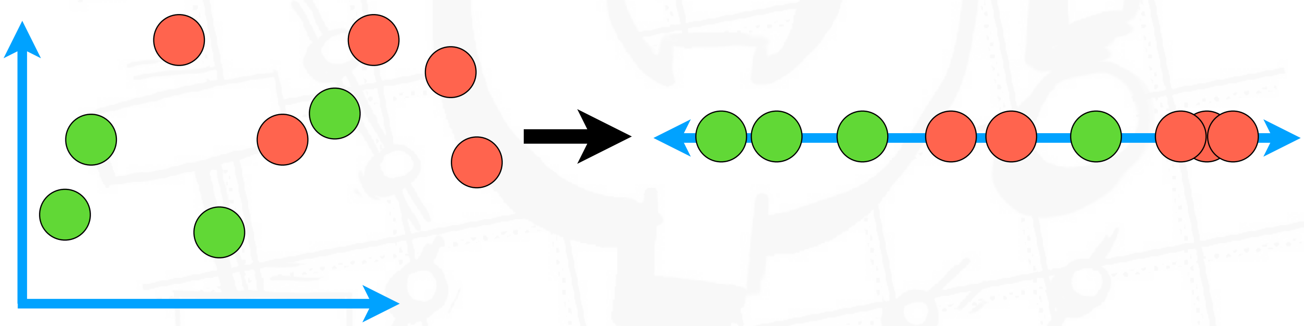
# StatQuest!!!

## Fisher's Linear Discriminant, aka

# Linear Discriminant Analysis (LDA)



# Study Guide!!!

## The Problem

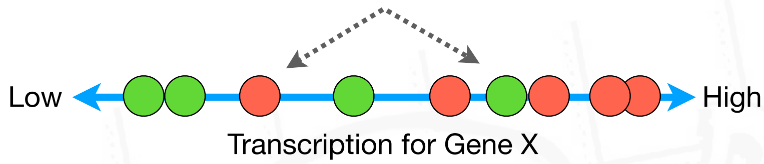
We have a cancer drug that works well in some people, but not others. How do we decide who gets the drug?

Maybe the reason the drug works in some people is due to Genetics. Perhaps gene expression can help us decide.

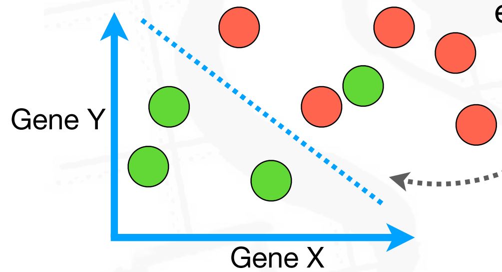
If we use one gene to decide, the data are easy to plot on a number line (a **1-D** graph), but there is overlap and no obvious cutoff for who to give the drug.

= The drug works (hooray!)

= The drug does not work (bummer!)




If we use two genes, the data are easy to plot on a **2-D** graph, but separation isn't perfect.

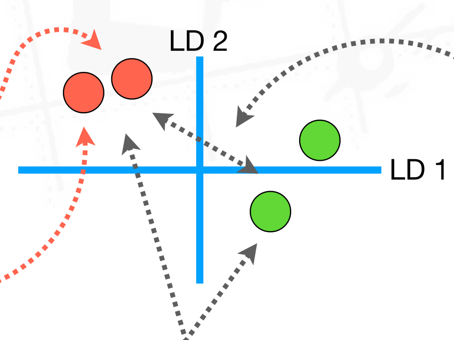



Since there are **20,000** genes in the human genome, it would be nice if we could use all of that data. Unfortunately, we can't draw a **20,000** Dimensional graph.

## The Solution - Linear Discriminant Analysis (LDA)

**Linear Discriminant Analysis (LDA)** is like **Principal Component Analysis (PCA)**, in that it provides a way plot data with a lot of dimensions onto a simple **2-D** graph. However, **LDA** focuses on maximizing the separability among the known categories.

| | Patient 1 | Patient 2 | Patient 3 | Patient 4 |
|---|---|---|---|---|
| Gene 1 | 2 | 3 | 8 | 9 |
| Gene 2 | 2 | 1 | 7 | 6 |
| ... | ... | ... | ... | ... |
| Gene 20000 | 32 | 10 | 0 | 12 |

