

LINEAR DISCRIMINANT ANALYSIS

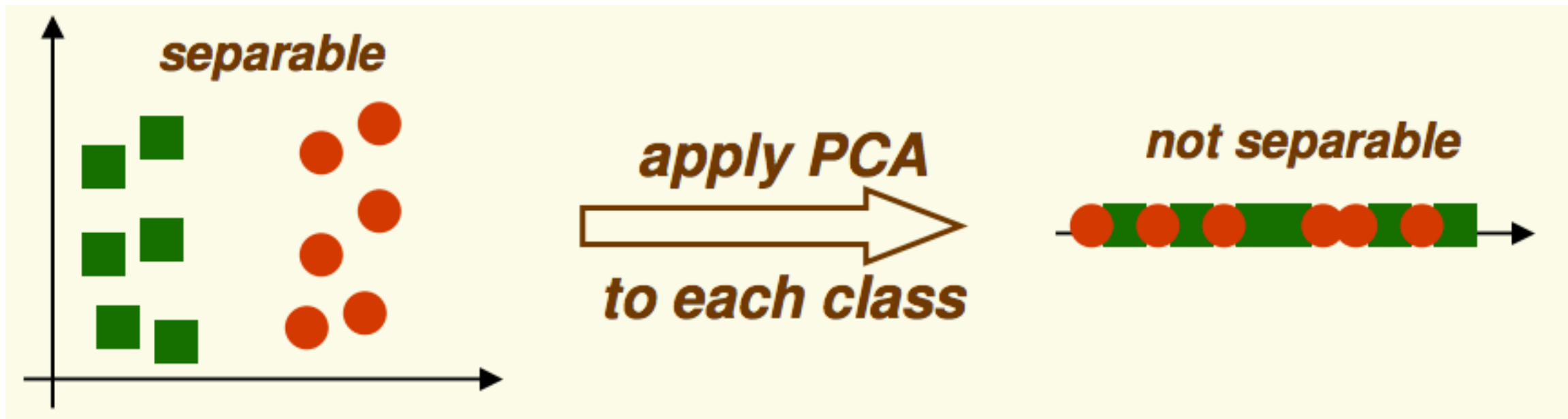
Joseph Nelson, Data Science Immersive

AGENDA

- What are PCA and LDA?
- Formal Procedure Behind LDA
- Bayes Rule
- LDA Code Implementation

BIG IDEA: LDA

- ▶ PCA finds the most accurate representation of our data in a lower dimension space (Remember: we project in the direction that maximizes variance)
- ▶ BUT: What happens when we project our data in a direction that maximizes variance, but does not separate between features? That is, maximizing variance AMONG features does not necessarily maximize variance BETWEEN features.



BIG IDEA: LDA

- ▶ Thus, we'll be introducing another form of preliminary data observation: Fisher's Linear Discriminant Analysis
- ▶ Fisher Linear Discriminant project to a line which preserves direction useful for **data classification**

