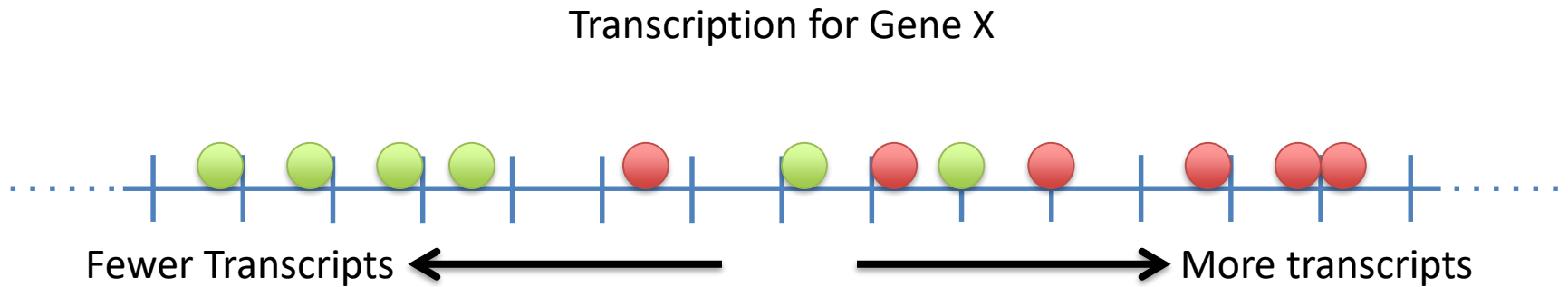


# Linear Discriminant Analysis

## The Problem...

- We've got a cancer drug
  - It works great for some people...
  - ... but it makes other people feel worse. :(
- How do we decide who to give the drug to?
  - Maybe gene expression can help us decide.

# Using one gene to decide...

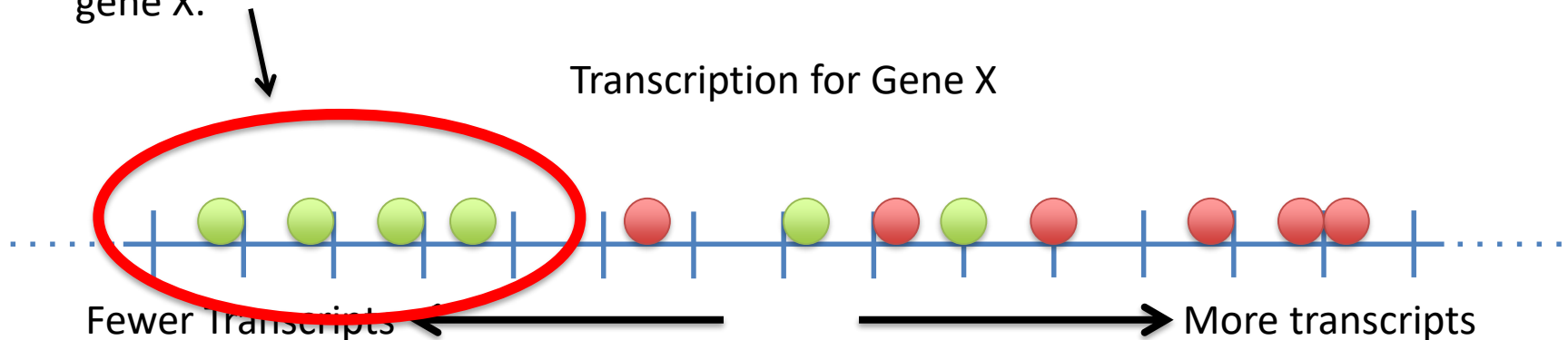


● = The drug works!

● = The drug does not work :(

# Using one gene to decide...

For the most part, the drug works in people with low transcription of gene X.



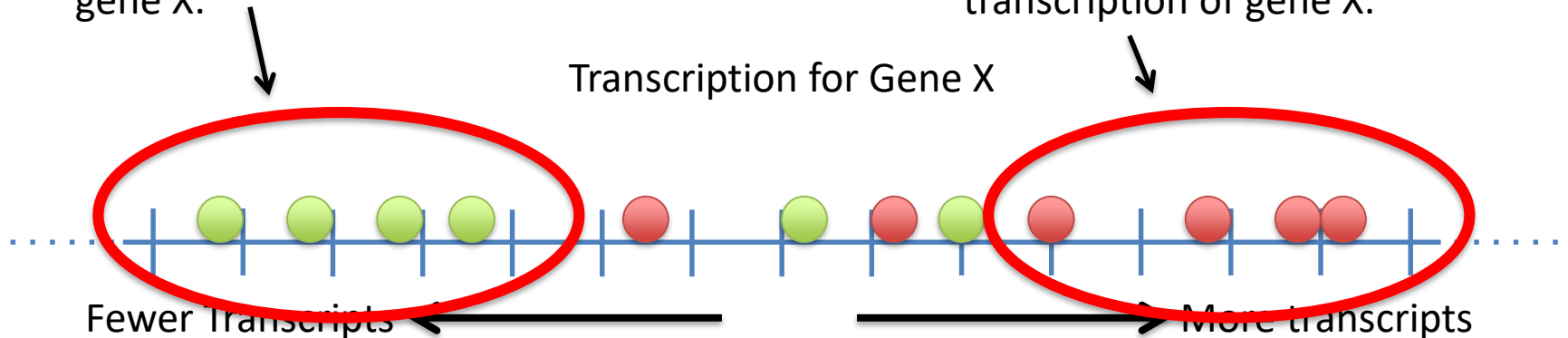
● = The drug works!

● = The drug does not work :(

# Using one gene to decide...

For the most part, the drug works in people with low transcription of gene X.

And, for the most part, the drug does not work in people with high transcription of gene X.

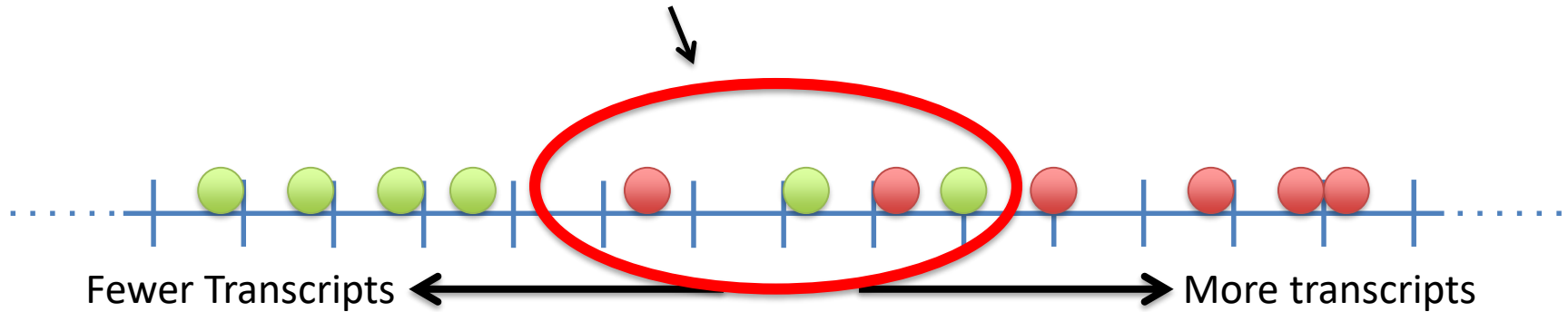


● = The drug works!

● = The drug does not work :(

# Using one gene to decide...

However, there is overlap and no obvious “cutoff” for who to give the drug to.

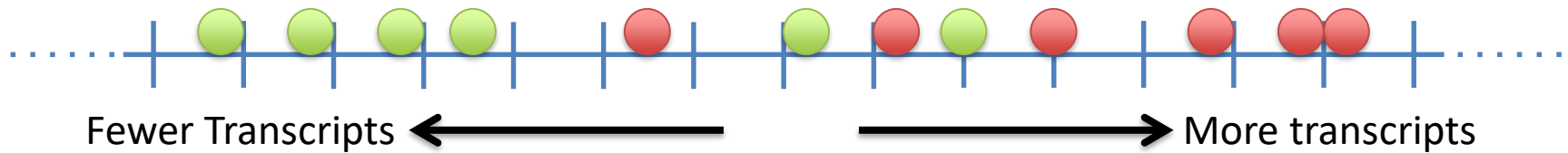


● = The drug works!

● = The drug does not work :(

# Using one gene to decide...

In summary, Gene X does an OK job at telling us who should take the drug (and who shouldn't).



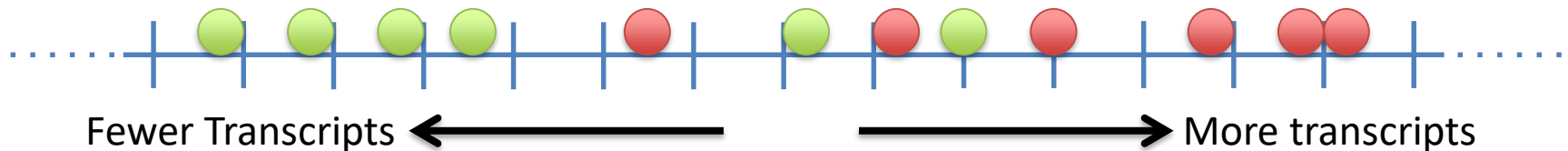
● = The drug works!

● = The drug does not work :(

# Using one gene to decide...

In summary, Gene X does an OK job at telling us who should take the drug (and who shouldn't).

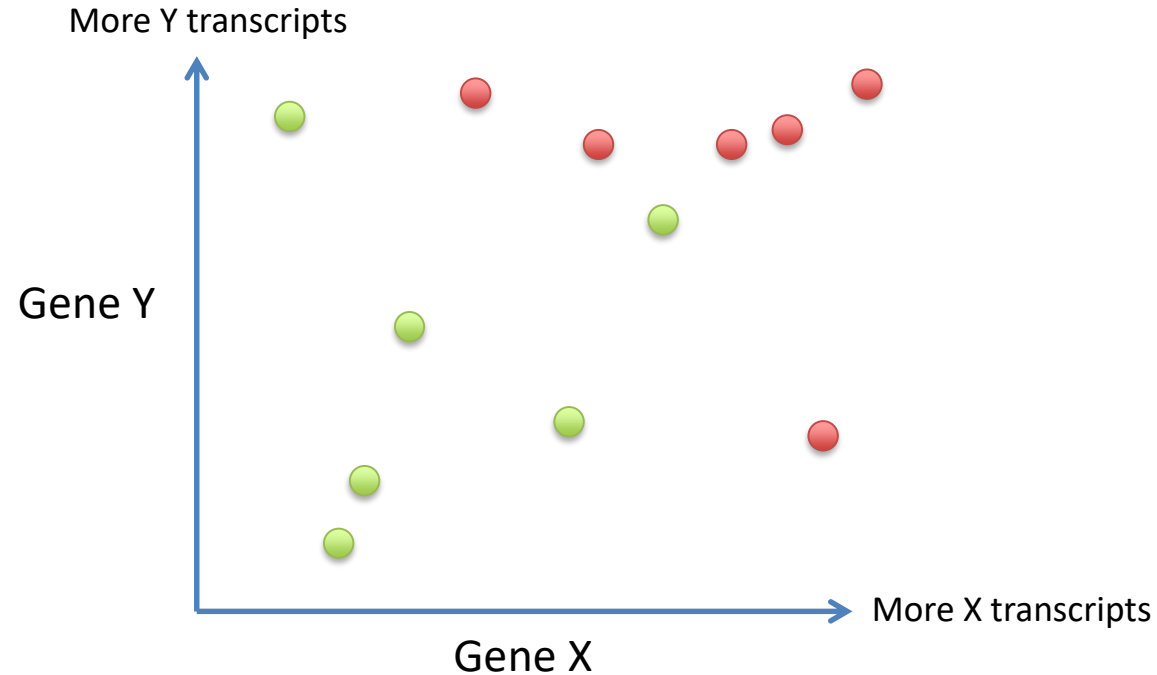
Can we do better? What if we used more than one gene?



● = The drug works!

● = The drug does not work :(

# Using two genes to decide...

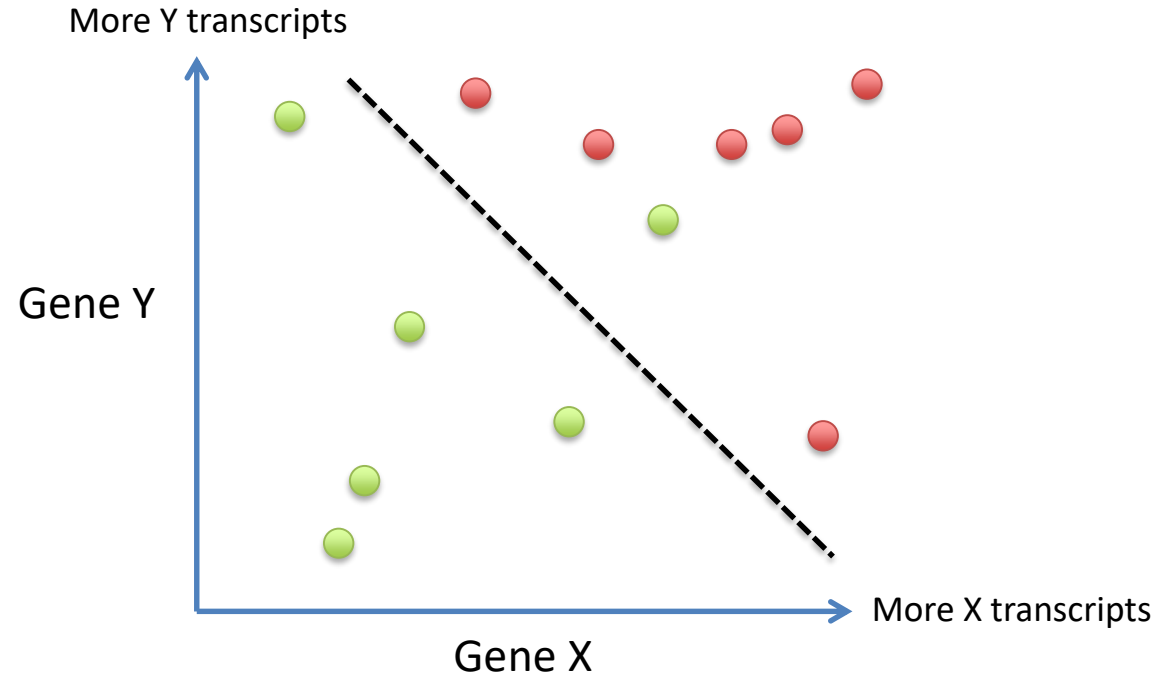


● = The drug works!

● = The drug does not work :(



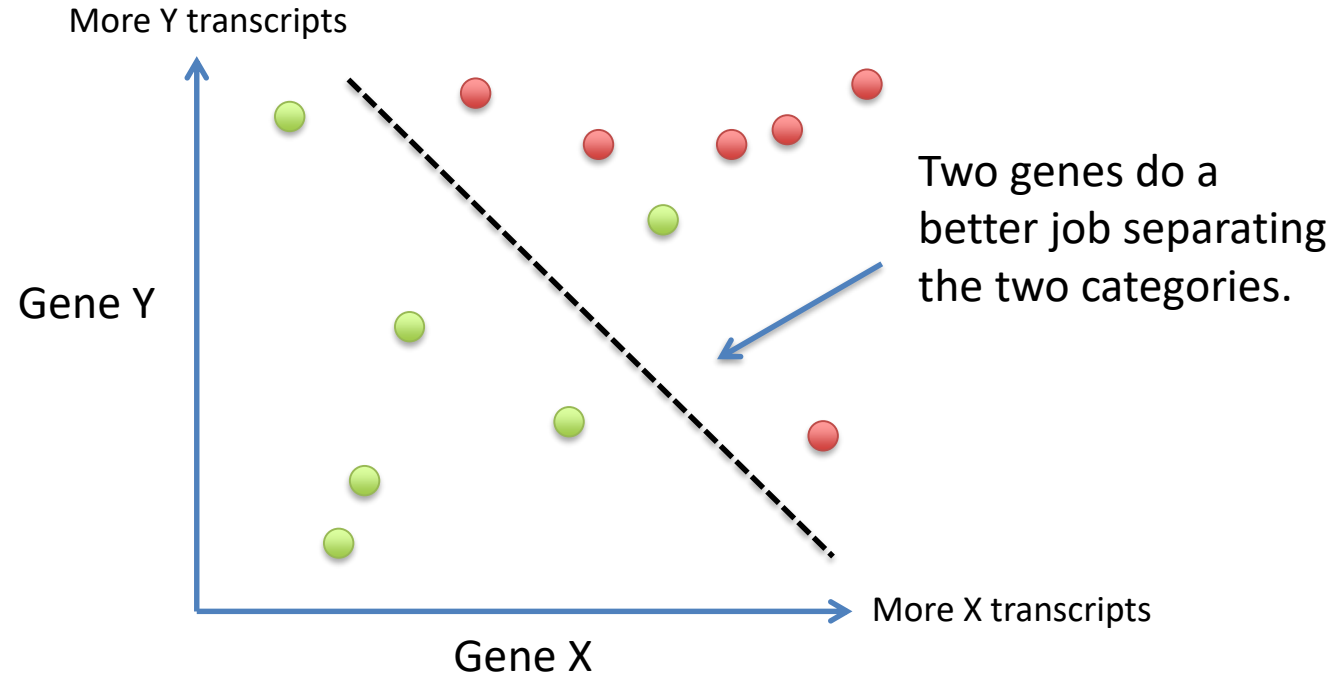
# Using two genes to decide...