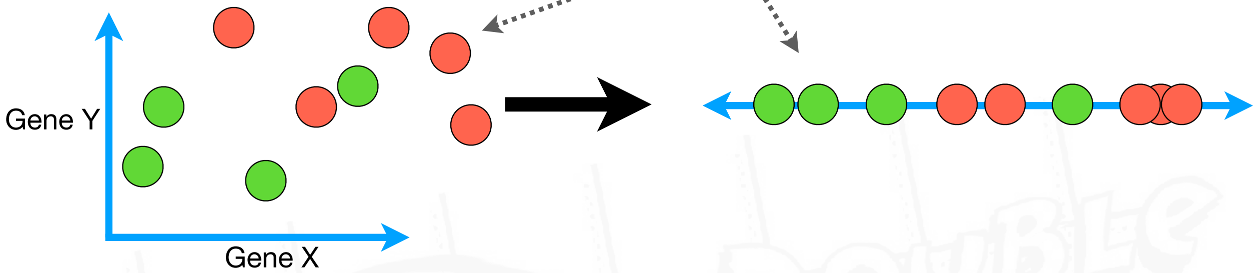



In this example, **LDA** reduced **20,000** genes to a **2-D** graph and maximized the separability between the people for whom drug works and the people for whom the drug does not work.

This makes it possible to determine who to give the drug.

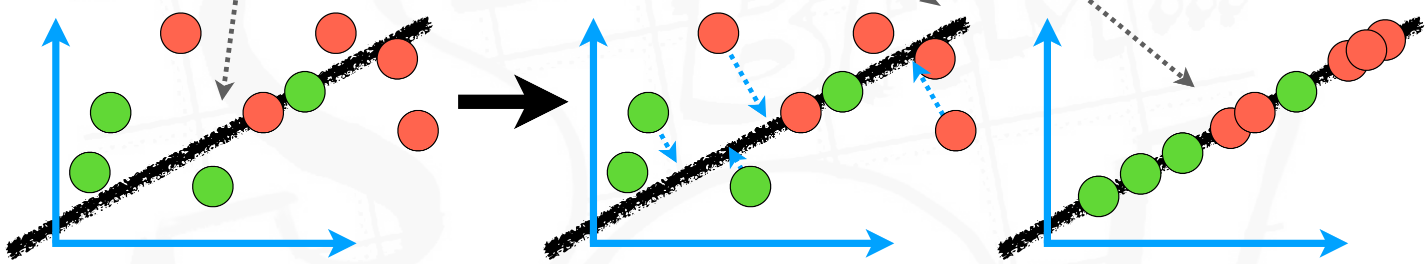
## A super simple example

Let's use **LDA** to reduce a **2-D** graph to a **1-D** graph while maximizing separation between the two categories.



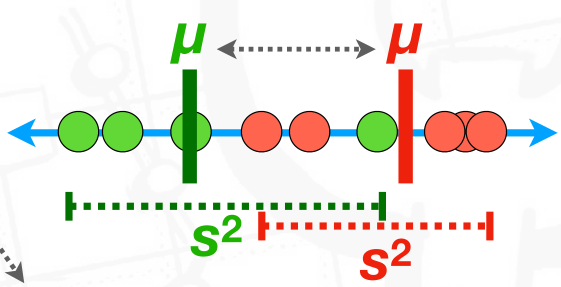

**LDA creates a new axis...**

**...and the data are projected onto it.**



1) Tries to maximize the distance ***between*** the category means, $\mu$ and $\mu$.

**LDA finds a new axis that simultaneously...**




2) Tries to minimize the **scatter** (or variation) ***within*** each category, $s^2$ and $s^2$.