BIG IDEA: LDA

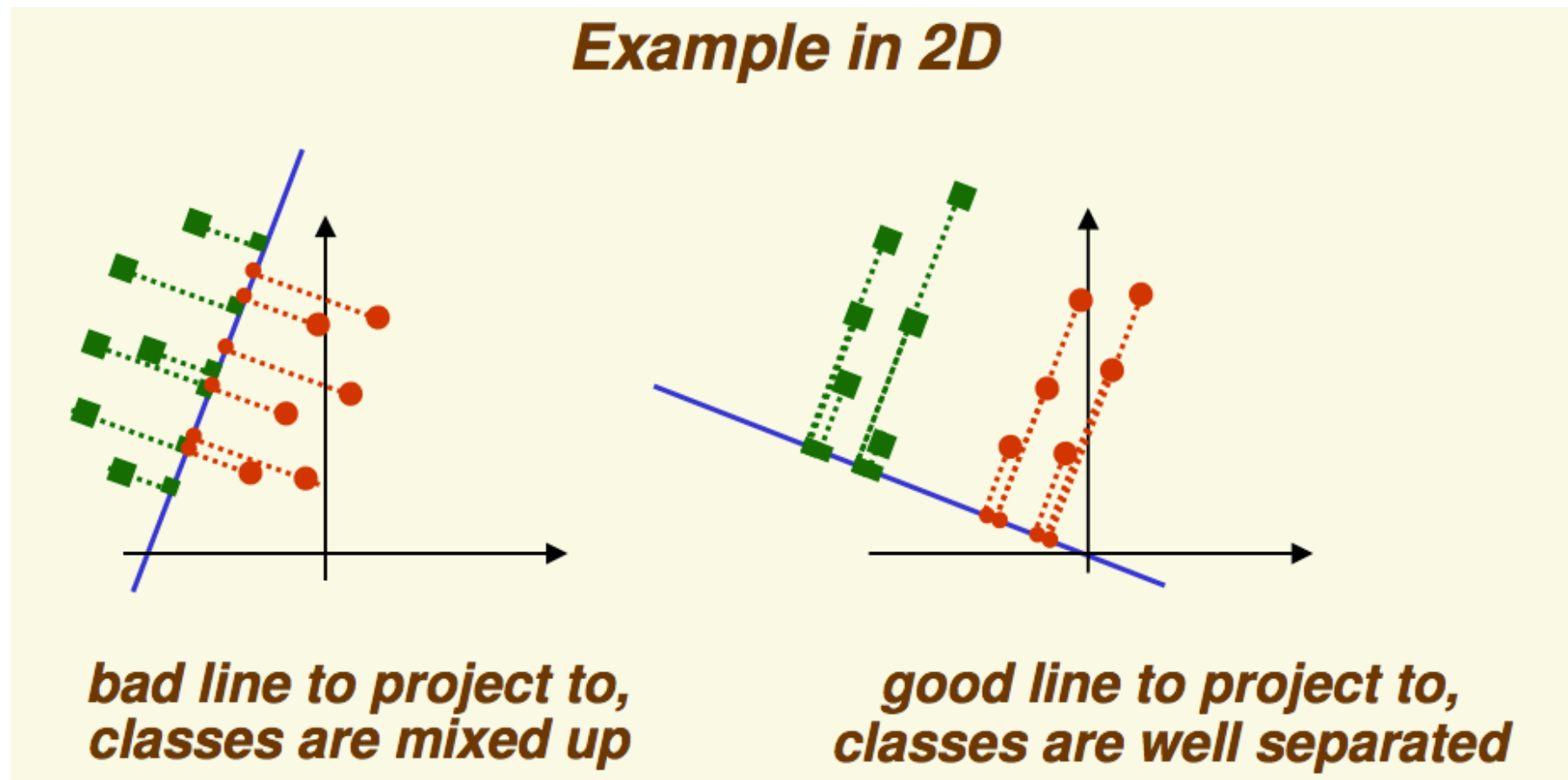
- Thus, we'll be introducing another form of preliminary data observation: Fisher's Linear Discriminant Analysis
- Fisher Linear Discriminant project to a line which preserves direction useful for **data classification**
- Pop quiz: What type of learning is PCA?

BIG IDEA: LDA

- Thus, we'll be introducing another form of preliminary data observation: Fisher's Linear Discriminant Analysis
- Fisher Linear Discriminant project to a line which preserves direction useful for **data classification**
- Pop quiz: What type of learning is PCA?
- PCA is an **unsupervised machine learning** technique that represents our data as matrices and seeks to transform our data in a direction that maximizes variance among our features.

BIG IDEA: LDA

- So, what about LDA?
- Find a projection to a line such that samples from different classes are well separated