● = The drug works!

● = The drug does not work :(

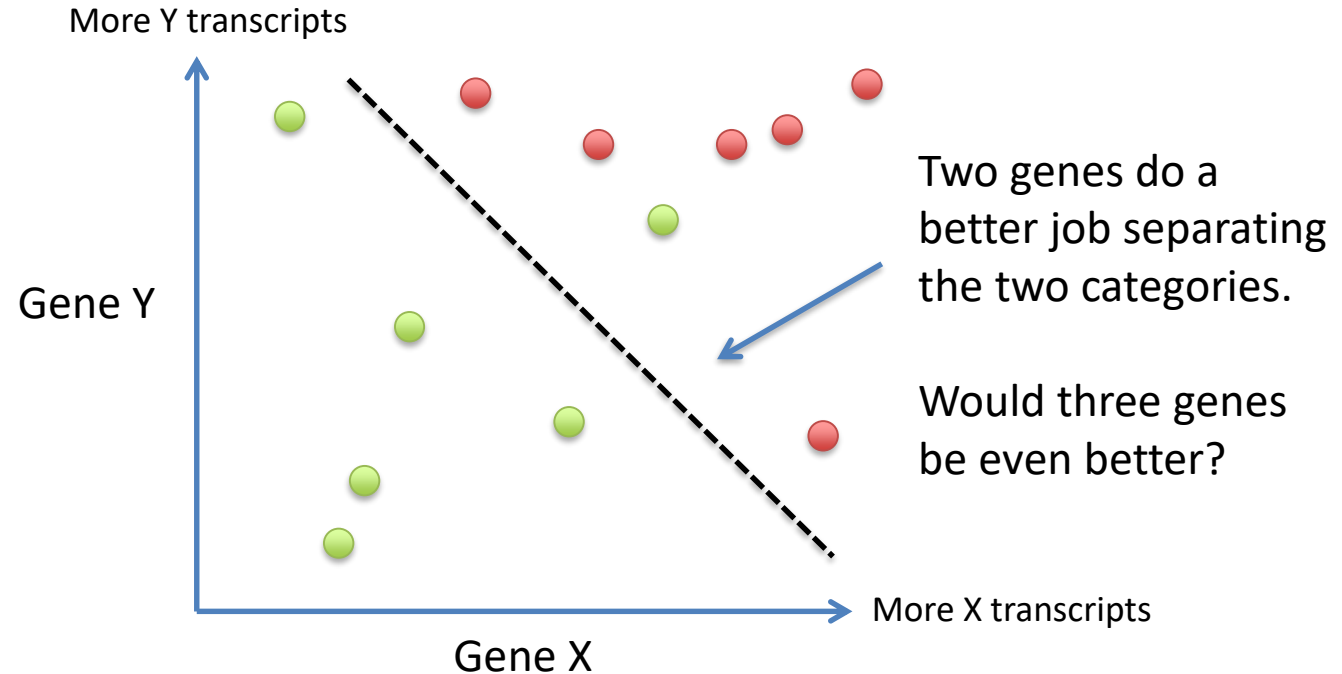
# Using two genes to decide...



● = The drug works!

● = The drug does not work :(

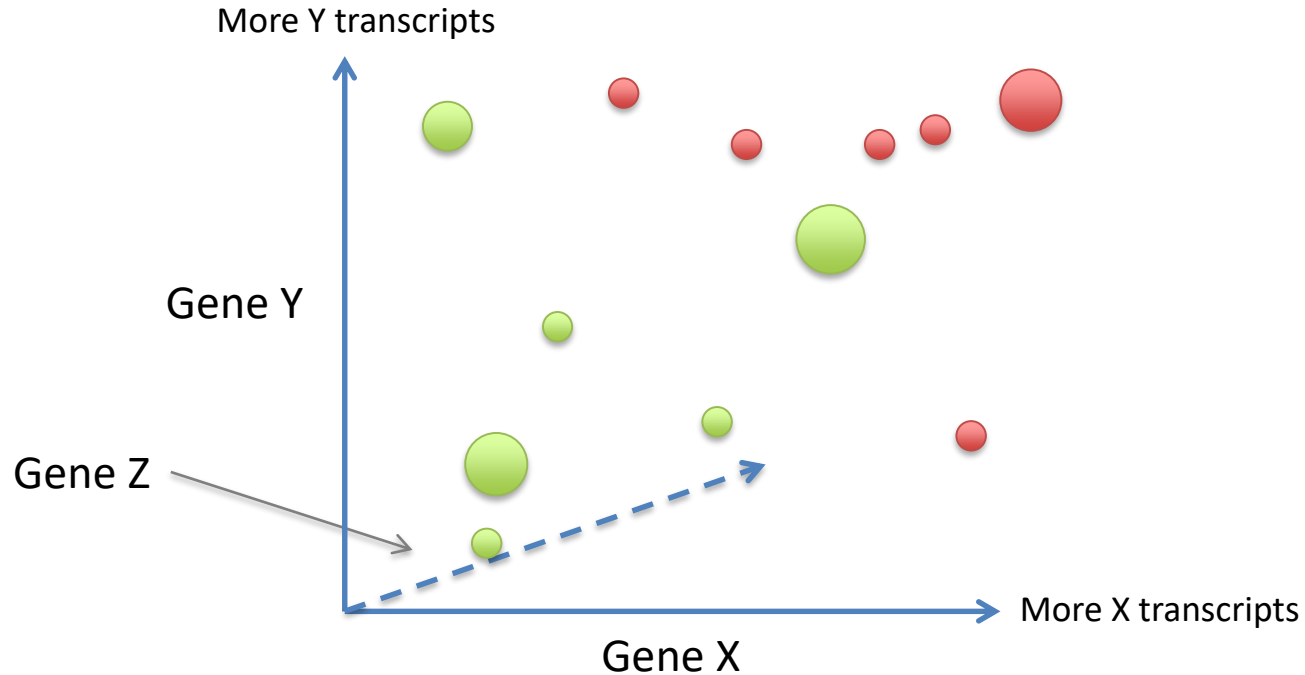
# Using two genes to decide...



● = The drug works!

● = The drug does not work :(

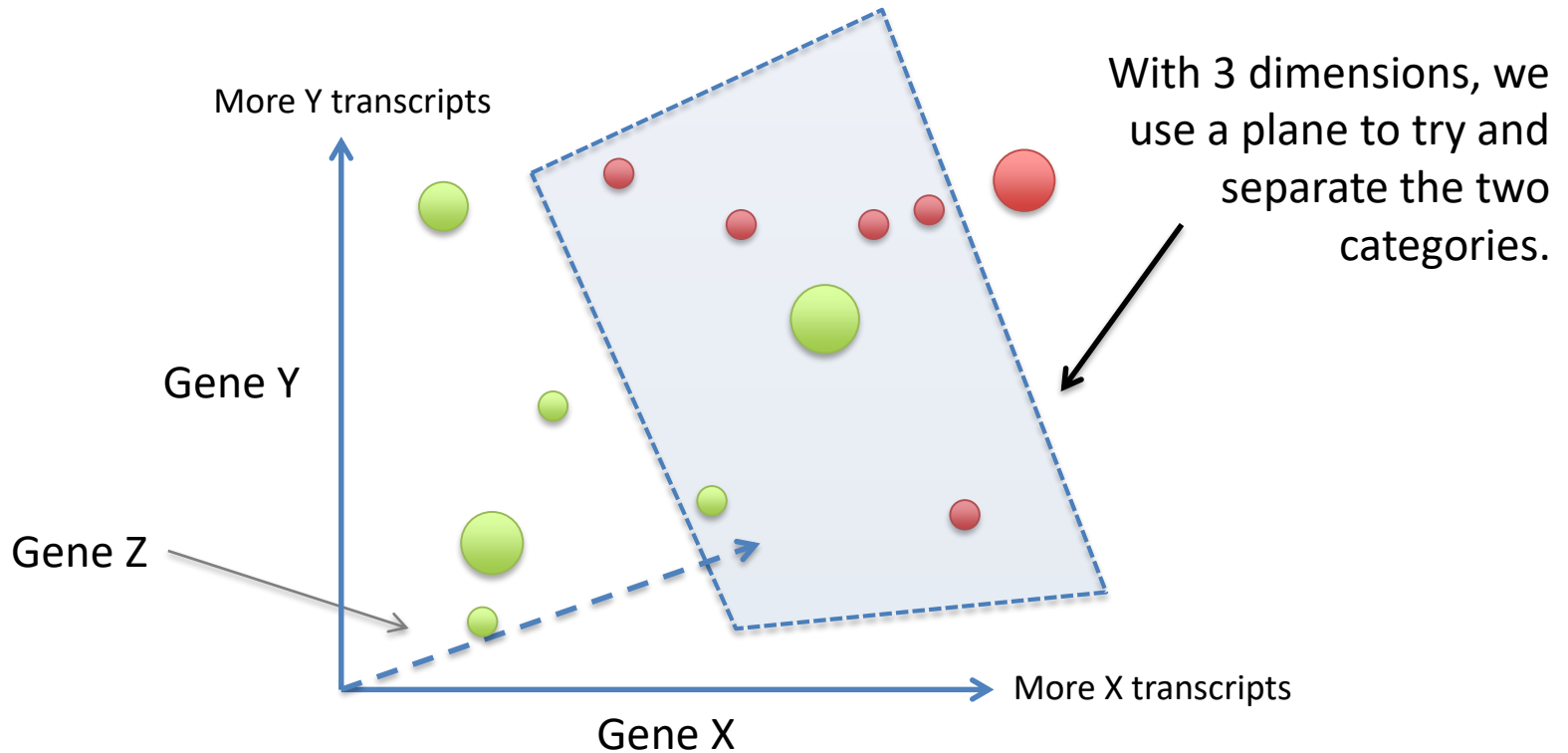
# Using three genes to decide...



● = The drug works!

● = The drug does not work :(

# Using three genes to decide...



● = The drug works!

● = The drug does not work :(

What if we need four (or more) genes to separate the two categories?

# Four or more genes...

- We can't draw a 4-D graph.... :(
- We ran into this same problem when we talked about PCA.
- PCA (remember that?) reduces dimensions by focusing on the genes with the most variation.
  - This is useful for plotting data with a lot of dimensions (or a lot of genes) onto a simple X/Y plot.
  - However, in this case we're not super interested in the genes with the most variation.
  - Instead, we're interested maximizing the separability between the two groups so we can make the best decisions.
- Linear Discriminant Analysis (LDA) is like PCA, but it focuses on maximizing the separability among known categories.

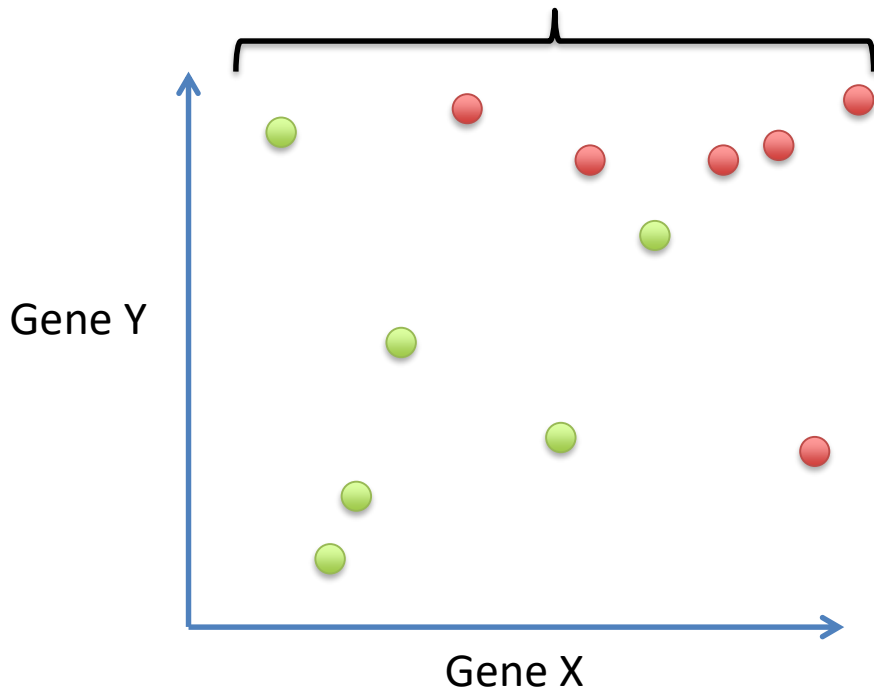
Linear Discriminant Analysis (LDA) is like PCA, but it focuses on maximizing the separability among known categories.



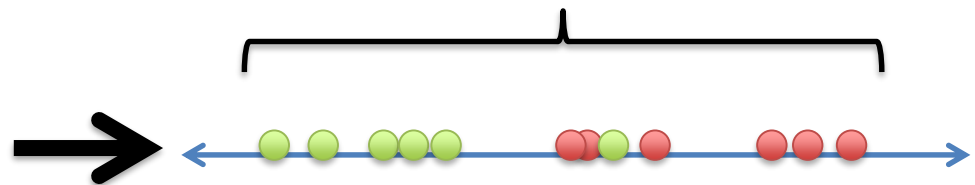
# A super simple example

Reducing a 2-D graph to a 1-D graph

A 2-D graph (aka X/Y Graph)



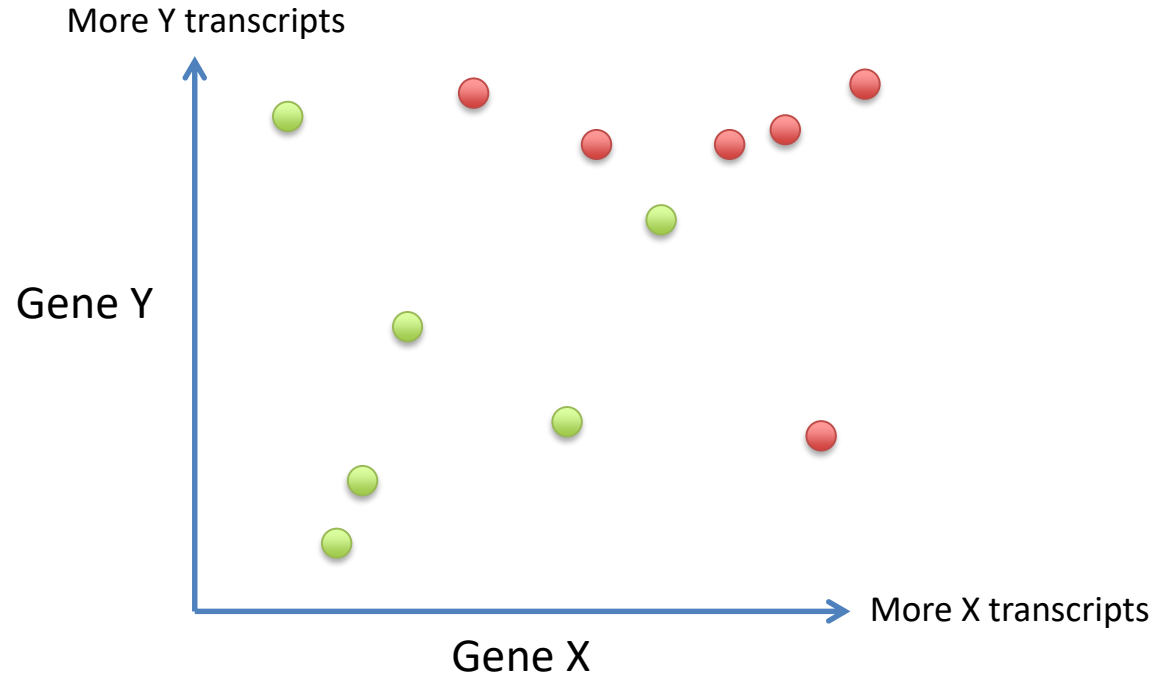
A 1-D graph (aka number line)