$$\frac{(\mu - \mu)^2}{s^2 + s^2}$$

The larger the distance, the larger the numerator

The smaller the scatter, the smaller the denominator

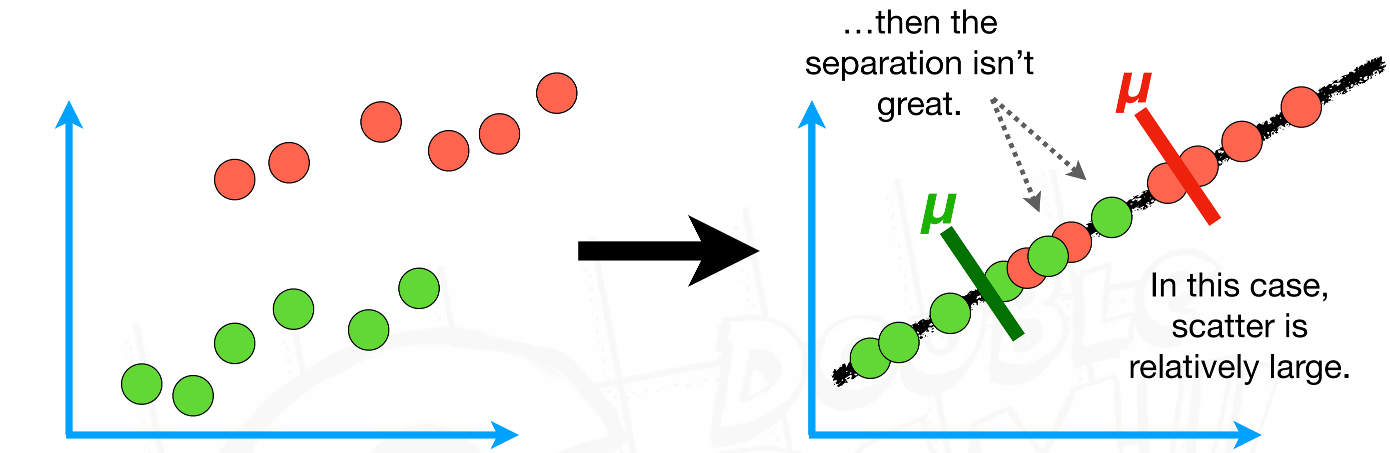
**...maximizes this ratio.**

**Scatter vs Variance:** They both work, but scatter is more often used in this context.

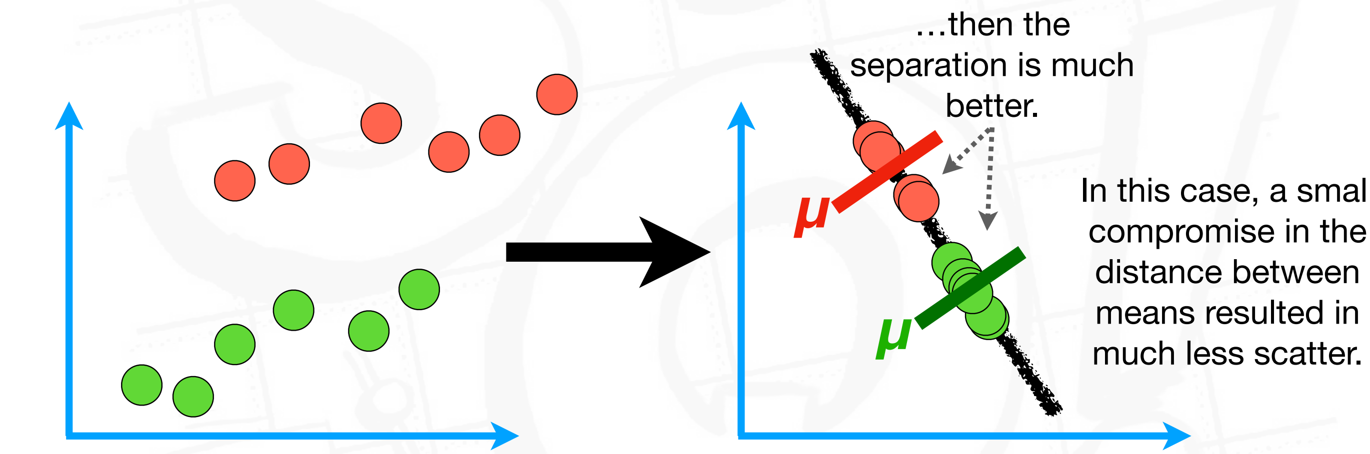
Scatter = $\Sigma(\text{value - mean})^2$

$$\text{Variance} = \frac{\Sigma(\text{value - mean})^2}{n - 1} = \frac{\text{Scatter}}{n - 1}$$

