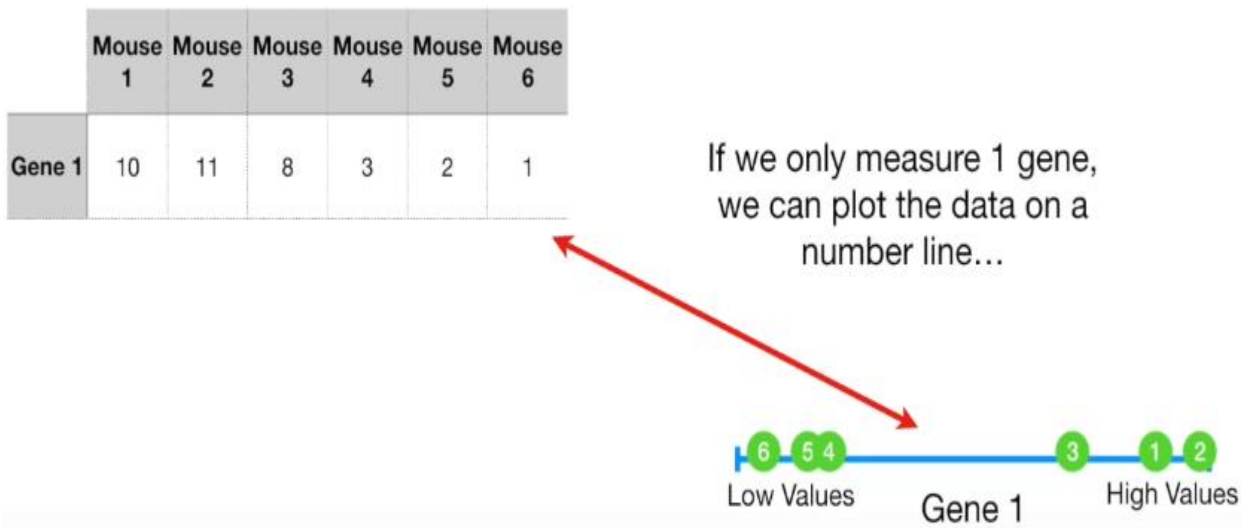
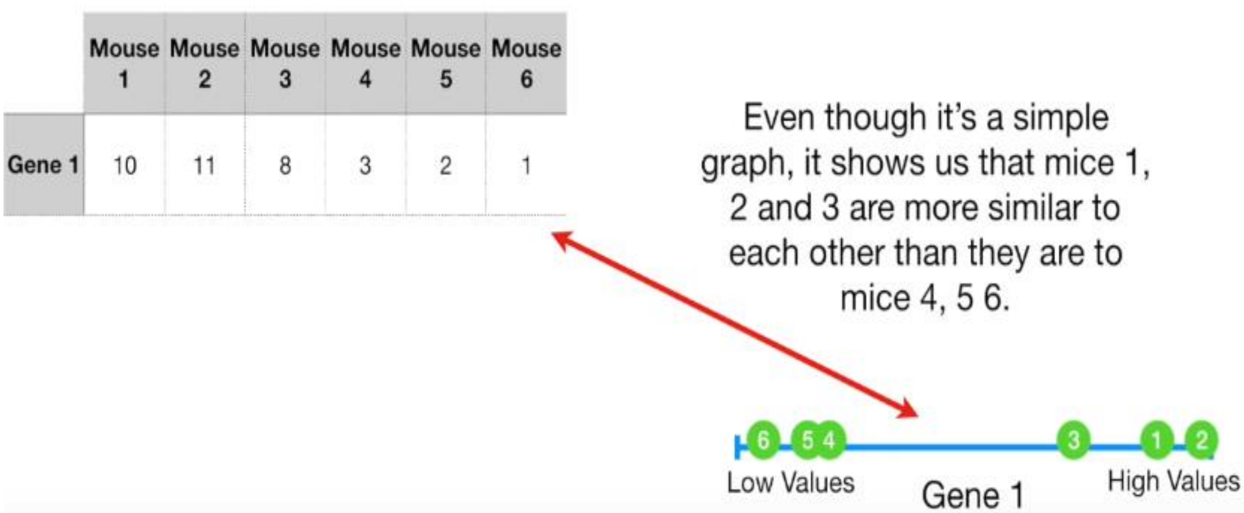
PCA Geometry