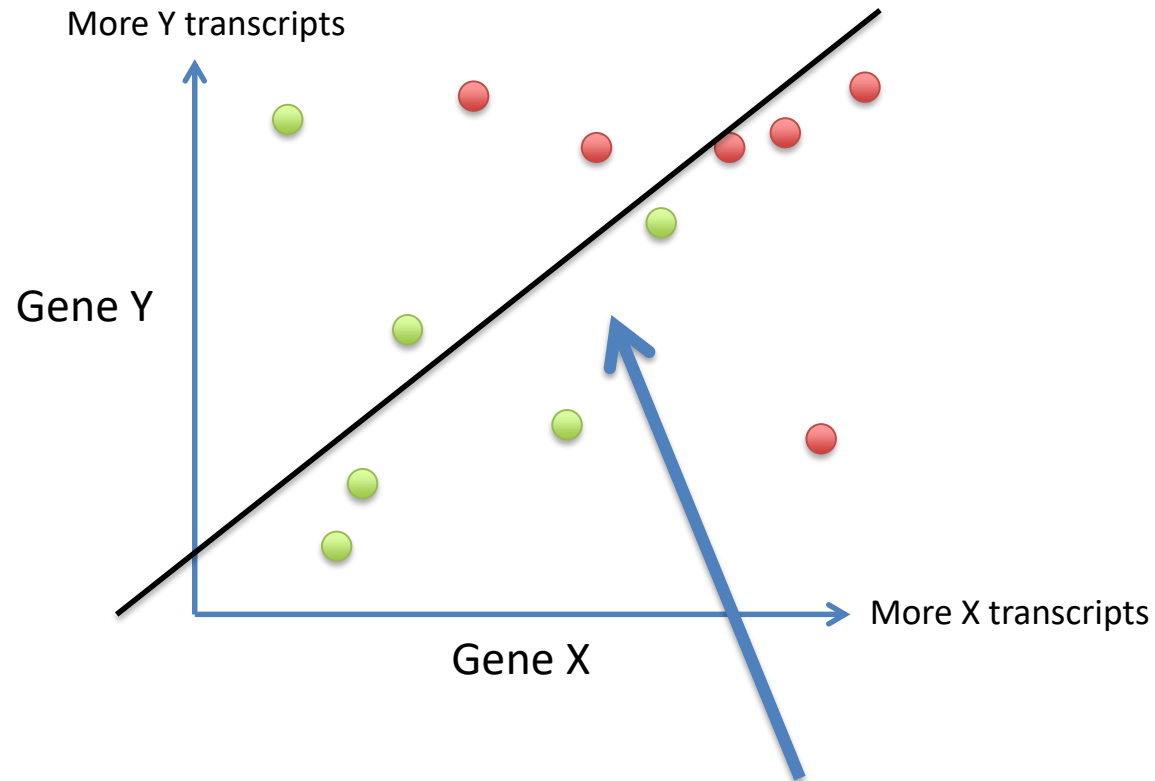
LDA provides a way to do this.

Let's see what it does, then we'll go over the details.

# Reducing a 2-D graph to a 1-D graph with LDA

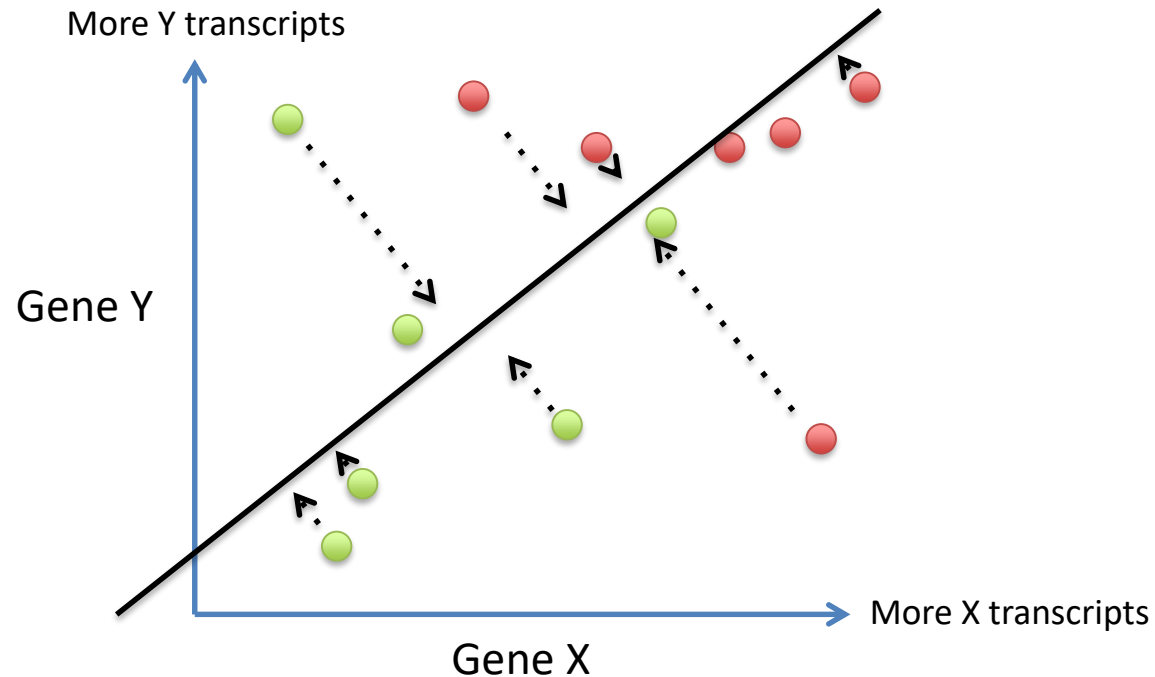


# Reducing a 2-D graph to a 1-D graph with LDA



LDA uses both genes to create a new axis...

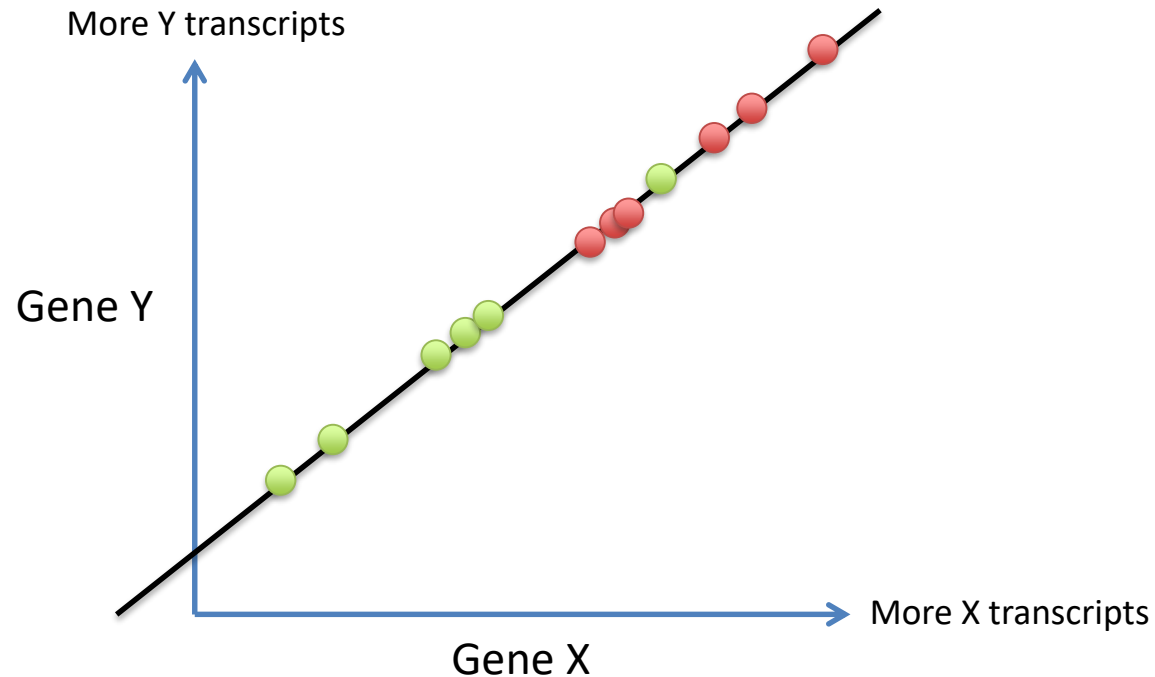
# Reducing a 2-D graph to a 1-D graph with LDA



LDA uses both genes to create a new axis...

...and projects the data onto this new axis in a way to maximize the separation of the two categories.

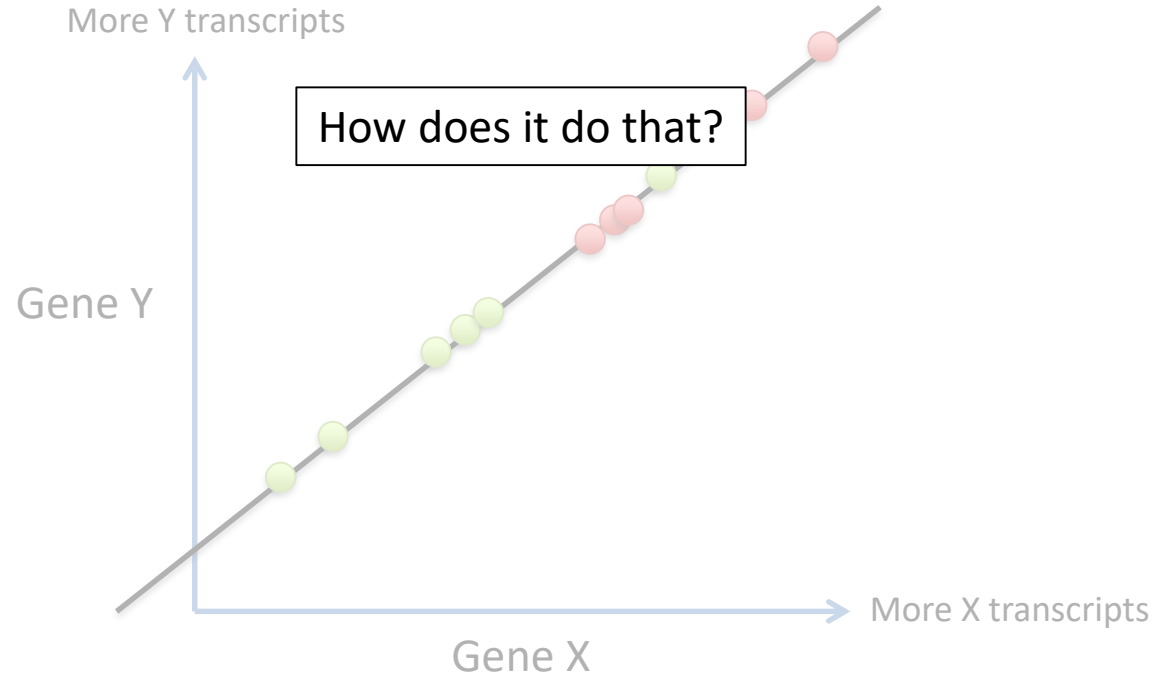
# Reducing a 2-D graph to a 1-D graph with LDA



LDA uses both genes to create a new axis...

...and projects the data onto this new axis in a way to maximize the separation of the two categories.

# Reducing a 2-D graph to a 1-D graph with LDA



LDA uses both genes to create a new axis...

...and projects the data onto this new axis in a way to maximize the separation of the two categories.

# How LDA creates a new axis...



# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

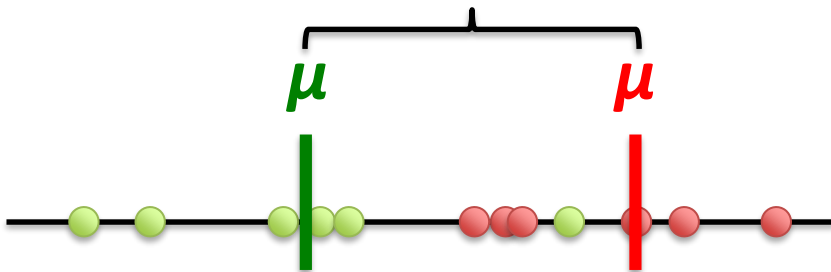




# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

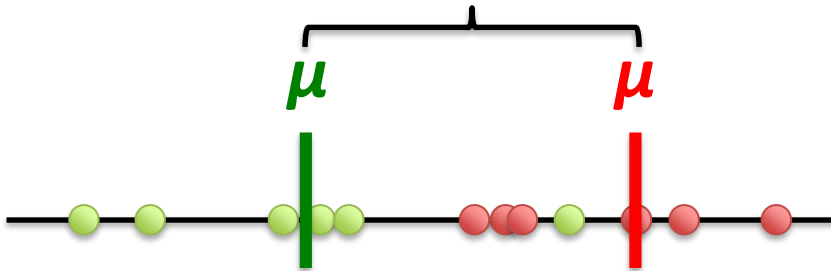
- 1) Maximize the distance between means.



# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

- 1) Maximize the distance between means.

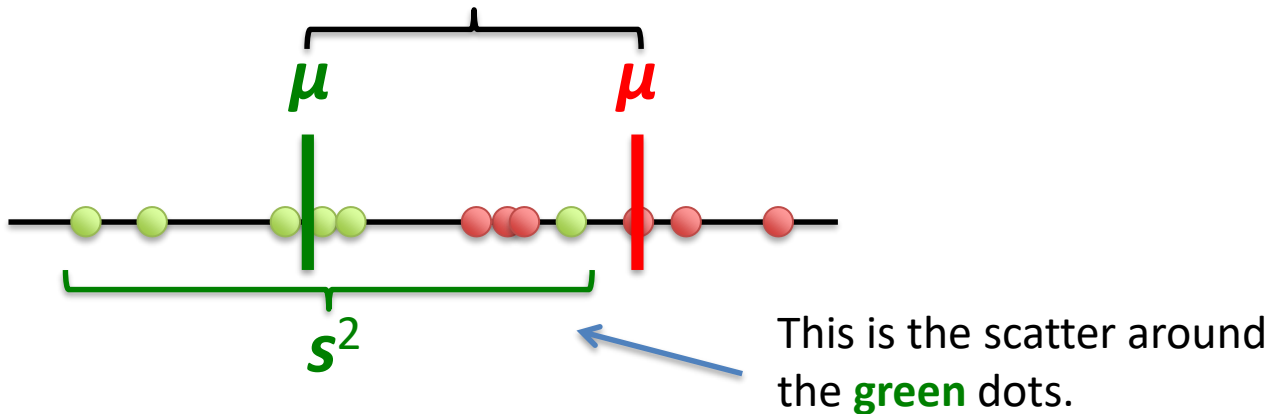


- 2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

- 1) Maximize the distance between means.

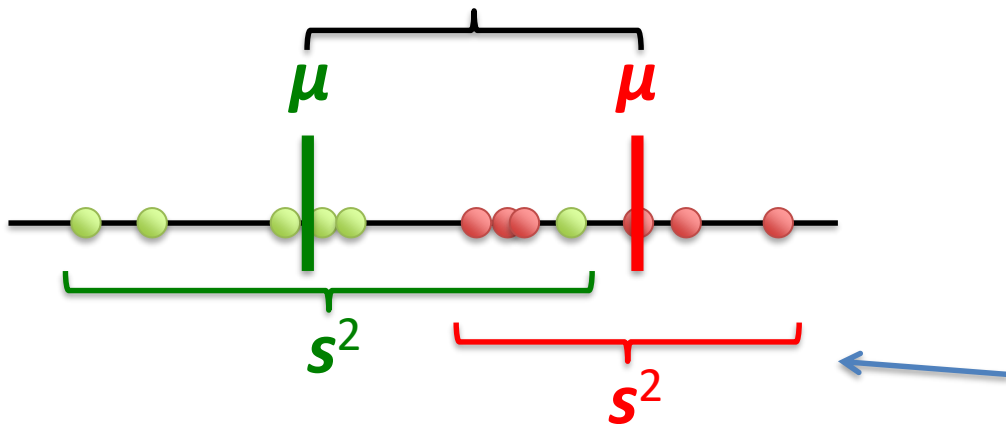


- 2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

- 1) Maximize the distance between means.



