#importing the data
df = pd.DataFrame.from_csv("inputdata1.csv")
data_matrix = df.as_matrix()

print(data_matrix)
clf.fit(data_matrix,classes)

print("Prediction:")
print(clf.predict([[2.81, 5.46]]))
```

References:

https://machinelearningmastery.com/linear-discriminant-analysis-for-machine-learning/

http://people.revoledu.com/kardi/tutorial/LDA/Numerical%20Example.html

https://github.com/kshitijgorde/classical-machine-learning-from-scratch/blob/9eb15a76a257a0f7d9dfd08b0ac41cb121b23736/LDA/Machine%20Learning.ipynb

https://byjus.com/covariance-matrix-formula