1D – Plot: Clear separation and depiction of data

One sample of data -



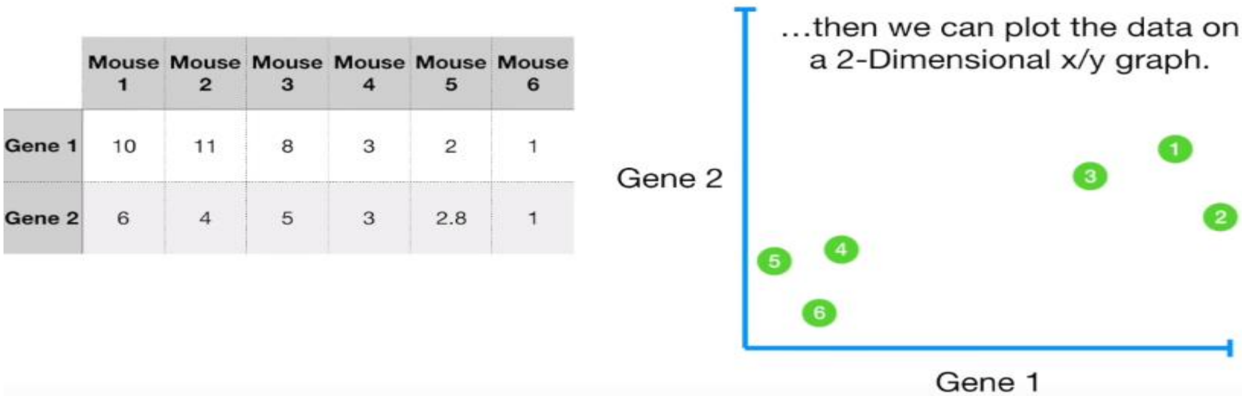
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1