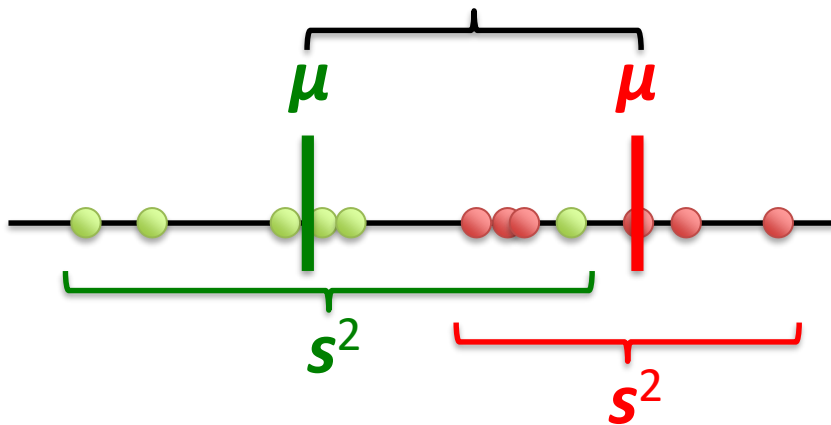
This is the scatter around the **red** dots.

- 2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

- 1) Maximize the distance between means.



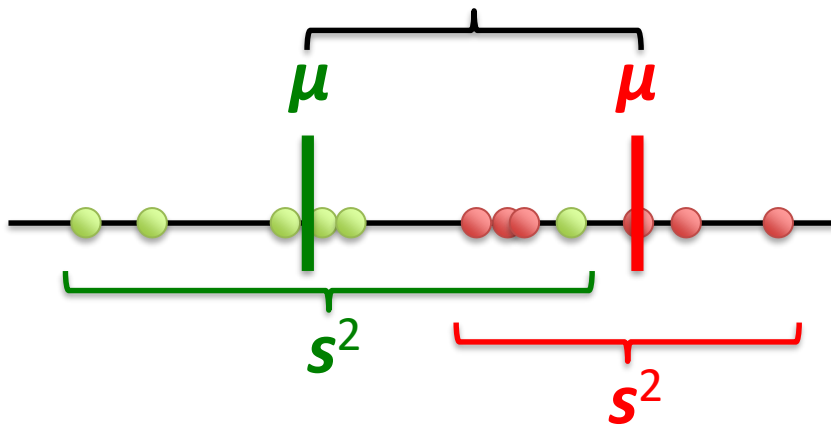
$$\frac{(\mu - \mu)^2}{s^2 + s^2}$$

- 2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

1) Maximize the distance between means.



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

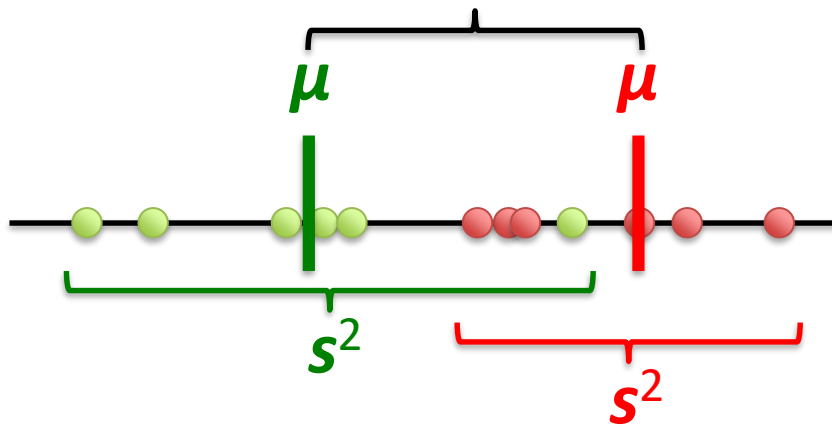
Ideally large  
Ideally small

2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

1) Maximize the distance between means.



Let's call  $(\mu - \mu)$   
 $d$  for  $d$ istance.

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

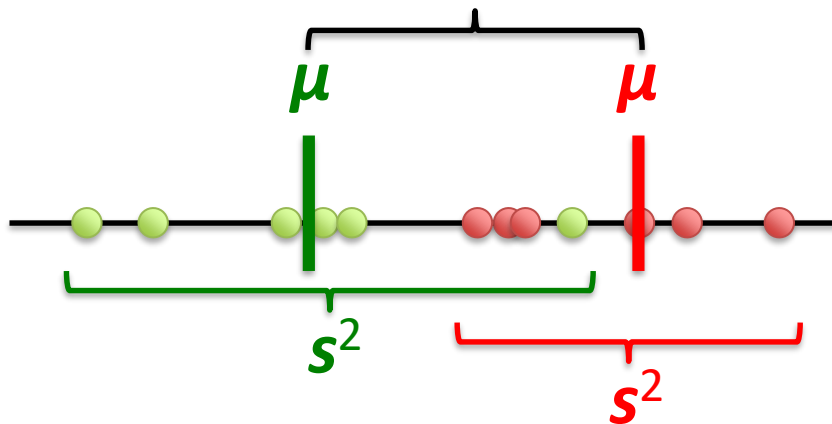
Ideally large  
Ideally small

2) Minimize the variation (which LDA calls “scatter” and is represented by  $s^2$ ) within each category.

# How LDA creates a new axis...

The new axis is created according to two criteria (considered simultaneously):

1) Maximize the distance between means.



Let's call  $(\mu - \mu)$   $d$  for *distance*.

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

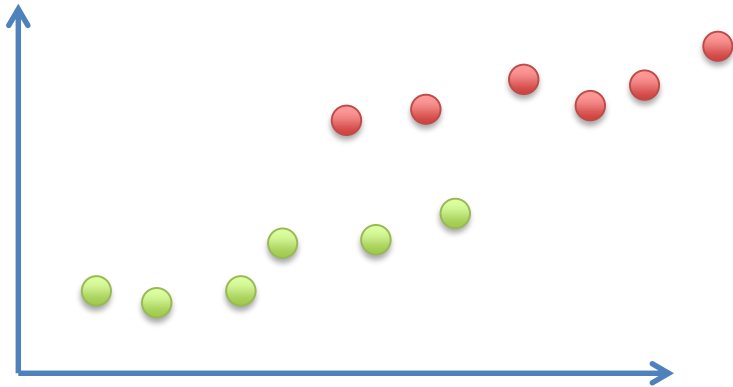
Ideally large  
Ideally small

2) Minimize the variation (which LDA calls "scatter" and is represented by  $s^2$ ) within each category.



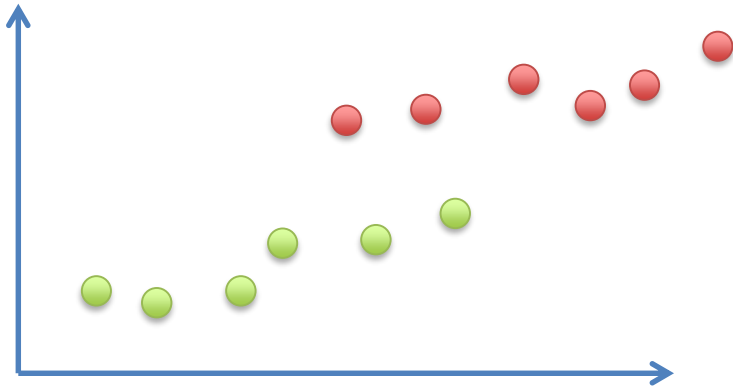
An example showing why both distance and scatter are important.

# An example showing why both distance and scatter are important.

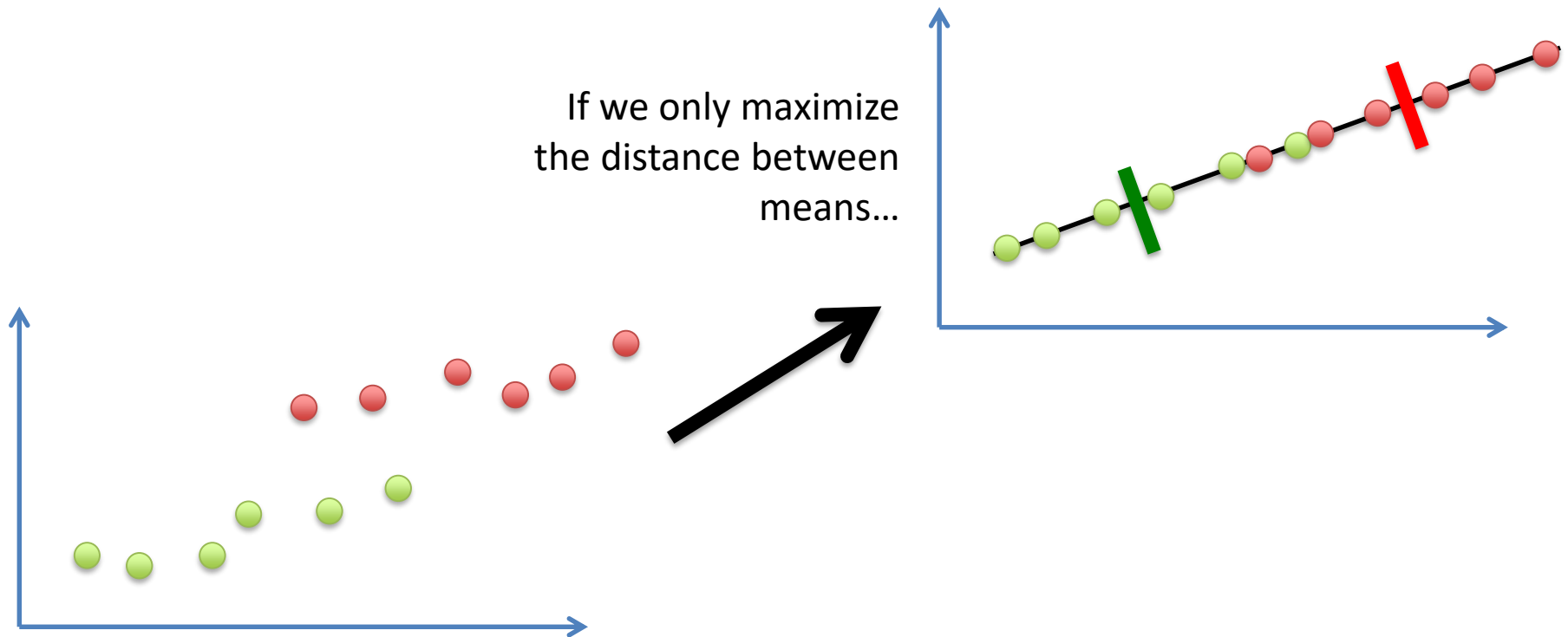


# An example showing why both distance and scatter are important.

If we only maximize  
the distance between  
means...

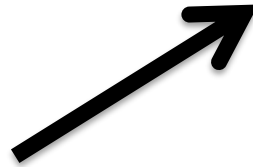
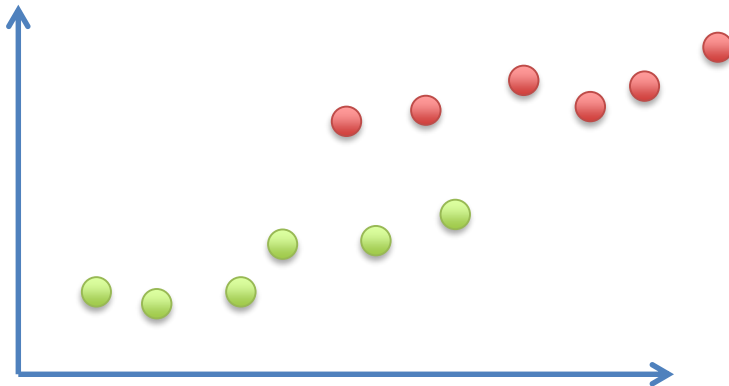


# An example showing why both distance and scatter are important.

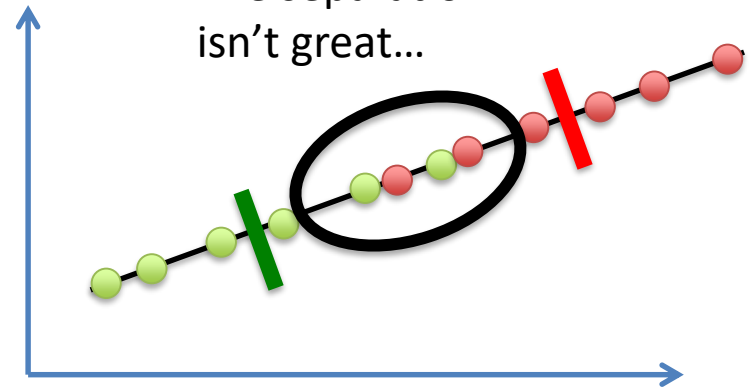


# An example showing why both distance and scatter are important.

If we only maximize the distance between means...

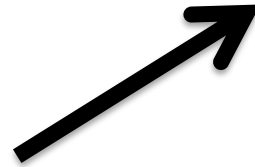
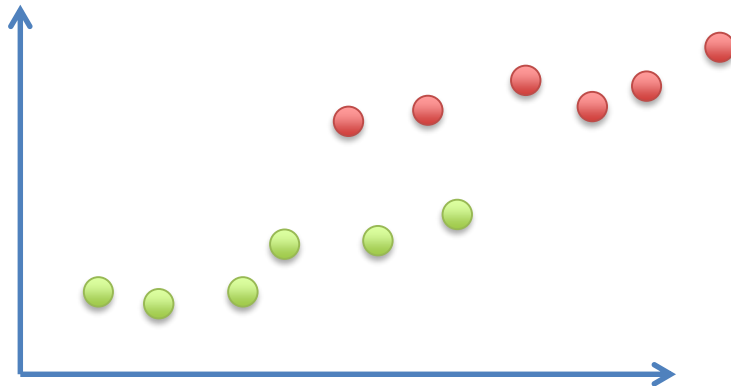


The separation isn't great...

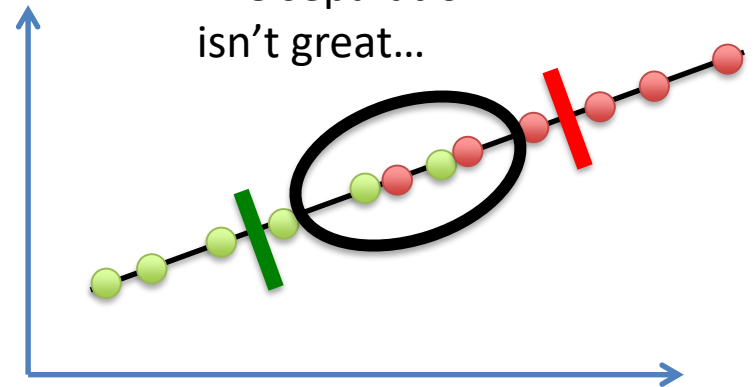


# An example showing why both distance and scatter are important.

If we only maximize the distance between means...



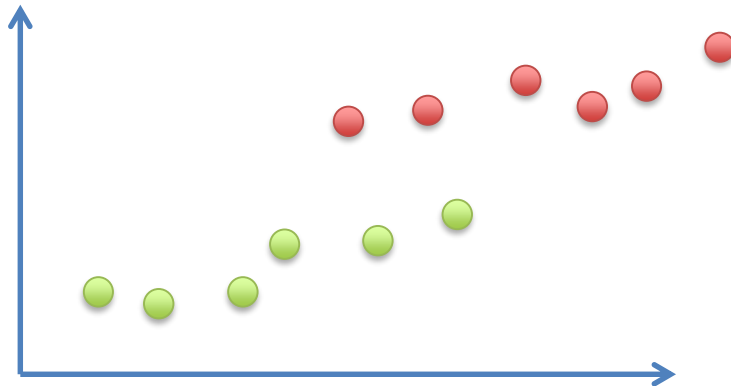
The separation isn't great...



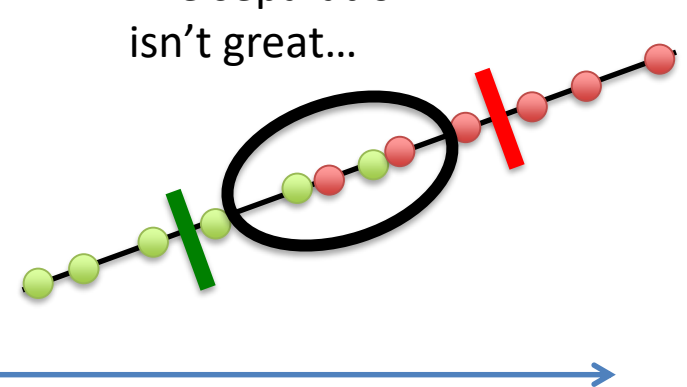
However, if we optimize the distance between means and scatter...