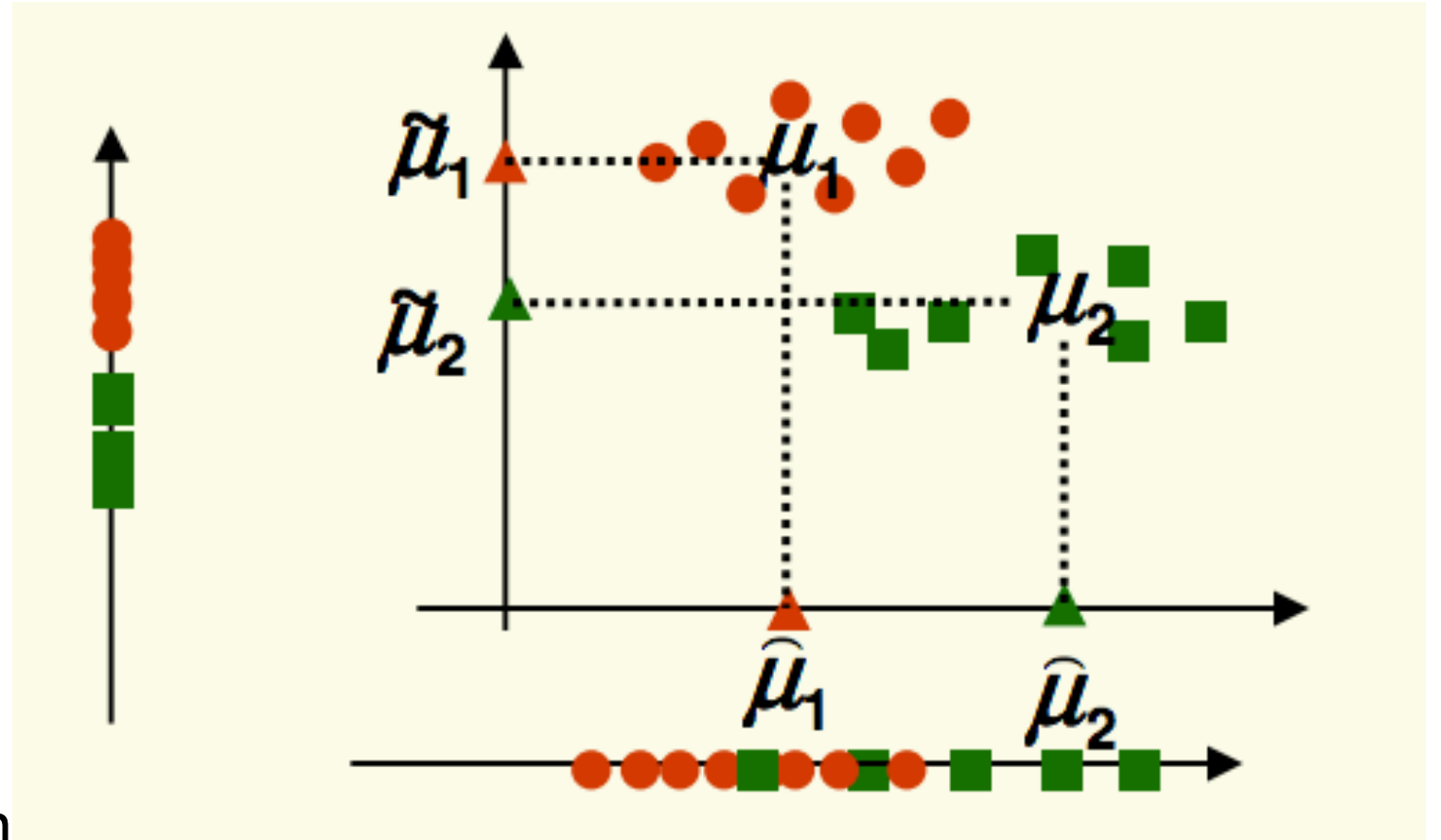


BIG IDEA: LDA

- But wait! We need to discuss any limitations and assumptions
- LDA requires that our data follows a multivariate normal distribution
- LDA explicitly requires a target/feature relationship
- Feature sets are mutually independent, and there is one covariance matrix for the multi-class target

