Linear Discriminent Analysis (LDA)

Linear model for classification and dimentionality reduction

Useful for feature extraction in pattern recognition face recognition

LDA is a stastical technique for catagorizing data into groups.

By maximizing the seperation between classes, it enables accurate classification of new data points.

Logistic regression falls short in multiclass classification where LDA shines.

What LDA does

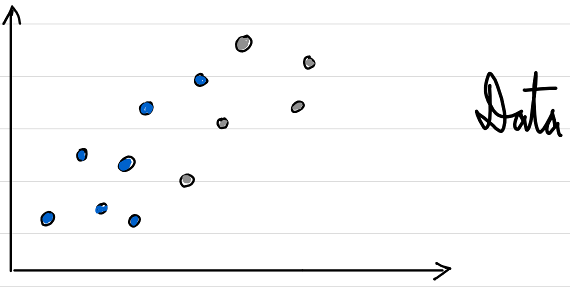
LDA is just like PCA with a different objective

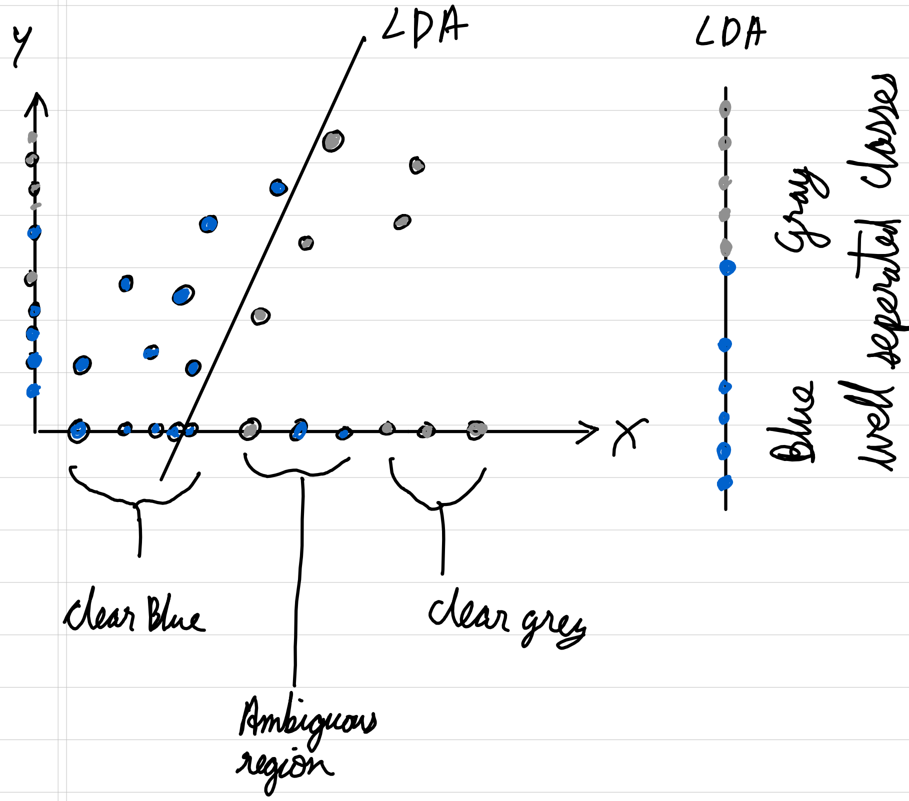
In PCA, focus is on the line that captures the most variation.

In LDA, we have labelled data. We know which class of the data it belongs to. LDA is supervised while PCA is unsupervised

The aim of LDA is to maximize the seperability between 2 classes (known)