# An example showing why both distance and scatter are important.

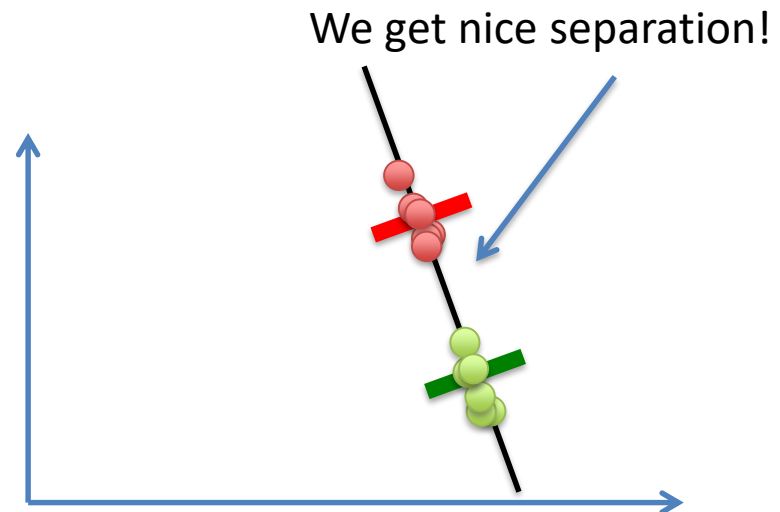
If we only maximize the distance between means...



The separation isn't great...



However, if we optimize the distance between means and scatter...

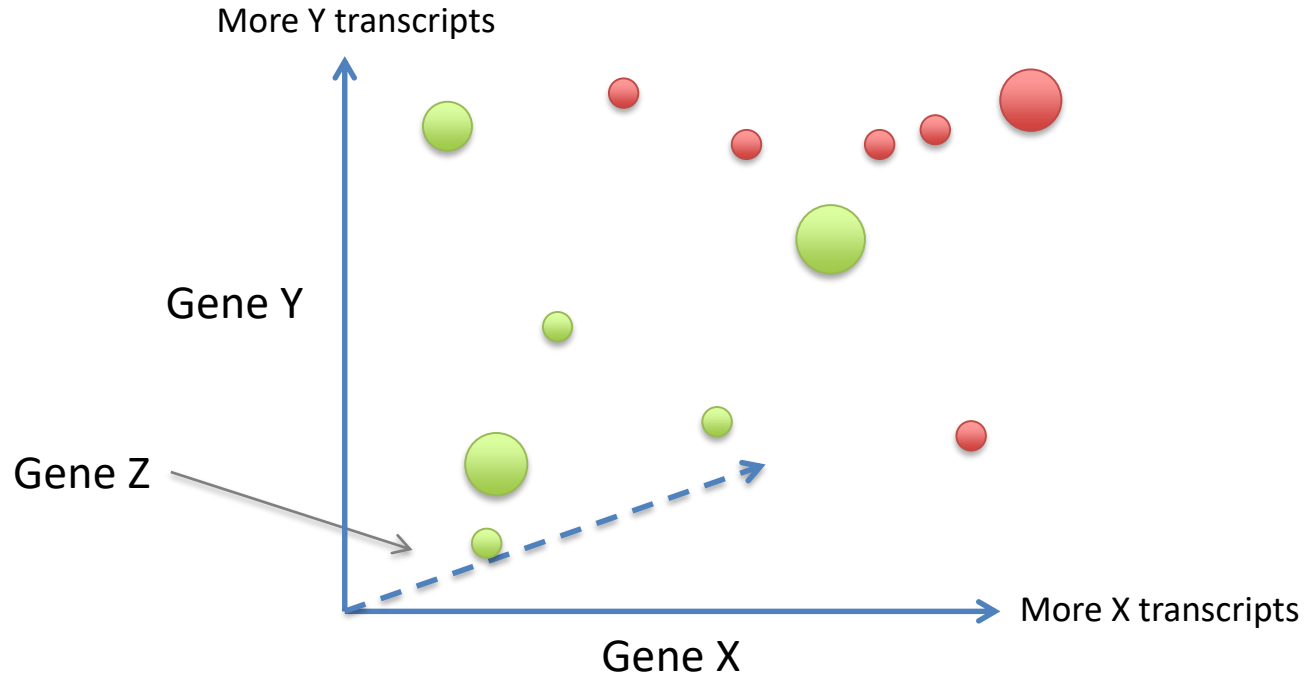


# What if we have more than 2 genes (more than 2 dimensions)?

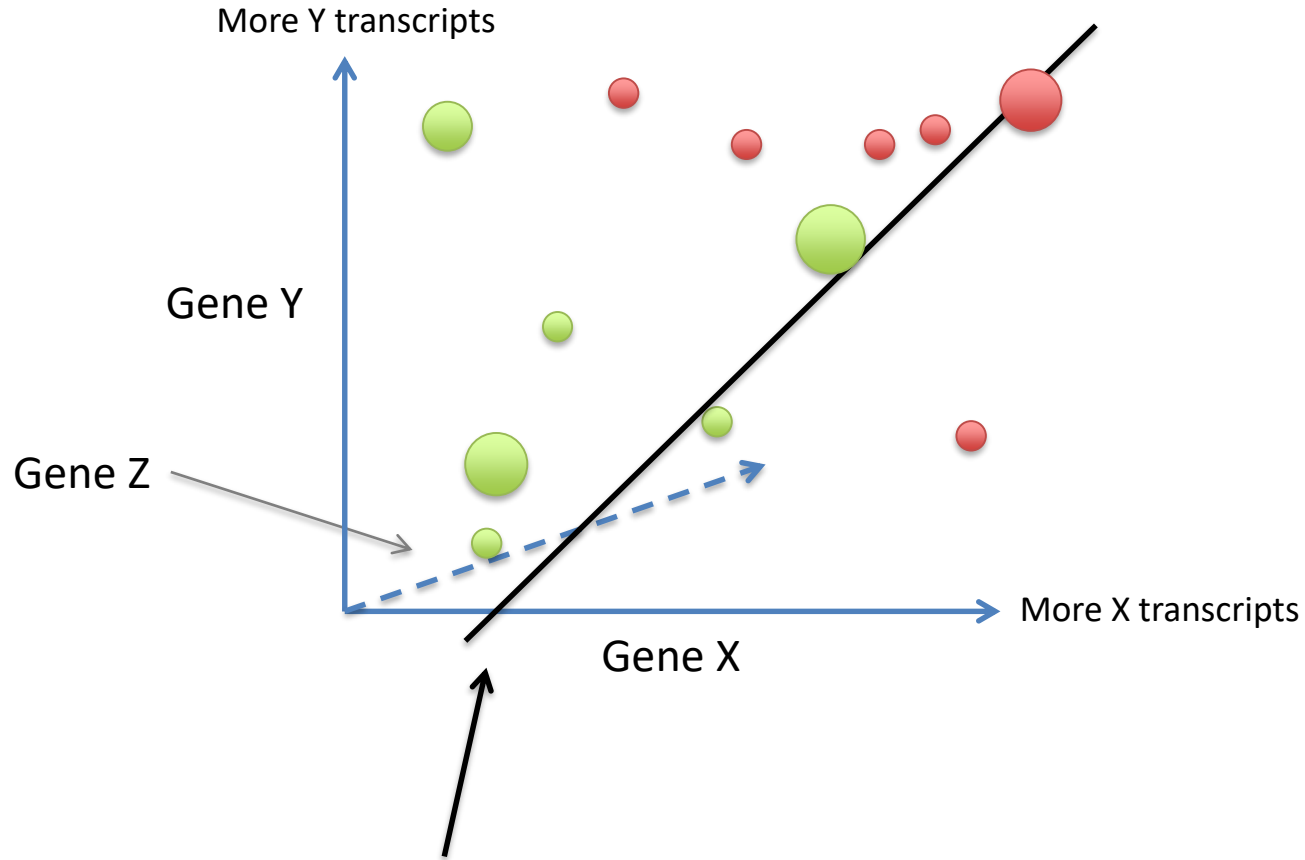
- The process is the same:
  - Create an axis that maximizes the distance between the means for the two categories while minimizing the scatter.



# LDA with 3 genes.



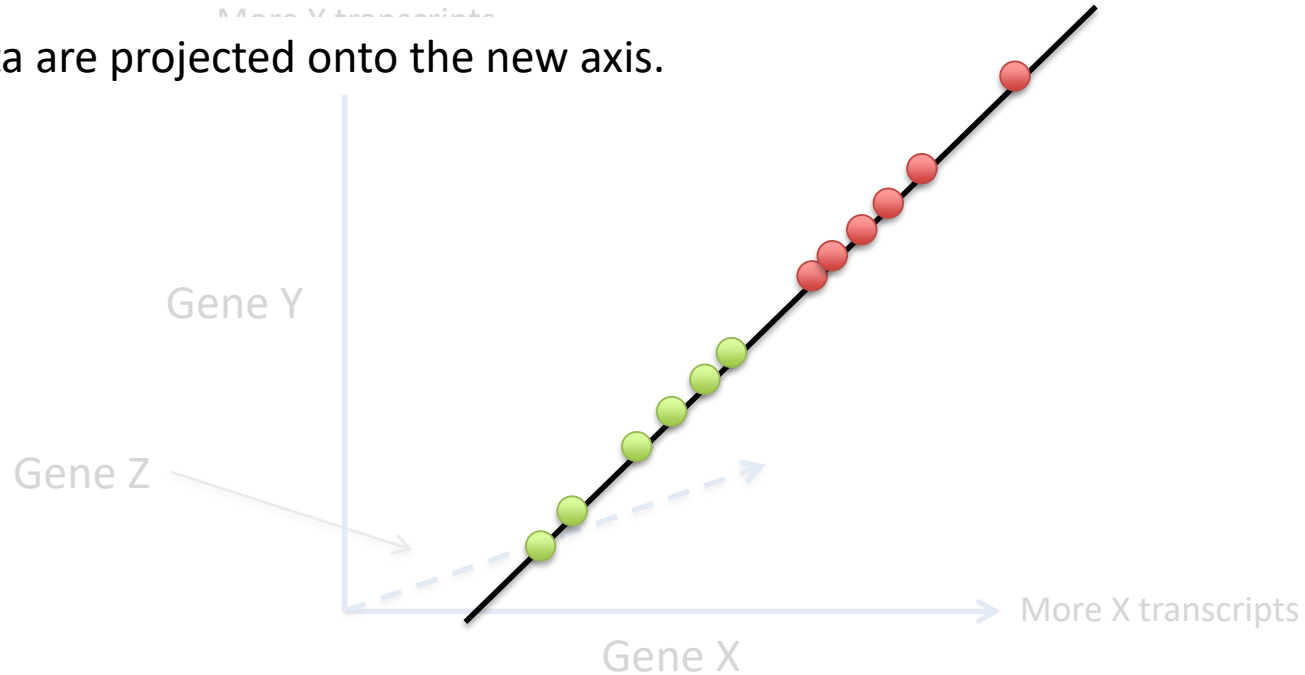
# LDA with 3 genes.



Here's the new axis that LDA created.

# LDA with 3 genes.

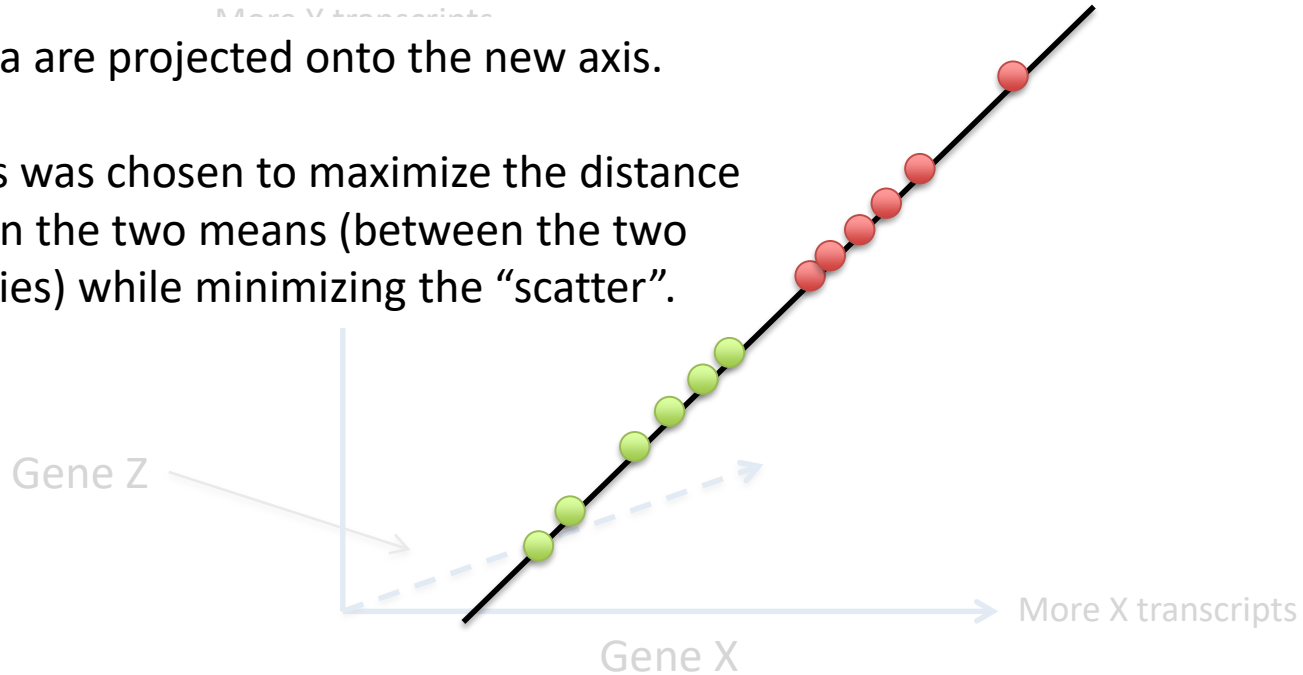
The data are projected onto the new axis.



# LDA with 3 genes.

The data are projected onto the new axis.

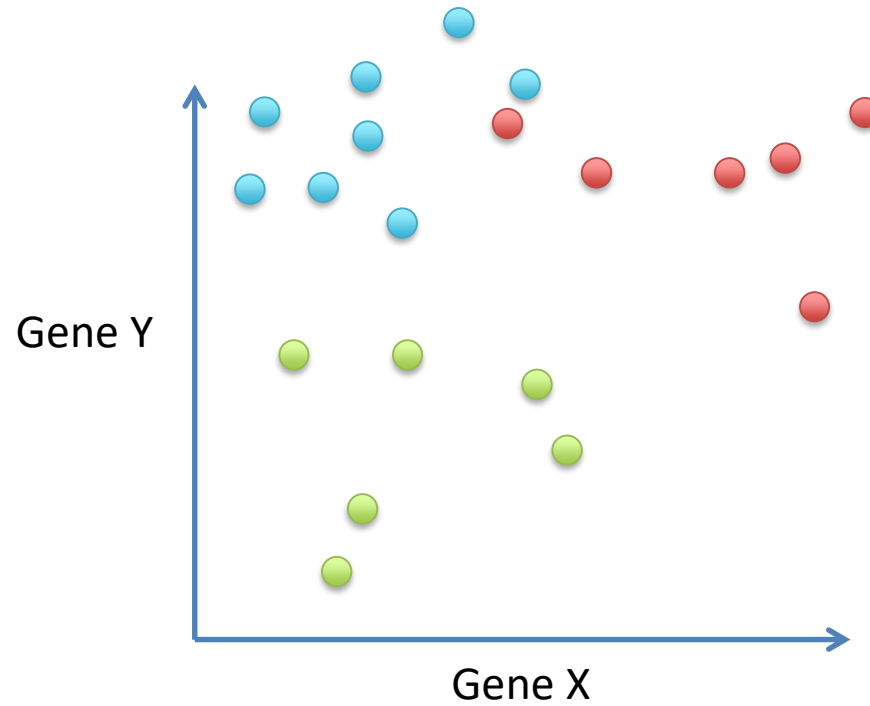
The axis was chosen to maximize the distance between the two means (between the two categories) while minimizing the “scatter”.