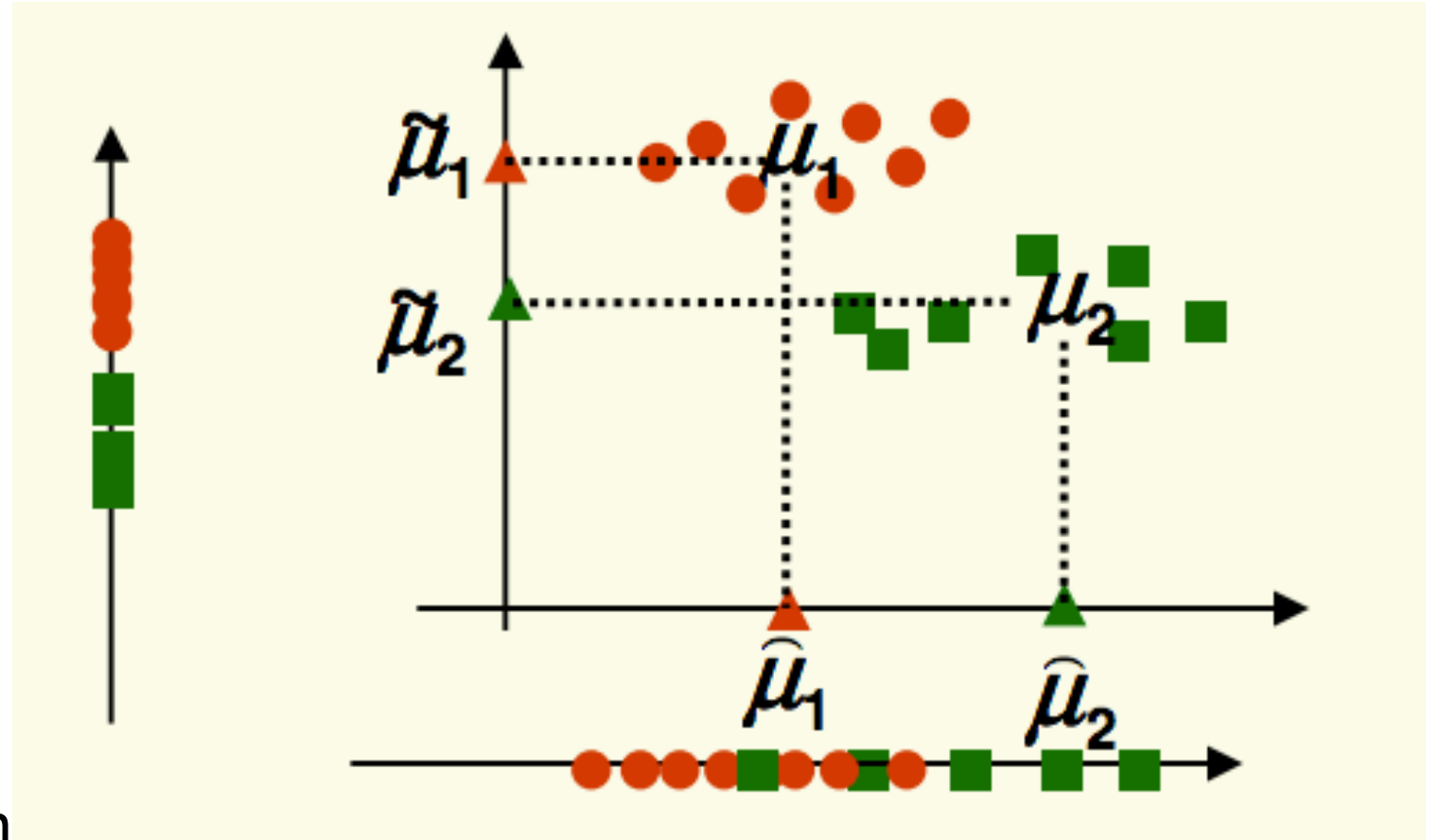
LDA: BEHIND THE SCENES

- ▶ Imagine the scatter plot.
- ▶ If we're trying to project the data onto a one-dimensional line to make them as distinguishable as possible, what do we want to consider?
- ▶ Exercise: Produce an ideal one-dimensional representation of our data