We know beforehand which point is in which class unlike t-SNE



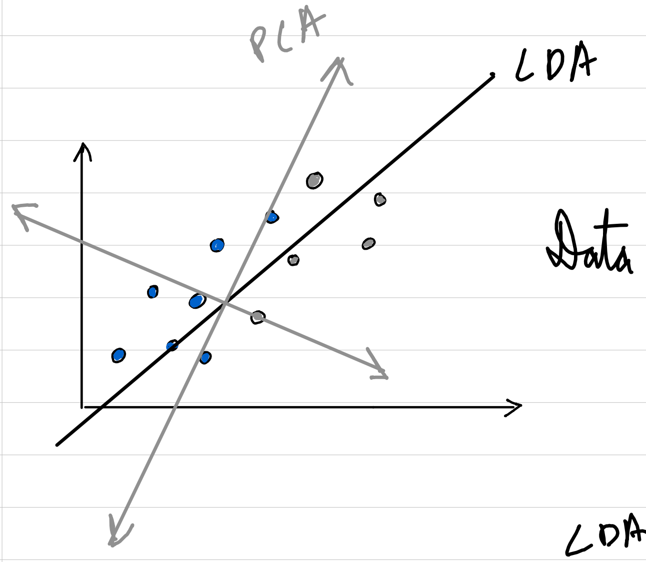


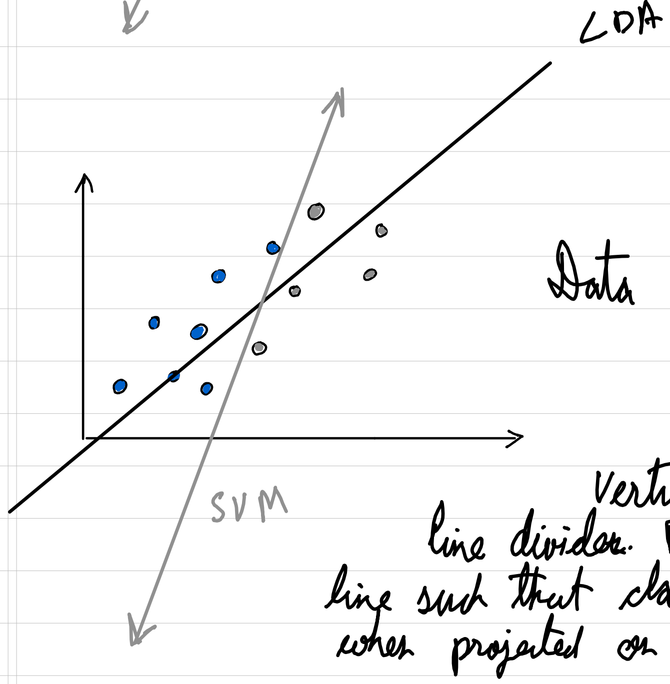
LDA uses information of the classes to create a new axis that maximizes the seperation between the two classes.

Like PCA & Unlike t-SNE, LDA just chooses a line to project on. It does not do any other change in the points.

t-SNE on the otherhand shuffles the points & then tries to match with the original ordering.

Goal is that even after dimentionality reduction the classes must remain classes, but when classes are known.





SVM splits data vertically accross as a line divider. But LDA draws line such that classes get divided when projected on to a lower dimention

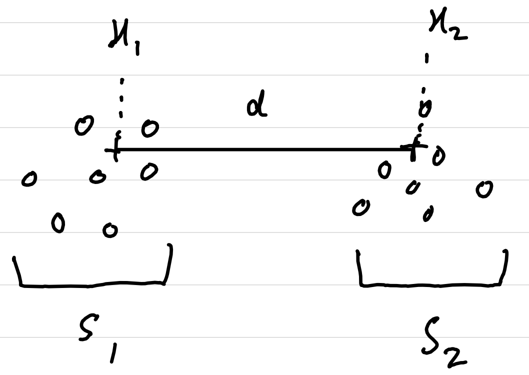
Well seperated classes

what does well seperated classes mean?

It means they must be far apart ① and elements in class must be close together ②

① Maximize the distance between 2 class means

② Minimize the variation (scatter) inbetween a class



m_1 & m_2 are projected means

$m_1 - m_2 \rightarrow$ large

$S_1 ; S_2 \longrightarrow$ small

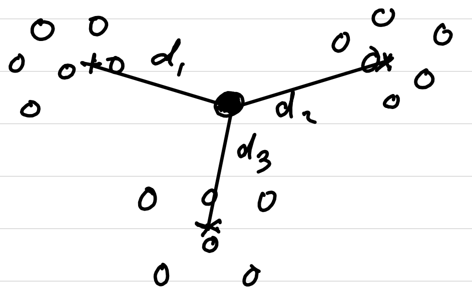
Hence the objective becomes

$$\text{Maximize} \quad \frac{(m_1 - m_2)^2}{s_1^2 + s_2^2}$$

This is known as Fisher Discriminant ratio

Note → If we dont consider condition ② and maximize the distance between the means without considering the scatter, then the seperation between the classes is not that great. The classes will be fuzzy. overlapping

for Multiple classes, first find the centroid of the whole dataset. Then maximize the distance of mean of every class from the datapoint



$$\text{Maximize} \quad \frac{d_1^2 + d_2^2 + d_3^2}{S_1^2 + S_2^2 + S_3^2}$$

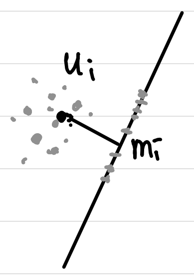
$S \rightarrow$ Variance of projection

$\mu \rightarrow$ initial centroid of all data
$m \rightarrow$ projected centroid of all data