## If we only maximize the distance between means…

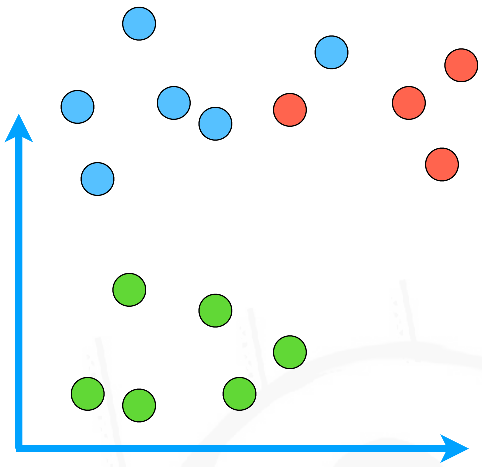



## If we maximize the distance between means and minimize scatter at the same time…

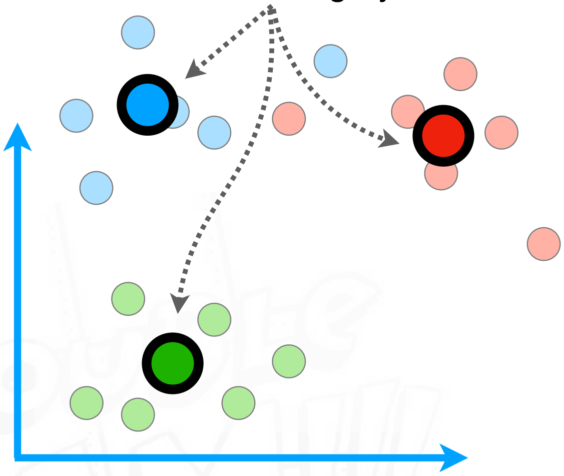



## NOTES:

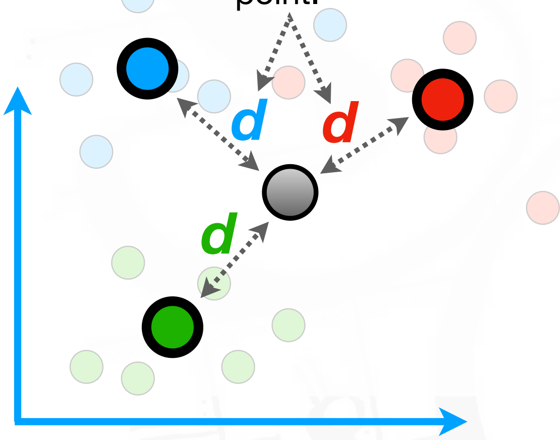
## LDA for 3 categories…



First, find the mean value for each category.



Then measure the distances, ***d***, between each mean and a central point.

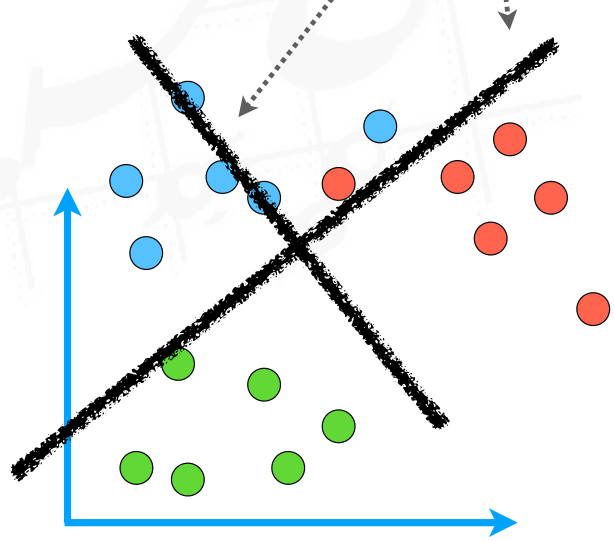



The sum of the squared distances from the central point is called the ***between class scatter***…

$$\frac{d^2 + d^2 + d^2}{s^2 + s^2 + s^2}$$

…and the sum of the scatter within each category is called the ***within class scatter***.

**LDA** finds new axes that maximize the ratio between ***between class scatter*** and ***within class scatter***.



**NOTES:**

**1)** When there are **3** categories, **LDA** finds **2** new axes because the **3** means for each category define a plane. If there are ***n*** categories, **LDA** finds ***n*-1** new axes.

**2)** Just like **PCA**, **LDA** ranks the axes in order of importance. So we can use **LDA** to reduce dimensions just like **PCA**.

**3)** Just like **PCA**, the new axes created by **LDA** have loading scores that tell us which variables had the largest influence on each axis.