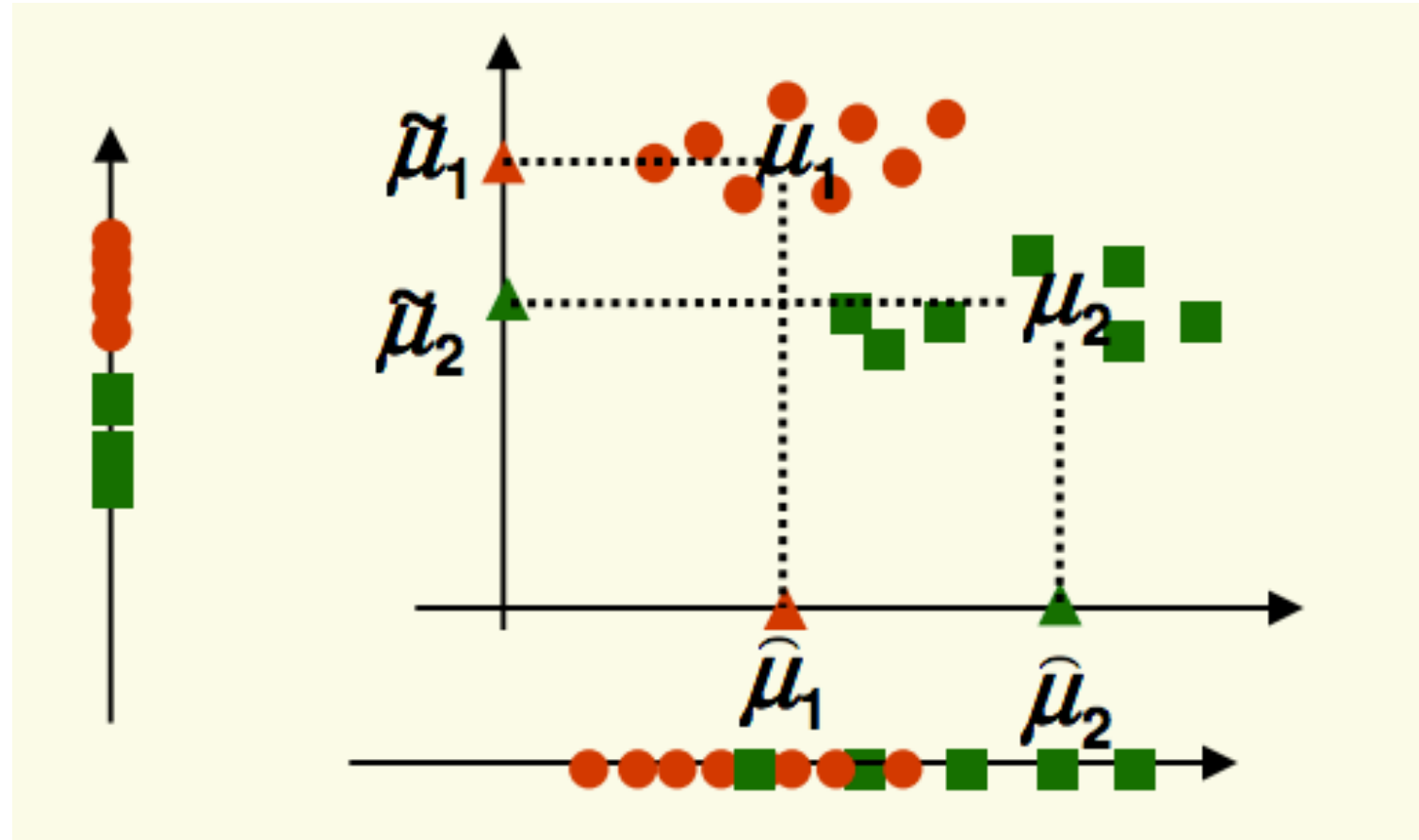
LDA: BEHIND THE SCENES

- ▶ Imagine the scatter plot.
- ▶ If we're trying to project the data onto a one-dimensional line to make them as distinguishable as possible, what do we want to consider?
- ▶ Exercise: Produce an ideal one-dimensional representation of our data
- ▶ What did you notice?



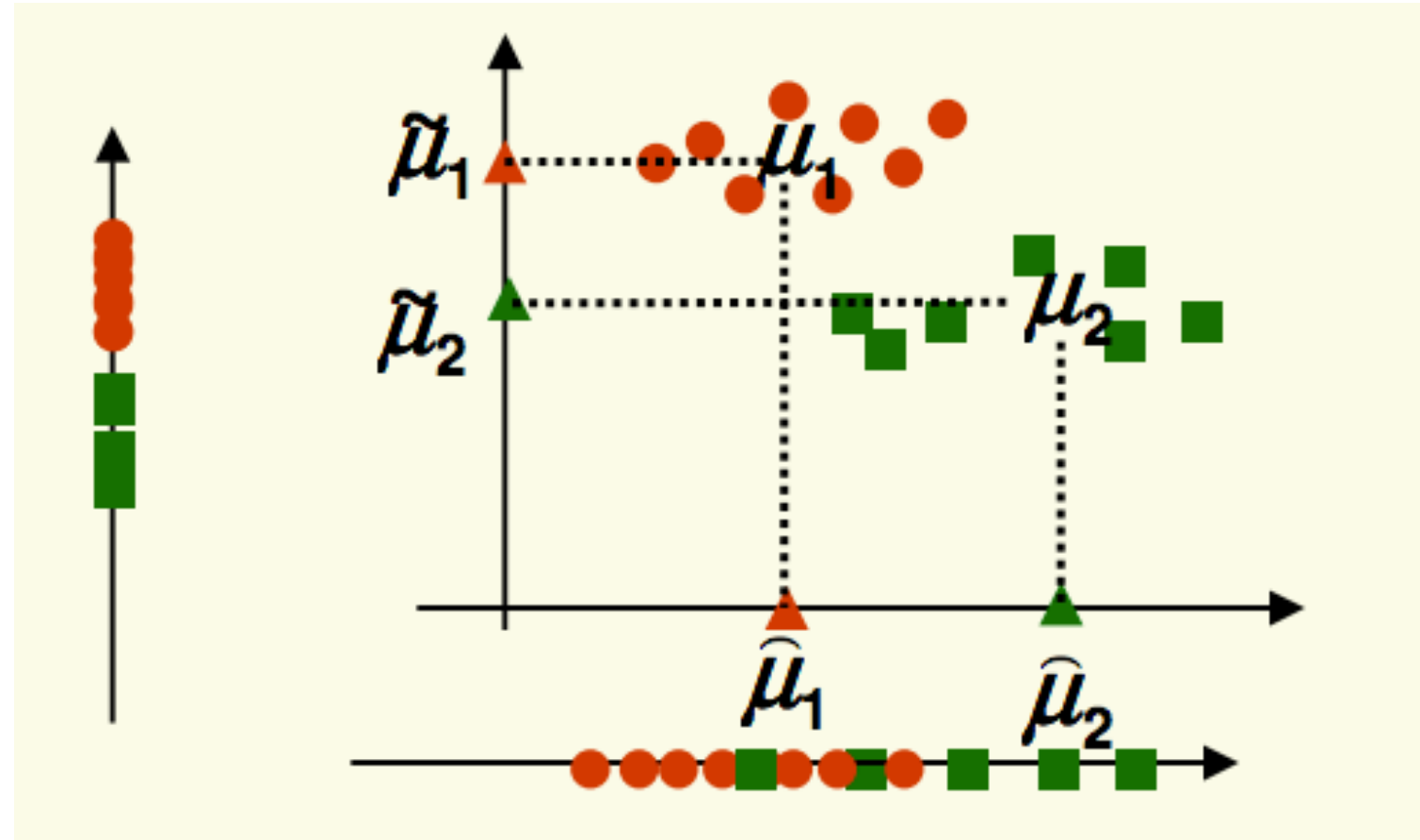
LDA: BEHIND THE SCENES

- ▶ The two values we're concerned with in LDA: variance WITHIN a single observation, and mean BETWEEN class observations.