# What if we have 3 categories?

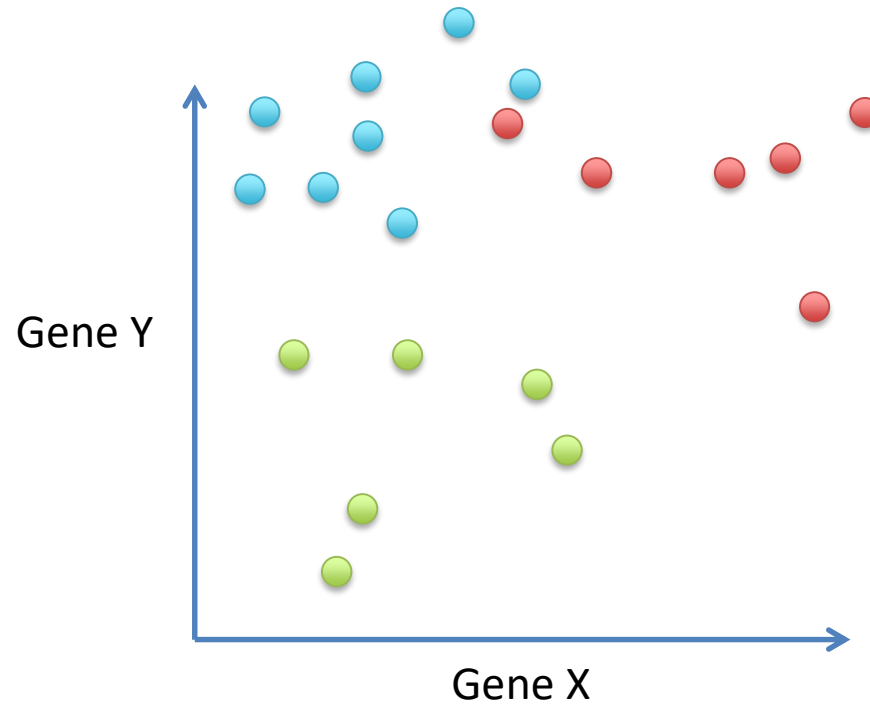
- Two things change, but barely...

# LDA for 3 categories

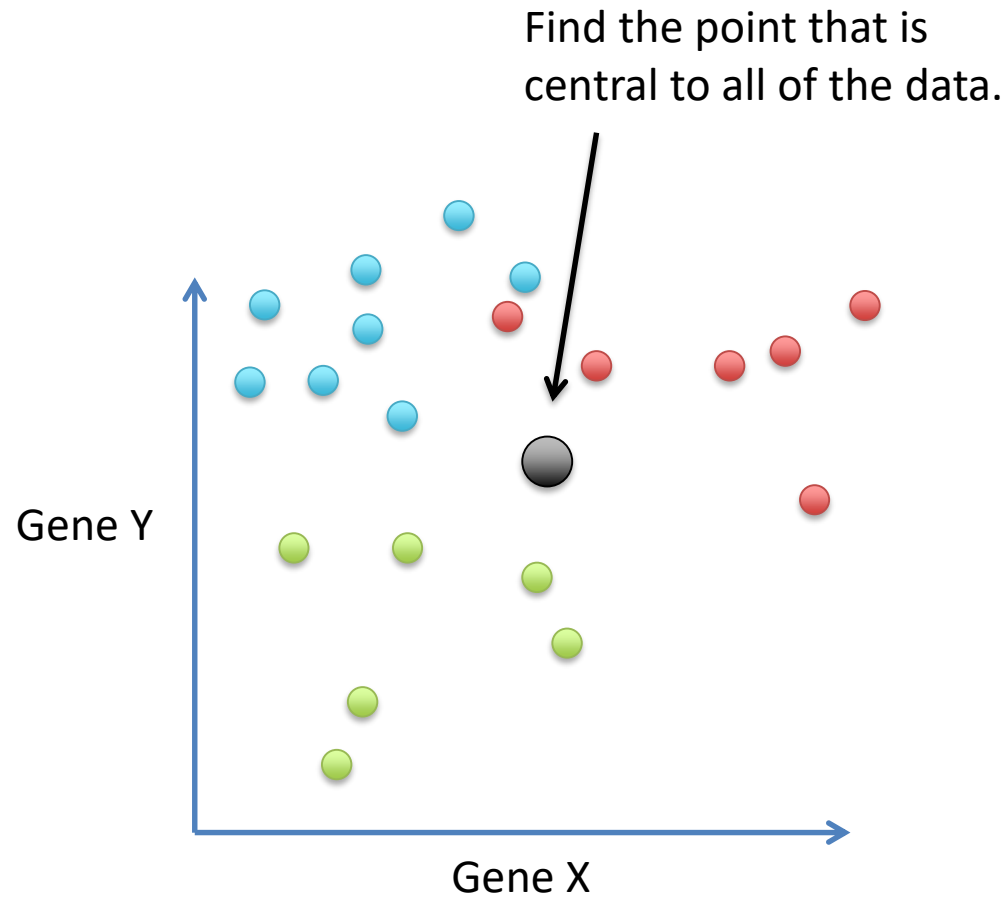


# LDA for 3 categories

**The first difference** is how you measure the distances among the means.



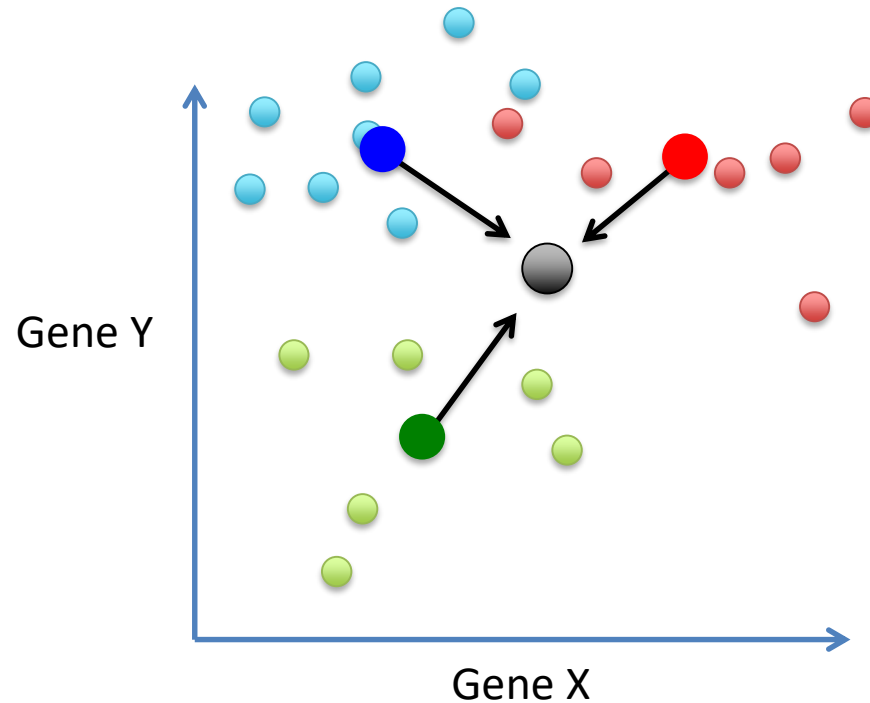
# LDA for 3 categories





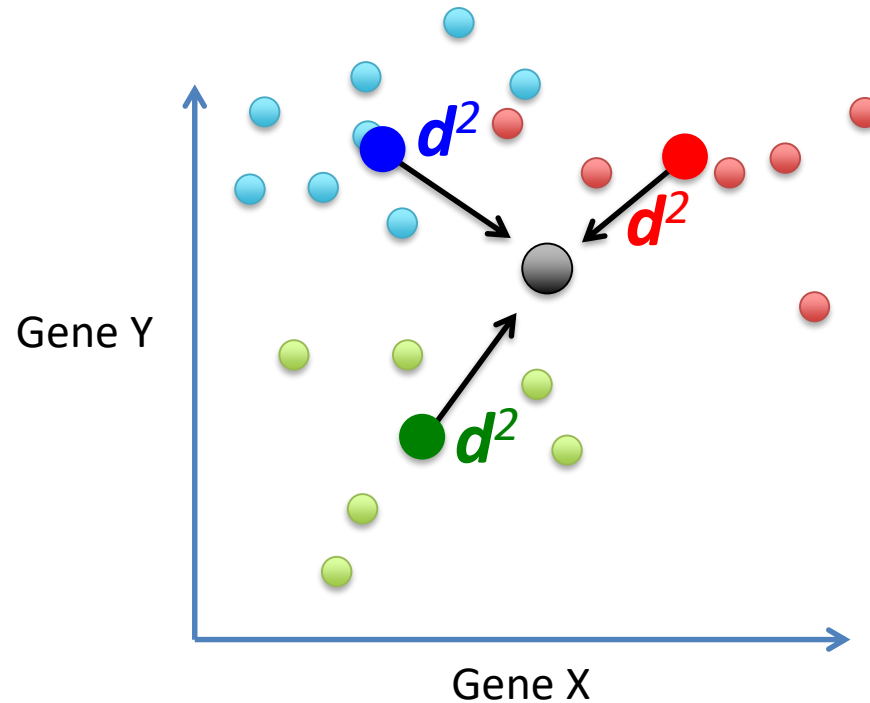
# LDA for 3 categories

Then measure the distances between a point that is central in each category and the main central point.

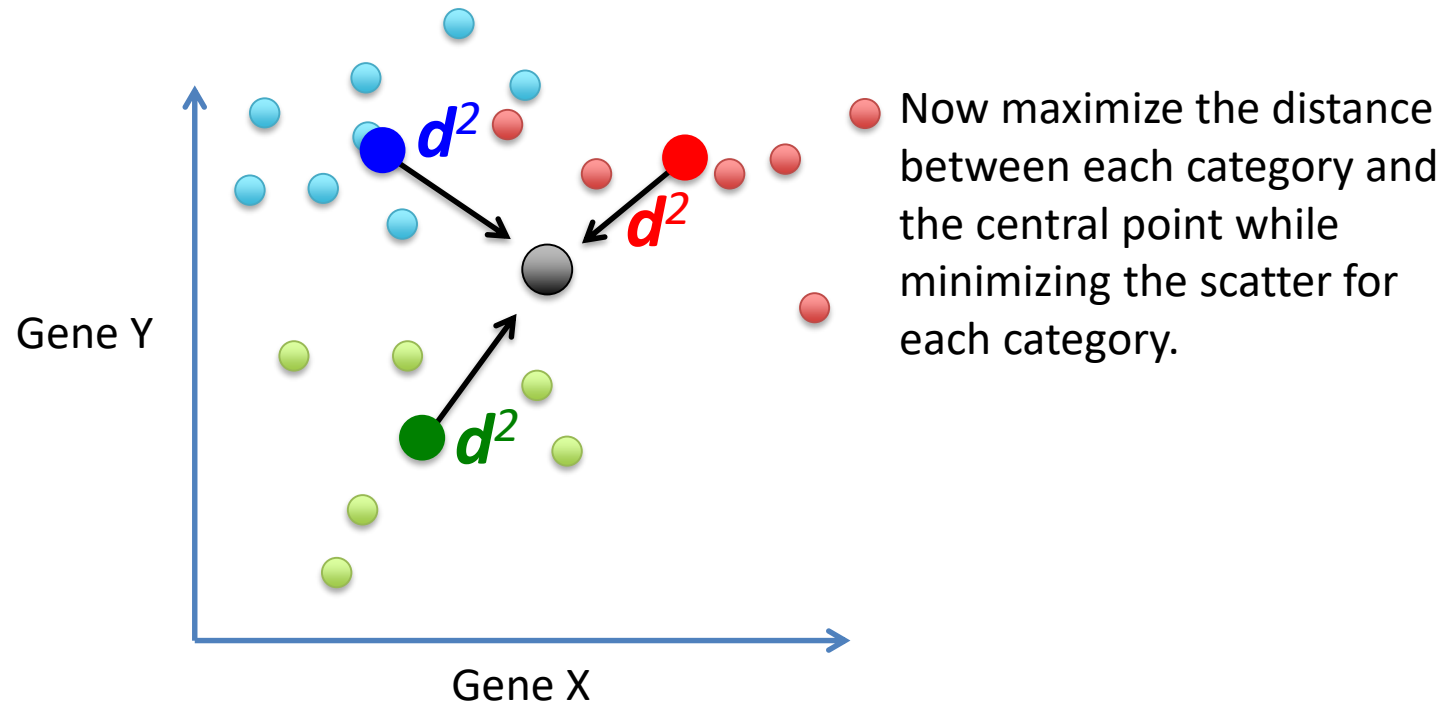


# LDA for 3 categories

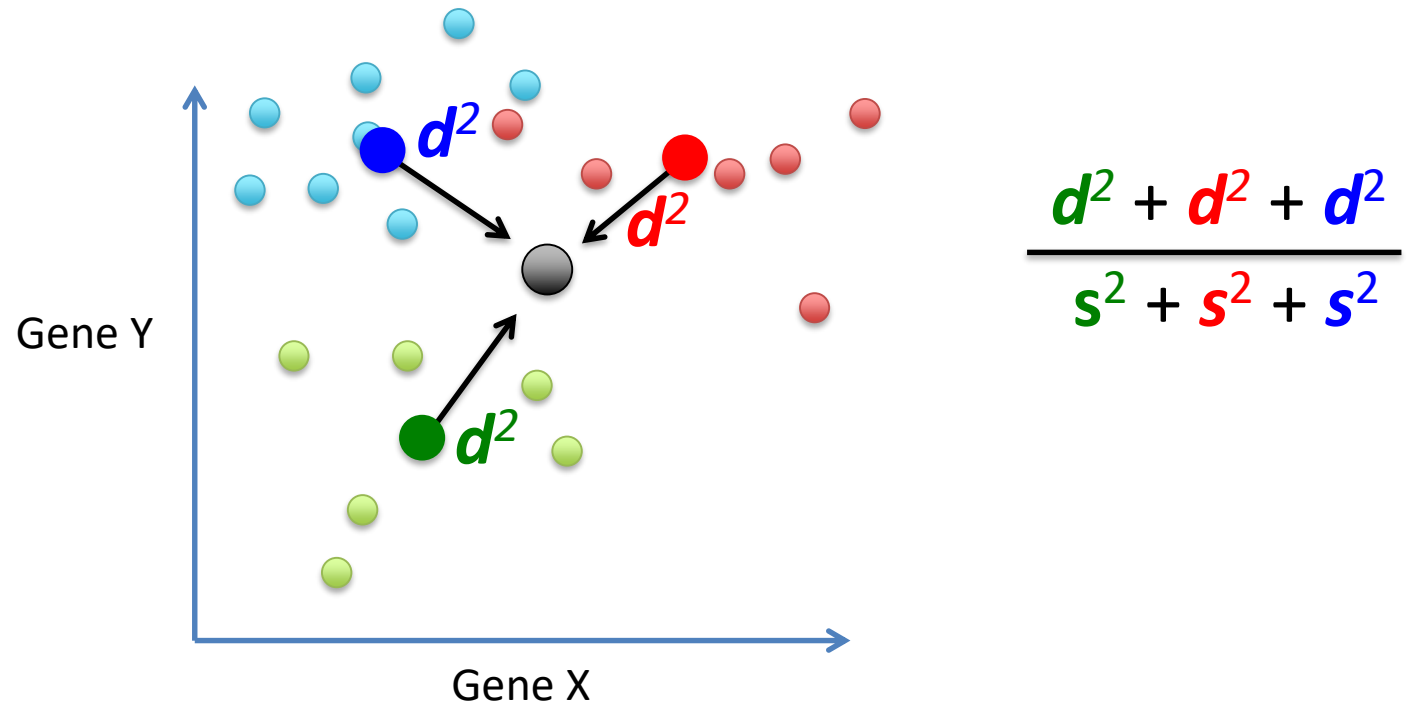
Then measure the distances between a point that is central in each category and the main central point.