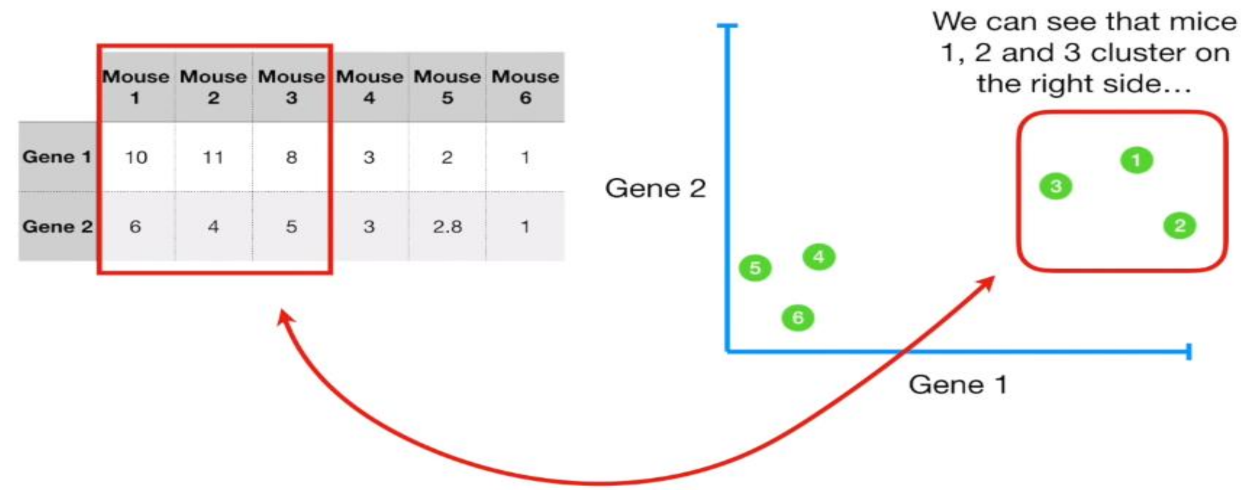


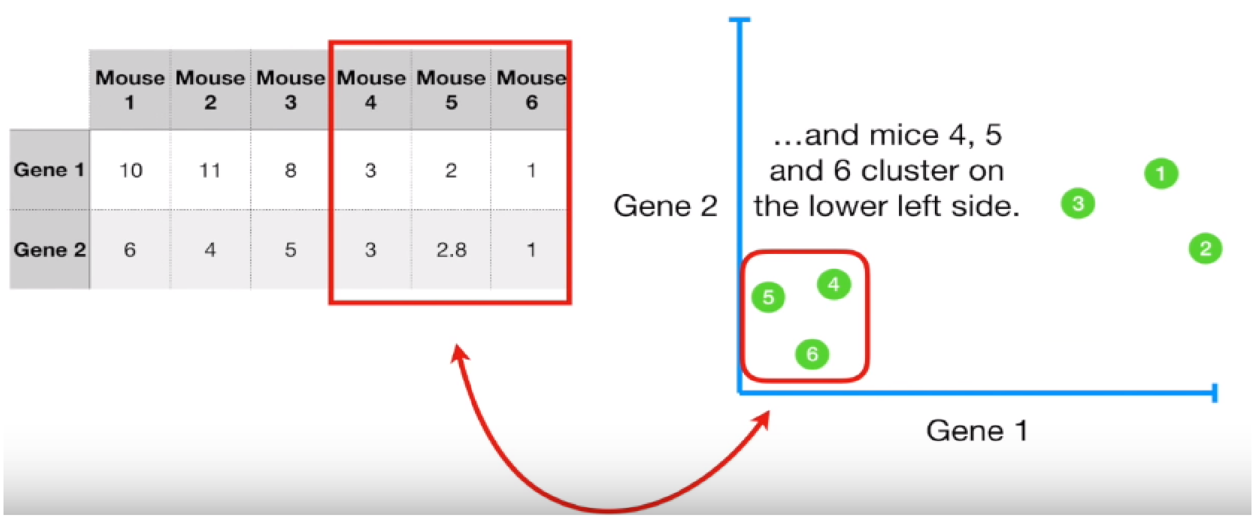
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1