LDA: BEHIND THE SCENES

- ▶ The two values we're concerned with in LDA: variance **WITHIN** a single observation, and variance **BETWEEN** class observations.
- ▶ We want the **SPREAD WITHIN** classes to be small
- ▶ We want the spread **BETWEEN** classes to be large



LDA: BEHIND THE SCENES

- Within class spread: Scatter is the measure of the variance between ONE GIVEN OBSERVATION (z) and the CLASS MEAN (μ)

$$s = \sum_{i=1}^n (z_i - \mu_z)^2$$

- In other words:

larger scatter:

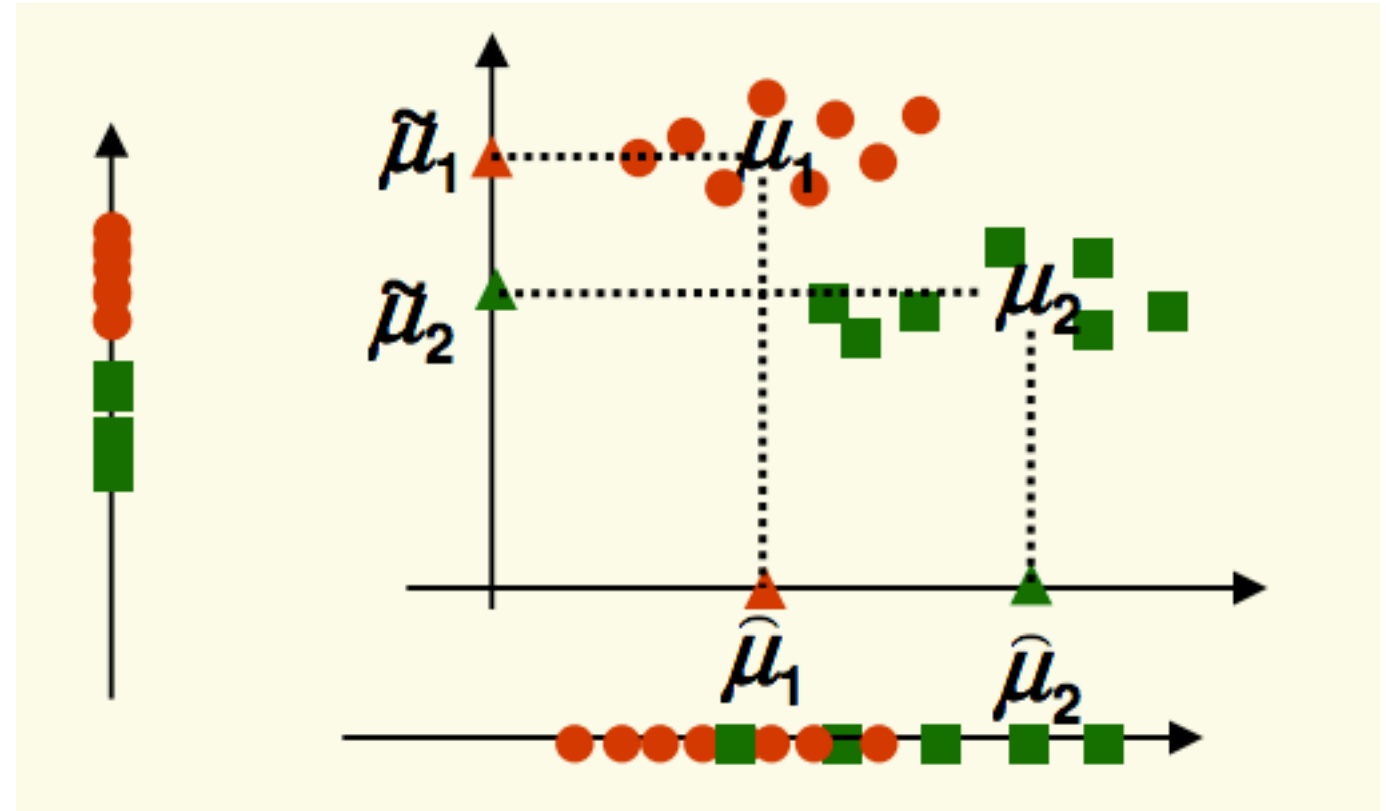


smaller scatter:



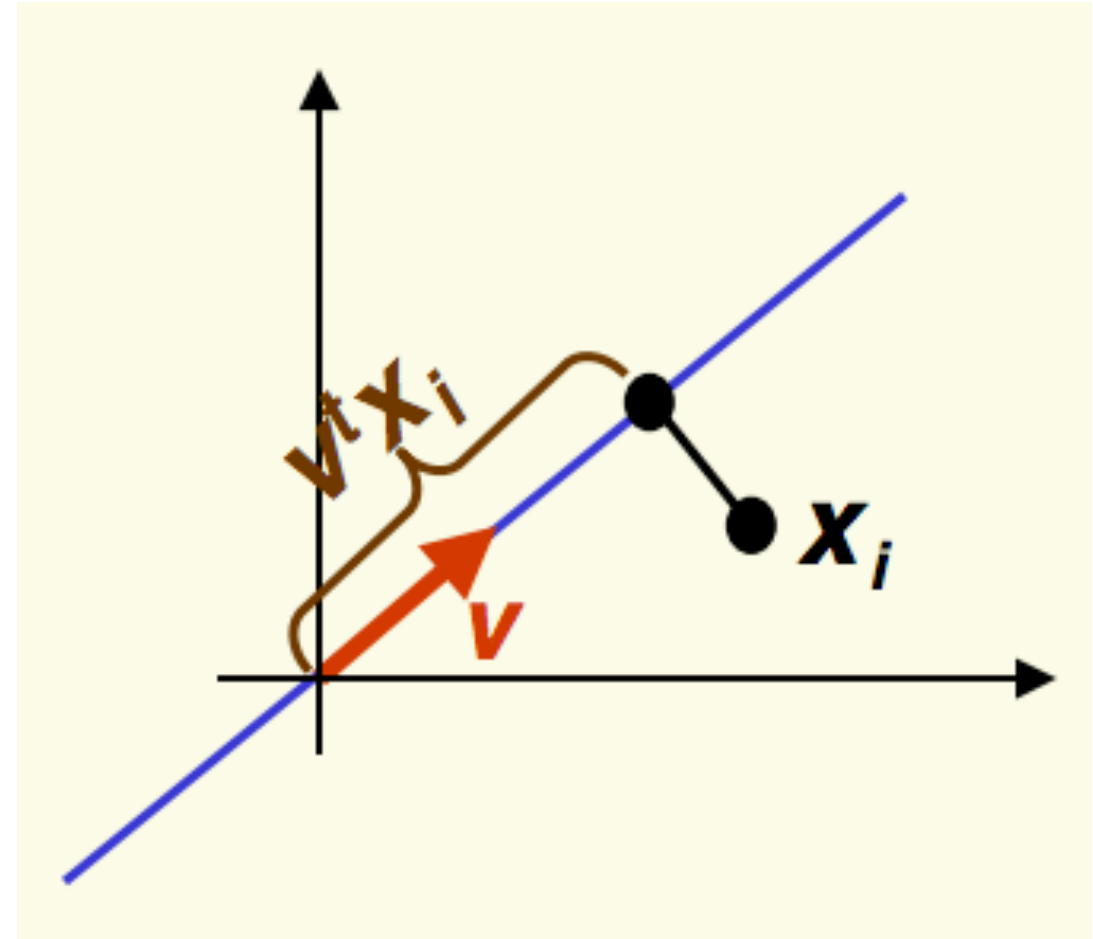
LDA: BEHIND THE SCENES

- Within class spread: Scatter
BETWEEN class means should be large
- Thus, maximizing the projected mean of the populations between u_1 and u_2



LDA: BEHIND THE SCENES

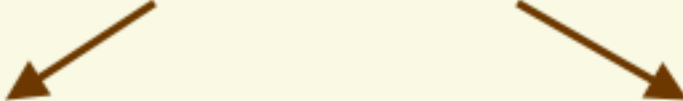
- ▶ So, as we try to find some line v , we want that line to maximize the distance between projected class means YET minimize distance within variance of a given class



LDA: BEHIND THE SCENES

► Which gives us...

want projected means are far from each other

$$J(\mathbf{v}) = \frac{\overbrace{(\tilde{\mu}_1 - \tilde{\mu}_2)^2}}{\tilde{\mathbf{s}}_1^2 + \tilde{\mathbf{s}}_2^2}$$


want scatter in class 1 is as small as possible, i.e. samples of class 1 cluster around the projected mean $\tilde{\mu}_1$

want scatter in class 2 is as small as possible, i.e. samples of class 2 cluster around the projected mean $\tilde{\mu}_2$

WHERE DOES BAYES RULE FIT IN?

- Because we're making class projections based on prior knowledge about the rest of our data, we want to maximize the probability that we choose the right class for some given observation X .

WHERE DOES BAYES RULE FIT IN?