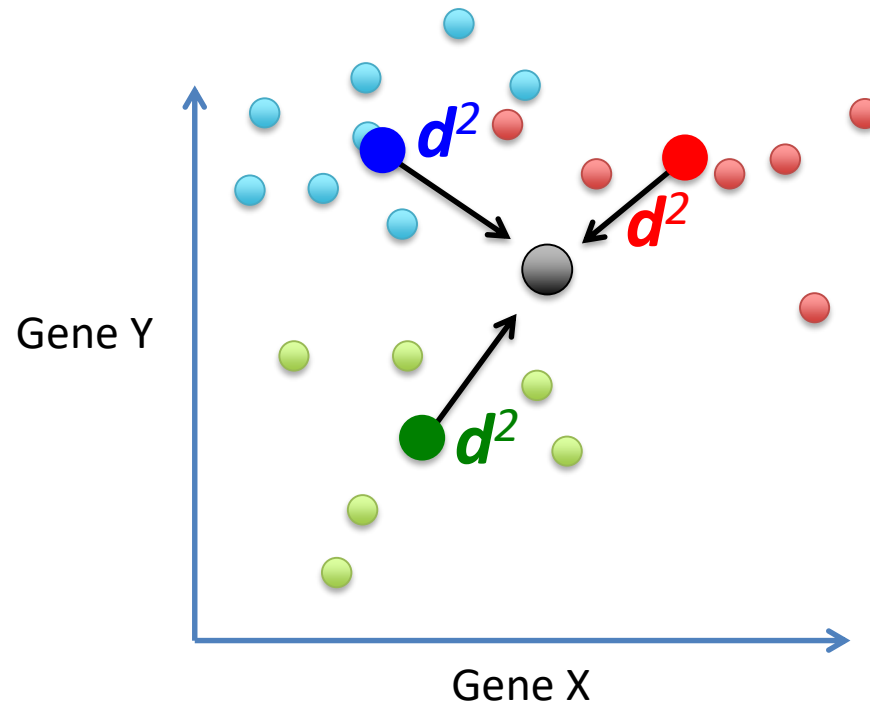
# LDA for 3 categories



# LDA for 3 categories



# LDA for 3 categories

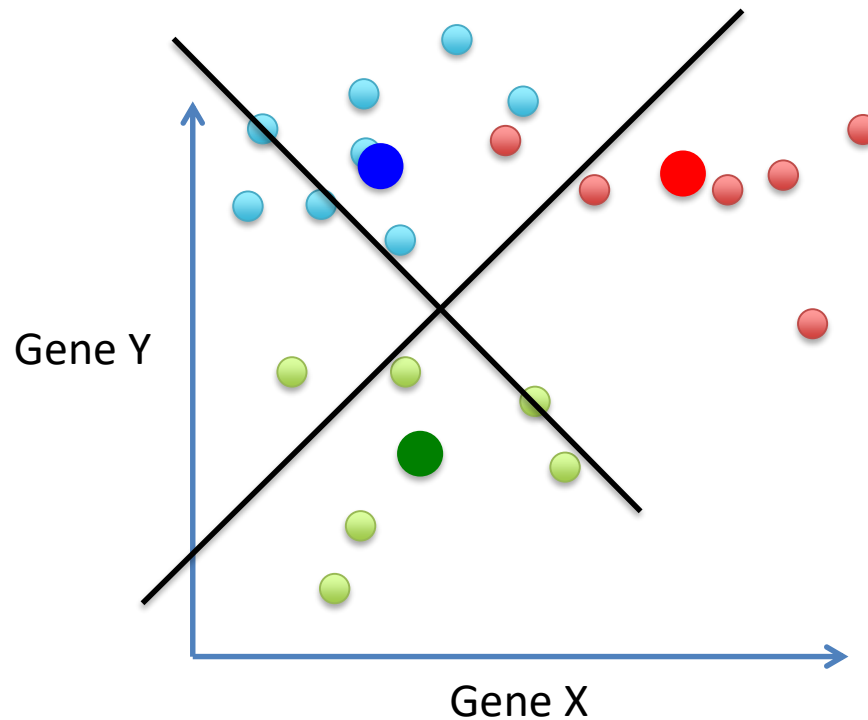


$$\frac{d^2 + d^2 + d^2}{s^2 + s^2 + s^2}$$

This is the same equation as before, but now there are terms for the **blue** category.

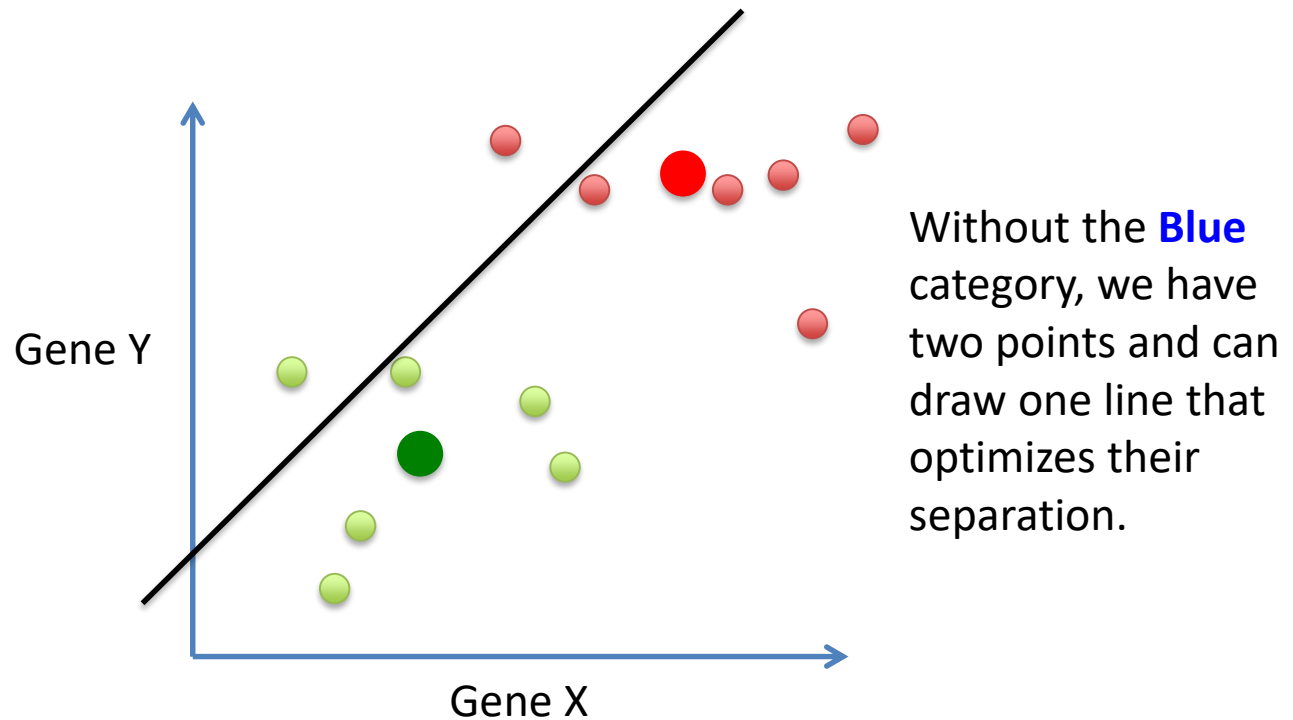
# LDA for 3 categories

**The second difference** is LDA creates 2 axes to separate the data. This is because the 3 central points for each category define a plane. (Remember from high school: 2 points define a line, 3 points define a plane...)



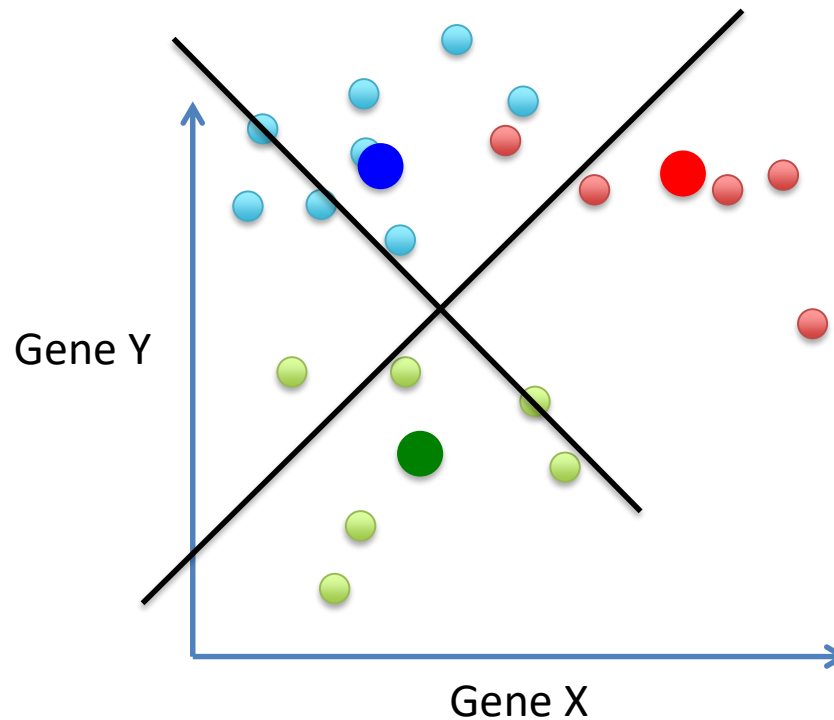
# LDA for 3 categories

**The second difference** is LDA creates 2 axes to separate the data. This is because the 3 central points for each category define a plane. (Remember from high school: 2 points define a line, 3 points define a plane...)



# LDA for 3 categories

**The second difference** is LDA creates 2 axes to separate the data. This is because the 3 central points for each category define a plane. (Remember from high school: 2 points define a line, 3 points define a plane...)

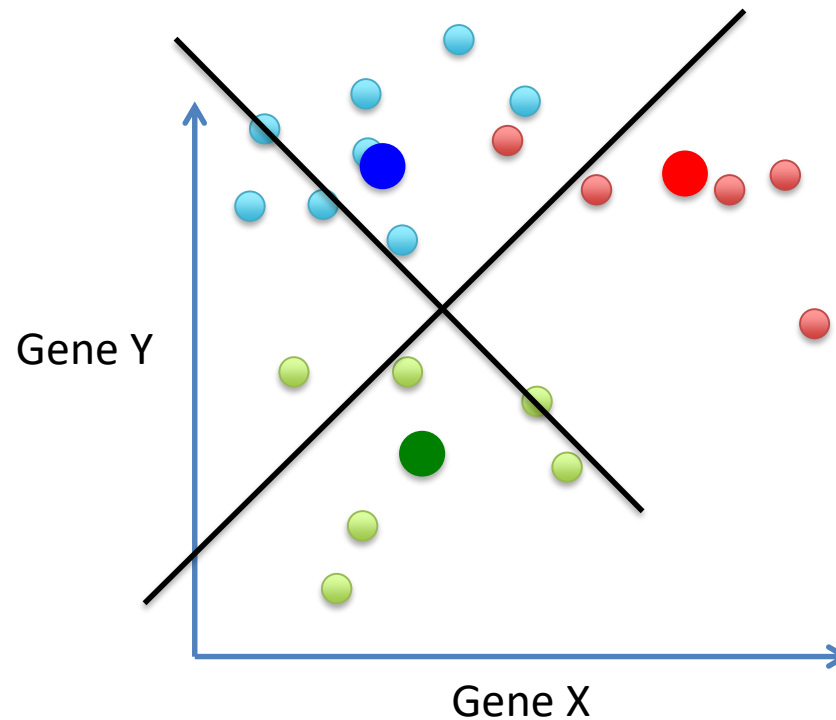


With three points, we can draw two lines to optimize separation.



# LDA for 3 categories

When we only use 2 genes, this is no big deal. The data started out on a X/Y plot and plotting them on a new X/Y plot doesn't change much.



# LDA for 3 categories

When we only use 2 genes, this is no big deal. The data started out on a X/Y plot and plotting them on a new X/Y plot doesn't change much.

But what if we used data from 10,000 genes?

# LDA for 3 categories

When we only use 2 genes, this is no big deal. The data started out on a X/Y plot and plotting them on a new X/Y plot doesn't change much.

But what if we used data from 10,000 genes?

That would mean we'd need 10,000 dimensions to draw the data.

# LDA for 3 categories

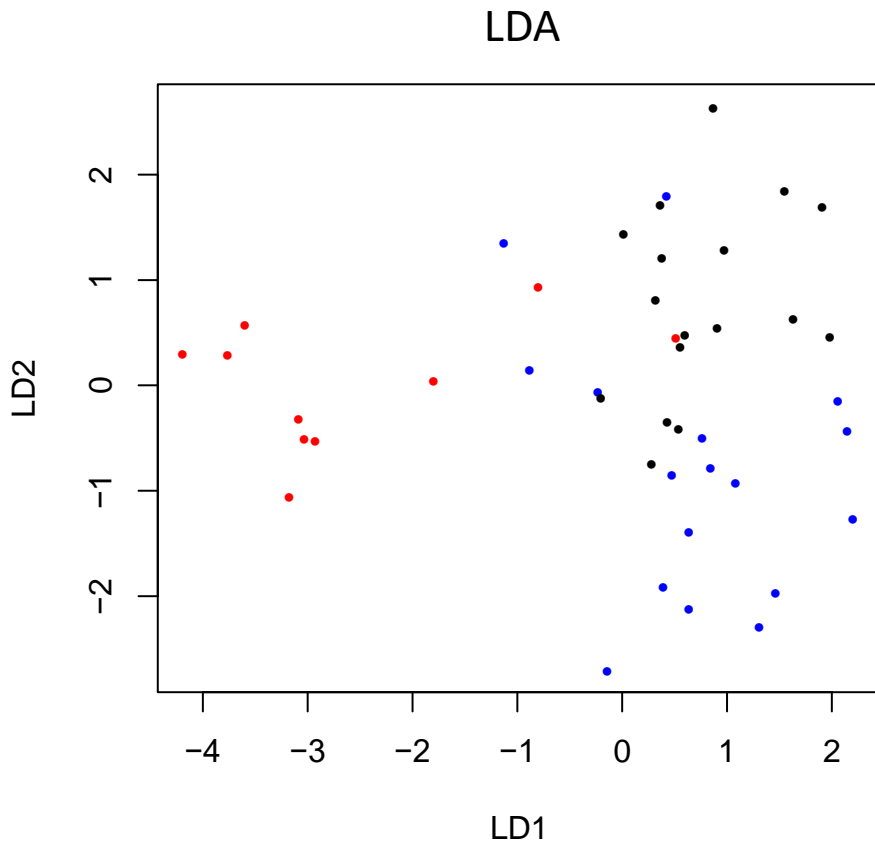
When we only use 2 genes, this is no big deal. The data started out on a X/Y plot and plotting them on a new X/Y plot doesn't change much.

But what if we used data from 10,000 genes?

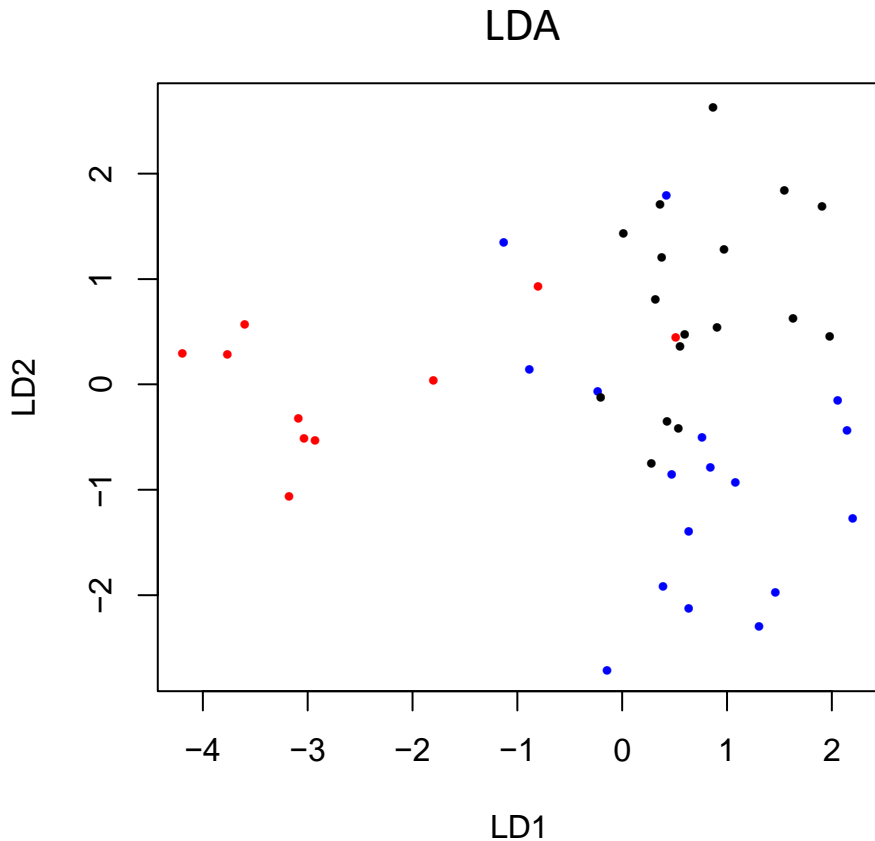
That would mean we'd need 10,000 dimensions to draw the data.

Suddenly, being able to create 2 axes that maximize the separation of three categories is super cool!!!!