2D – Plot: Clear separation and depiction of data

Two sample of data -

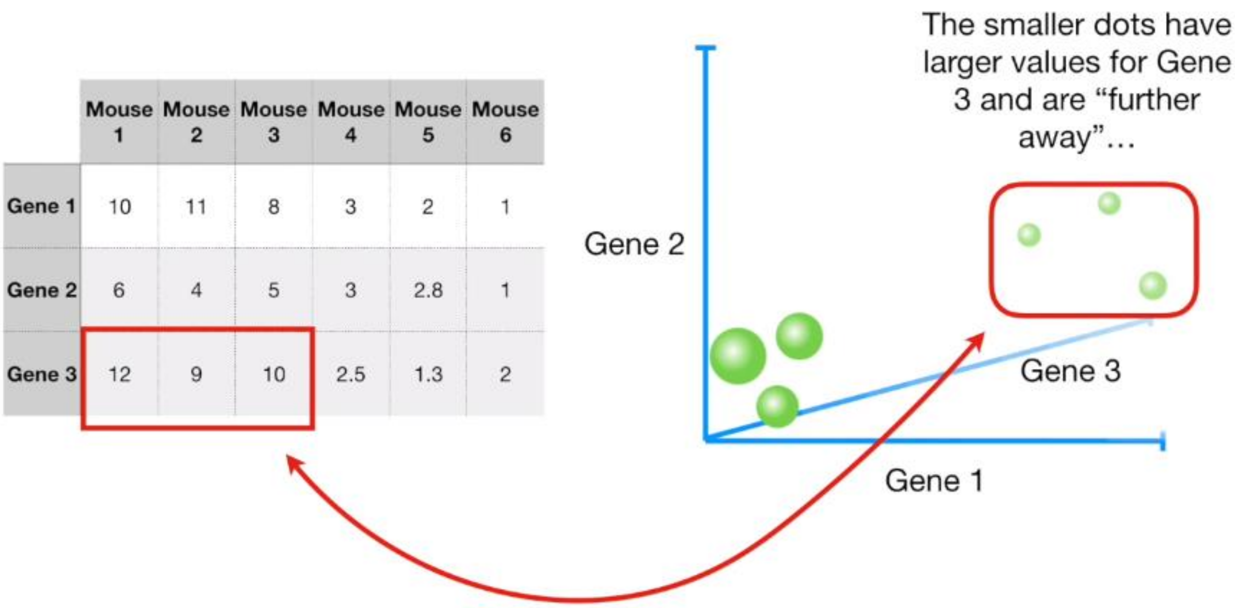




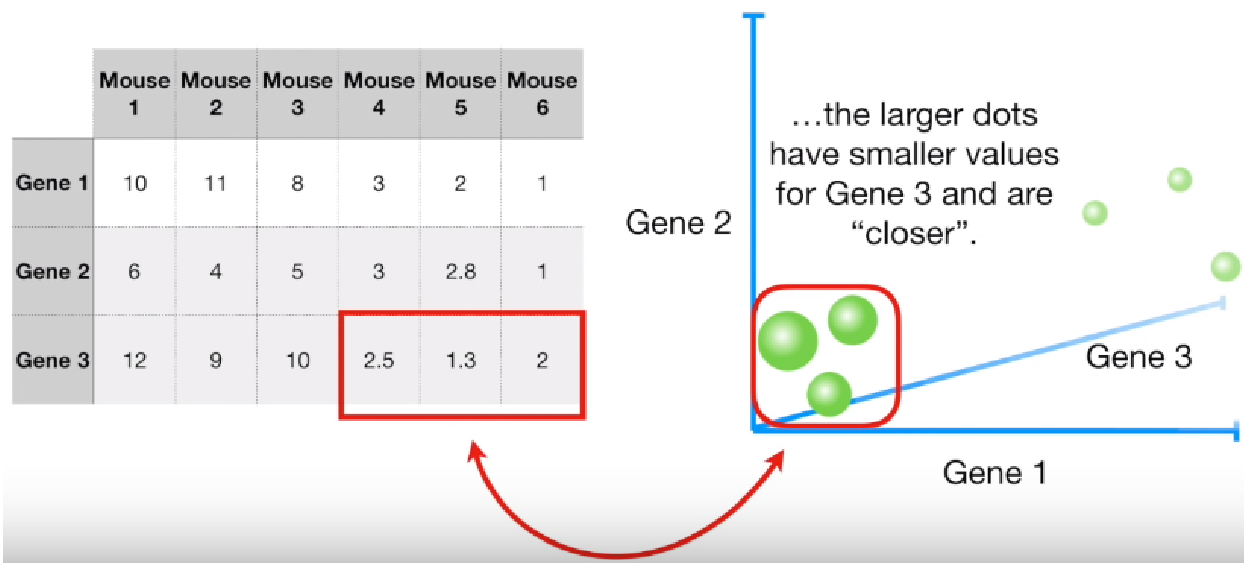


3D – Plot: Clear separation and depiction of data

Three sample of data -



	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2



	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2

4D – Plot: Not Possible

Introducing four sample of data -

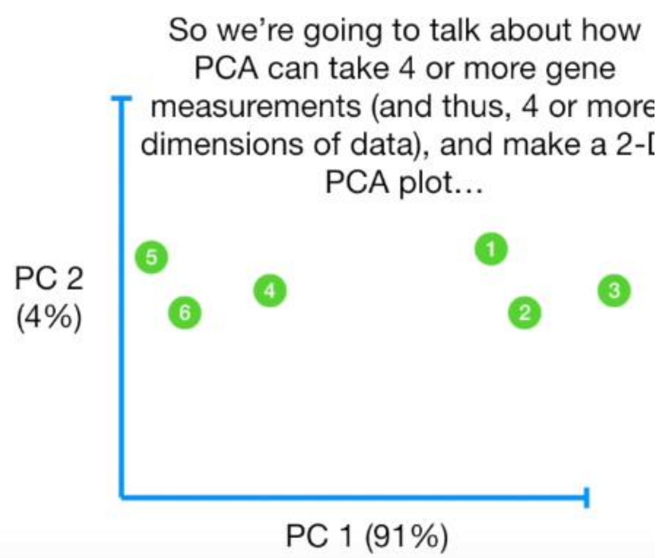
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2
Gene 4	5	7	6	2	4	7

If we measured 4 genes, however, we can no longer plot the data - 4 genes require 4 dimensions.

:(

Need for PCA for representing high dimensional data in lower space (dimension)

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1
Gene 3	12	9	10	2.5	1.3	2
Gene 4	5	7	6	2	4	7

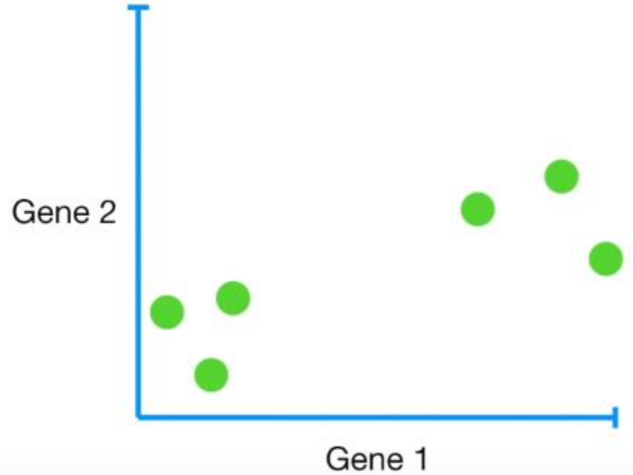


Example with two-dimensional data for illustrating PCA concept!!