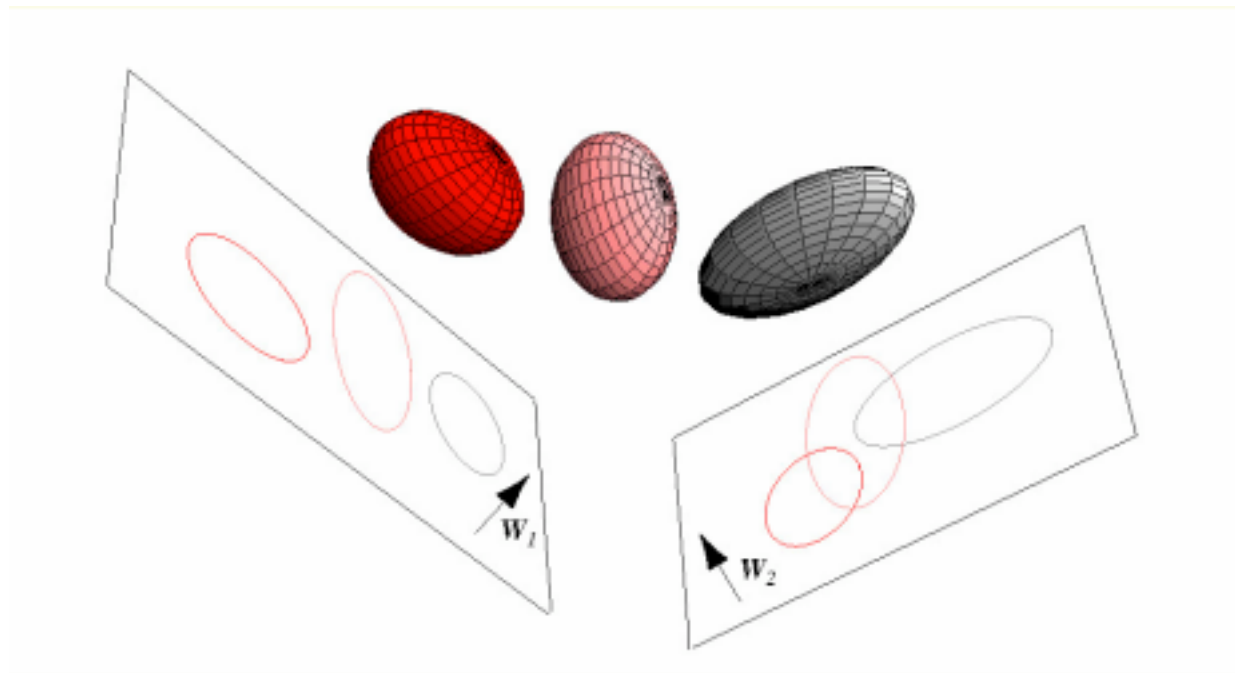
- ▶ Because we're making class projections based on prior knowledge about the rest of our data, we want to maximize the probability that we choose the right class for some given observation X .

$$P(y = k|X) = \frac{P(X|y = k)P(y = k)}{P(X)} = \frac{P(X|y = k)P(y = k)}{\sum_l P(X|y = l) \cdot P(y = l)}$$

- ▶ Thus, we are identifying what class, k , maximizes our probabilistic determination
- ▶ (Psstt... This is just like Naïve Bayes)

LDA: BEHIND THE SCENES

- ▶ Math footnote: We've been looking at all of this in the context of two dimensional data. How does LDA grasp multidimensional data?
- ▶ In case of c classes, can reduce dimensionality to 1, 2, 3, ..., $c-1$ dimensions
- ▶ Project sample x_i to a linear subspace $y_i = V^t x_i$



$$J(V) = \frac{\det(V^t S_B V)}{\det(V^t S_W V)}$$

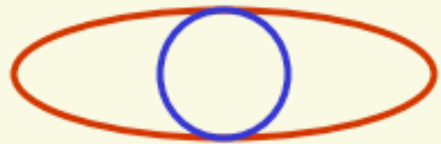
LDA VS PCA


- LDA Key Advantage:
- LDA is built for classification problems. When we are identifying insights from our data for the purpose of classification, LDA is an excellent choice. PCA maximizes intra class variance, but does not consider inter class variance
- Limited tuning parameters
- LDA Key Disadvantage:
- For complex data, projection to even the best line may result in inseparable projected samples

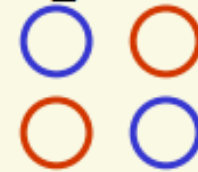
LDA VS PCA

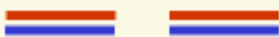
- ▶ LDA doesn't not work if the expected mean of some sample equals the expected mean of some other sample. Examples:

1. $J(\mathbf{v})$ is always 0: happens if $\mu_1 = \mu_2$



PCA performs
reasonably well
here: 



PCA also
fails: 

2. If $J(\mathbf{v})$ is always large: classes have large overlap when projected to any line (PCA will also fail)



CODE

► To the repo...