# LDA with 3 categories and 10,000 genes.



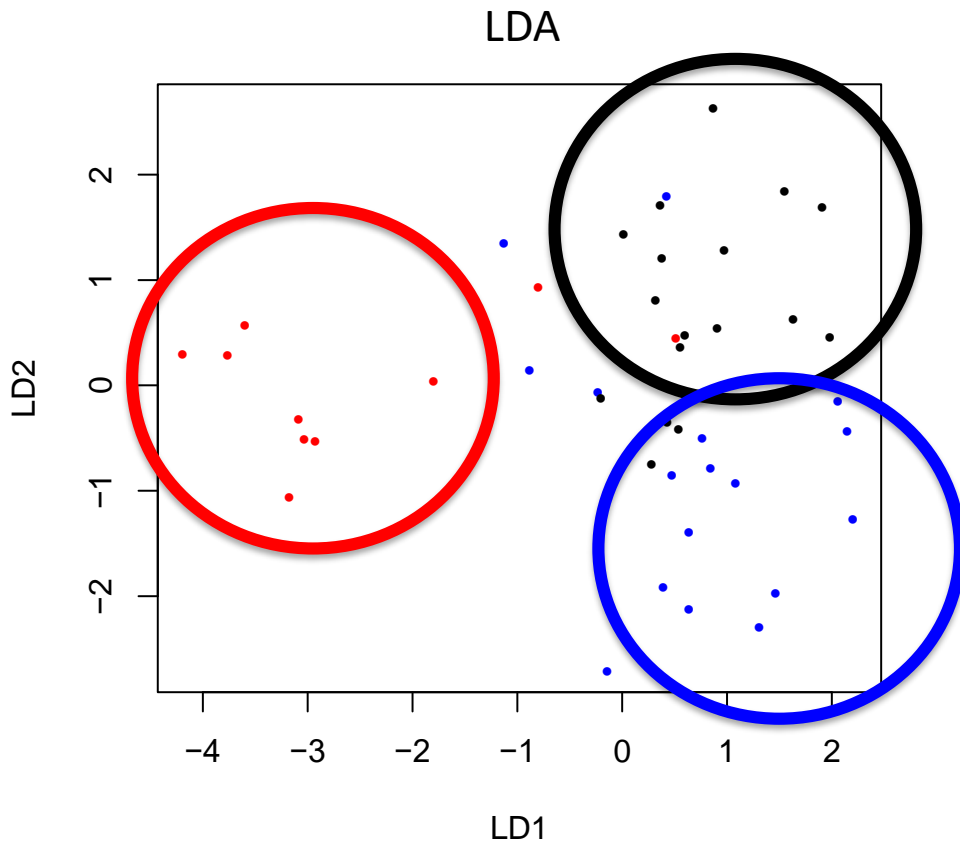
# LDA with 3 categories and 10,000 genes.



Plotting the raw data would require 10,000 axes.

We used LDA to reduce that number to 2.

# LDA with 3 categories and 10,000 genes.

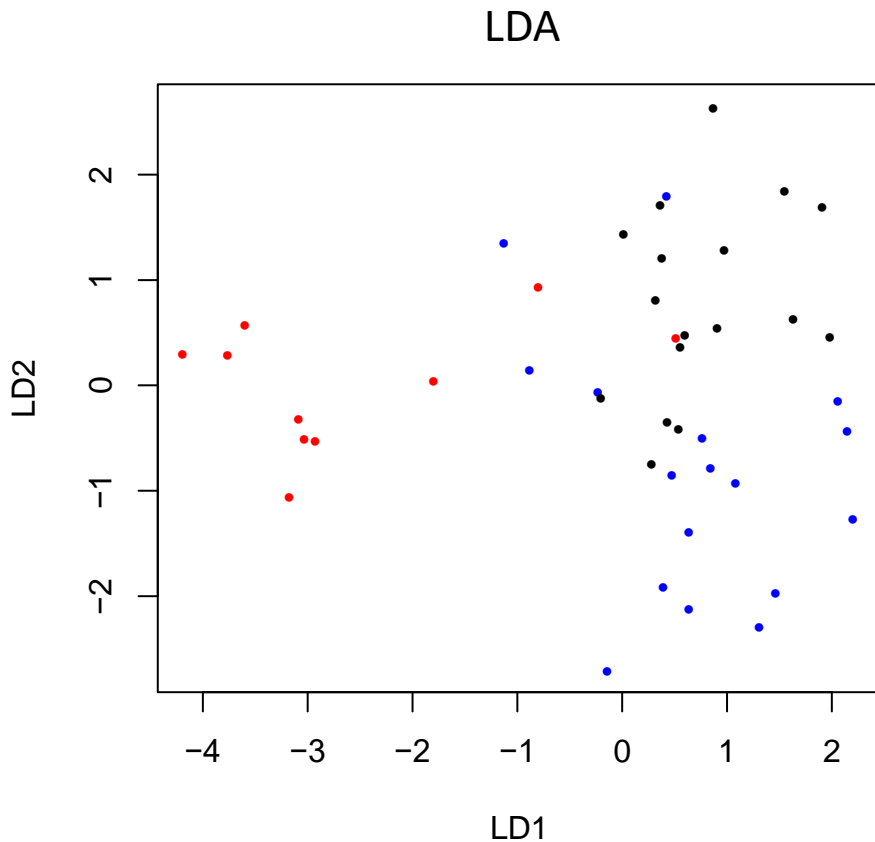


Although the separation isn't perfect, it is still easy to see three separate categories.

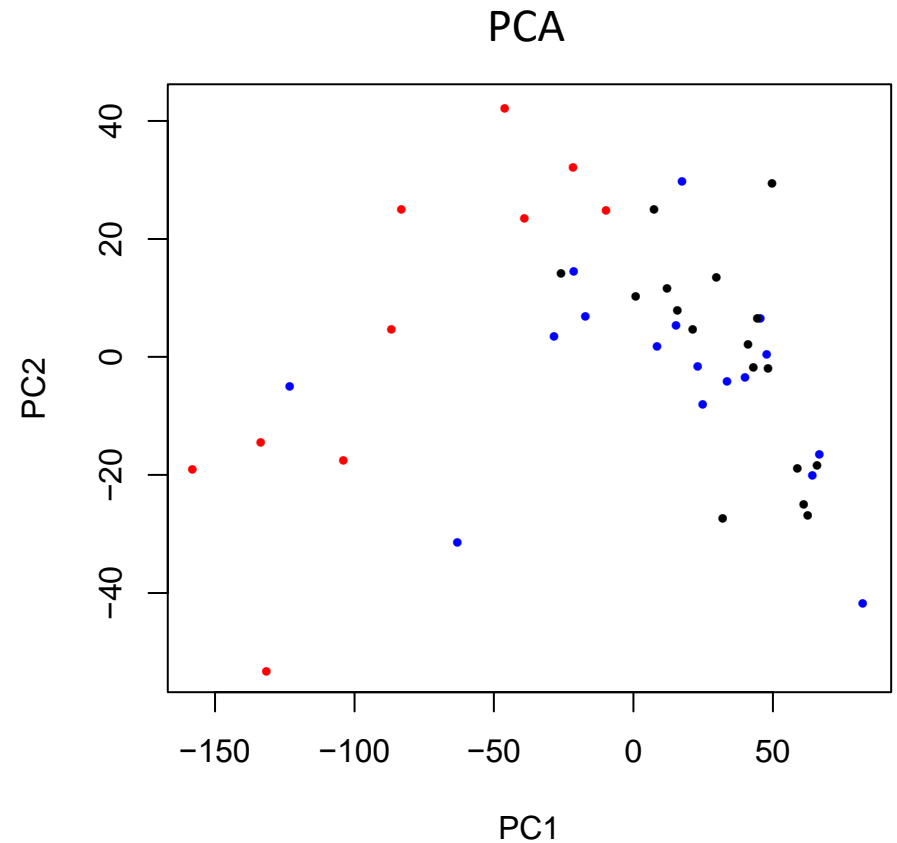
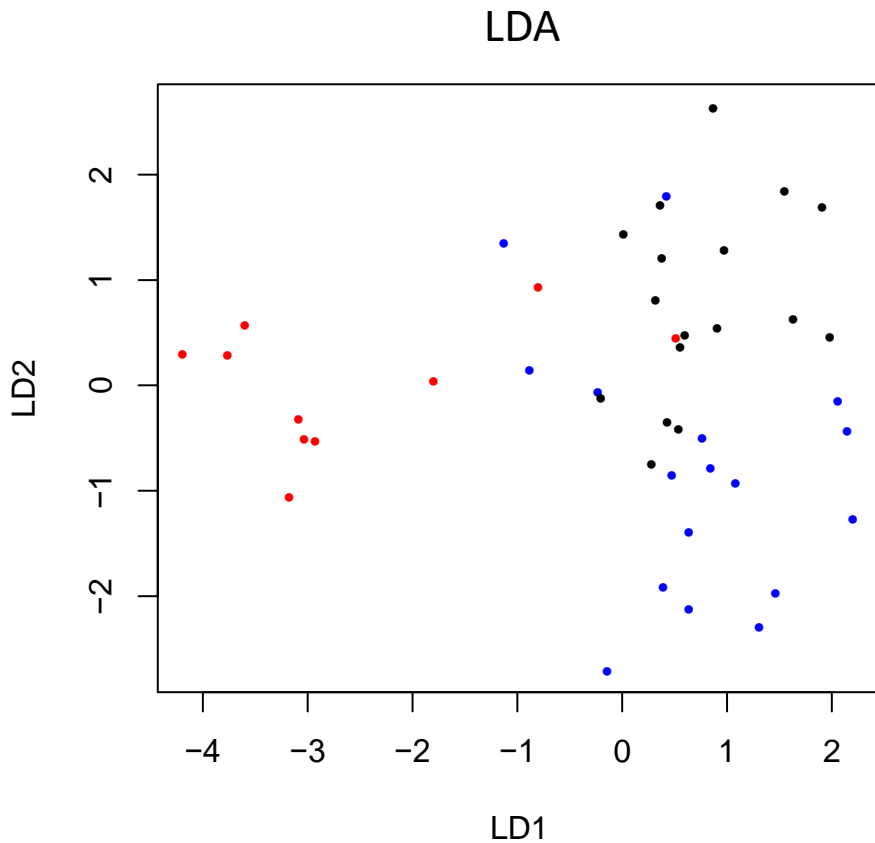
# Comparing LDA to PCA with 10,000 genes.



# Comparing LDA to PCA with 10,000 genes.

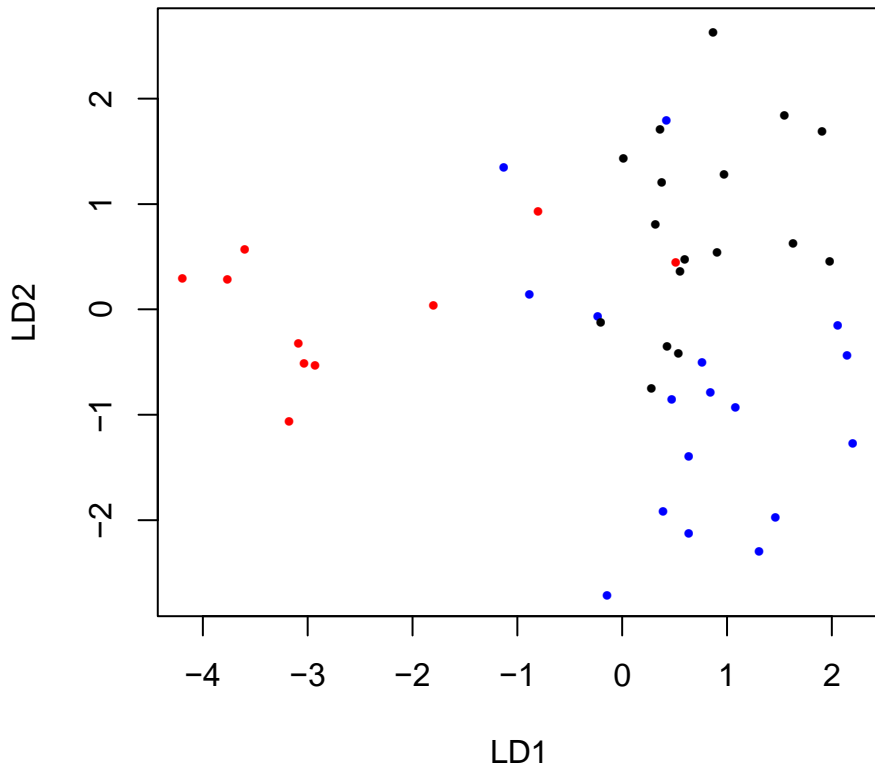


# Comparing LDA to PCA with 10,000 genes.



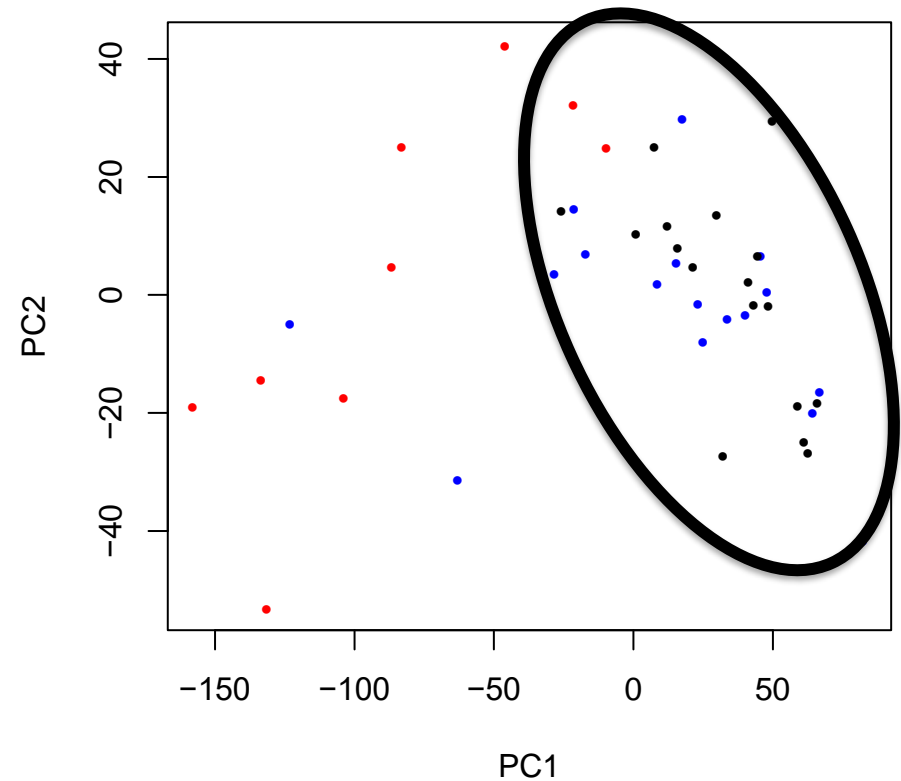
# Comparing LDA to PCA with 10,000 genes.

LDA



PCA

Separation isn't  
nearly as good.



# Similarities between PCA and LDA

# Similarities between PCA and LDA

- Both rank the new axes in order of importance.

# Similarities between PCA and LDA

- Both rank the new axes in order of importance.
  - PC1 (the first new axis that PCA creates) accounts for the most variation in the data.

# Similarities between PCA and LDA

- Both rank the new axes in order of importance.
  - PC1 (the first new axis that PCA creates) accounts for the most variation in the data.
    - PC2 (the second new axis) does the second best job...

# Similarities between PCA and LDA

- Both rank the new axes in order of importance.
  - PC1 (the first new axis that PCA creates) accounts for the most variation in the data.
    - PC2 (the second new axis) does the second best job...
  - LD1 (the first new axis that LDA creates) accounts for the most variation between the categories.



# Similarities between PCA and LDA

- Both rank the new axes in order of importance.
  - PC1 (the first new axis that PCA creates) accounts for the most variation in the data.
    - PC2 (the second new axis) does the second best job...
  - LD1 (the first new axis that LDA creates) accounts for the most variation between the categories.
    - LD2 (the second new axis) does the second best job...

# Similarities between PCA and LDA

- Both rank the new axes in order of importance.
  - PC1 (the first new axis that PCA creates) accounts for the most variation in the data.
    - PC2 (the second new axis) does the second best job...
  - LD1 (the first new axis that LDA creates) accounts for the most variation between the categories.
    - LD2 (the second new axis) does the second best job...
- Both can let you dig in and see which genes are driving the new axes.

# In summary

- LDA is like PCA – both try to reduce dimensions

# In summary

- LDA is like PCA – both try to reduce dimensions
  - PCA looks at the genes with the most variation.

# In summary

- LDA is like PCA – both try to reduce dimensions
  - PCA looks at the genes with the most variation.
  - LDA tries to maximize the separation of known categories.

**THE END!!!**