$m_i \rightarrow$ Projected Mean of class i (Projected Centroid)
$\mu_i \rightarrow$ Initial mean of class i (Initial Centroid)

$y = Wx \rightarrow$ Projection line

$\therefore \; m_i = W^T \mu_i$



Numerator of objective $\rightarrow$ Maximize

$$d_i^2 = (m_i - m)^2 = (W^T \mu_i - W^T \mu)^2$$

(Inter class scatter projected)

$$= W^T (\mu_i - \mu)(\mu_i - \mu)^T W$$

Inter class scatter Initial $\rightarrow S_{b_i}$

$$S_{b_i} = (\mu_i - \mu)(\mu_i - \mu)^T$$

$$\therefore d_i^2 = W^T S_{b_i} W$$

$$d_1^2 + d_2^2 + \cdots = W^T \underset{\substack{\uparrow \\ \text{Matrix of all classes}}}{S_B} W$$

Denominator $\rightarrow$

Within class scatter matrix Initial $\rightarrow$

for class i
$$S_i' = \sum_{x_i \in \text{class } i} (x_i - \mu_i)(x_i - \mu_i)^T$$

Variance for each class (projected) $\rightarrow$

for class i
$$s_i^2 = \sum_{x_i \in \text{class } i} (W^T x_i - m_i)$$

s_i^2 can be rewritten as $\rightarrow$

$$s_i^2 = \sum_{x_i \in w_i} (W^T x_i - W^T \mu_i)(W^T x_i - W^T \mu_i)^T$$

$$s_i^2 = W^T S_i' W$$

Combining all classes, total variance =

$$s_1^2 + s_2^2 + s_3^2 + \cdots = W^T (S_1' + S_2' + \cdots) W$$

$$= W^T \underset{\substack{\uparrow \\ \text{Matrix}}}{S_W} W$$

where

$$S_W = \sum_i S_i' = \sum_{\substack{\text{all data points} \\ x}} (\underset{\substack{\uparrow \\ \text{data Point}}}{x_i} - \underset{\substack{\uparrow \\ \text{mean of class it belongs to}}}{\mu_i})(x_i - \mu_i)^T$$

Our objective becomes

$$J(W) = \frac{W^T S_b W}{W^T S_W W}$$

(form of generalized Rayleigh quotient)

Differentiating & setting to 0 gives

$$S_b W = \underset{\substack{\uparrow \\ \text{eigen values to be calculated}}}{\lambda} S_W W$$

[S_b, S_W are known as they can be calculated from dataset]

From this, W is found out

Note

Suppose there are two classes A & B of people with height x & weight y

In general $\mathrm{Cov}_A(x, y) \neq \mathrm{Cov}_B(x, y)$

But LDA assumes that

$$\mathrm{Cov}_A(x, y) = \mathrm{Cov}_B(x, y) = \mathrm{Cov}_{\text{Population}}(x, y)$$

Finding the within class scatter takes $N \times (d + d^2)$ time

Eigen decomposition & matrix multiplication takes $O(d^3)$

If $N > d$ then $O(d^3)$
else $O(N d^2)$ (if no of features is trivial as compared to samples)

Assumptions in LDA –

LDA assumes that data is normally distributed within each class.

LDA assumes that every class has equal covariance matrix

LDA is linear

Disadvantages –

1. LDA is sensative to outliers
2. LDA requires large number of samples relative to number of features

Classification →

The line obtained from LDA can not only be used for data reduction, but also for classification (eg using gaussian assumption or simple nearest neighbour)

After LDA projects the data, classification can be done by various methods

eg → Gaussian distribution assumption

Assuming that every distribution has a gaussian and then like GMM finding the class with maximum probability

$$\delta_k(x) = x^T \Sigma^{-1} m_k - \frac{1}{2} m_k^T \Sigma^{-1} m_k + \log P_{ik}$$

(Linear score function depends linearly on x)