PCA Functioning with 2D

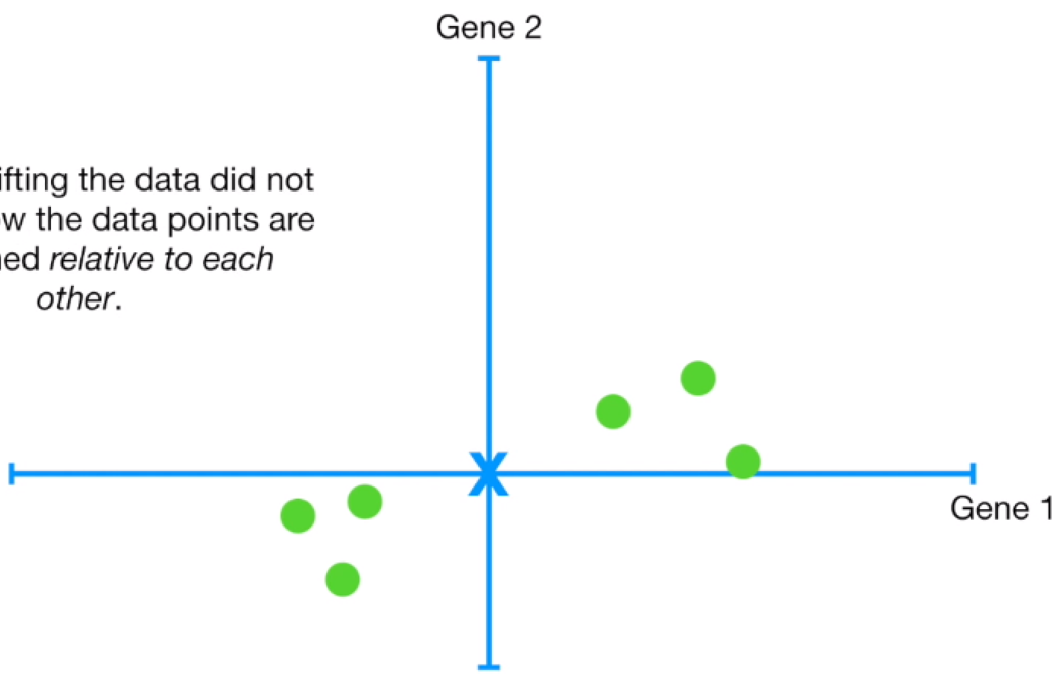
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	10	11	8	3	2	1
Gene 2	6	4	5	3	2.8	1

We'll start by plotting the data…

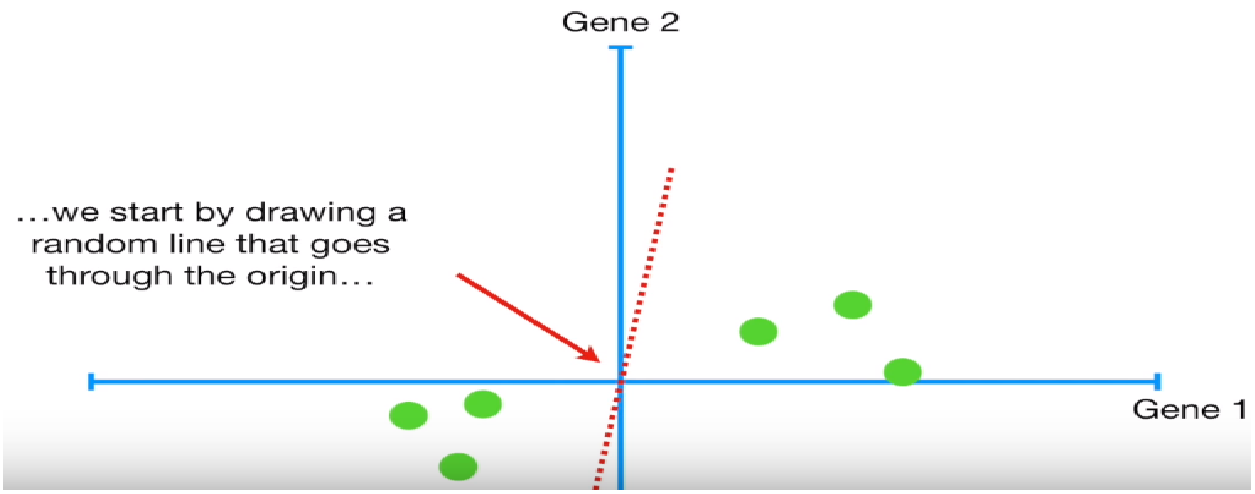


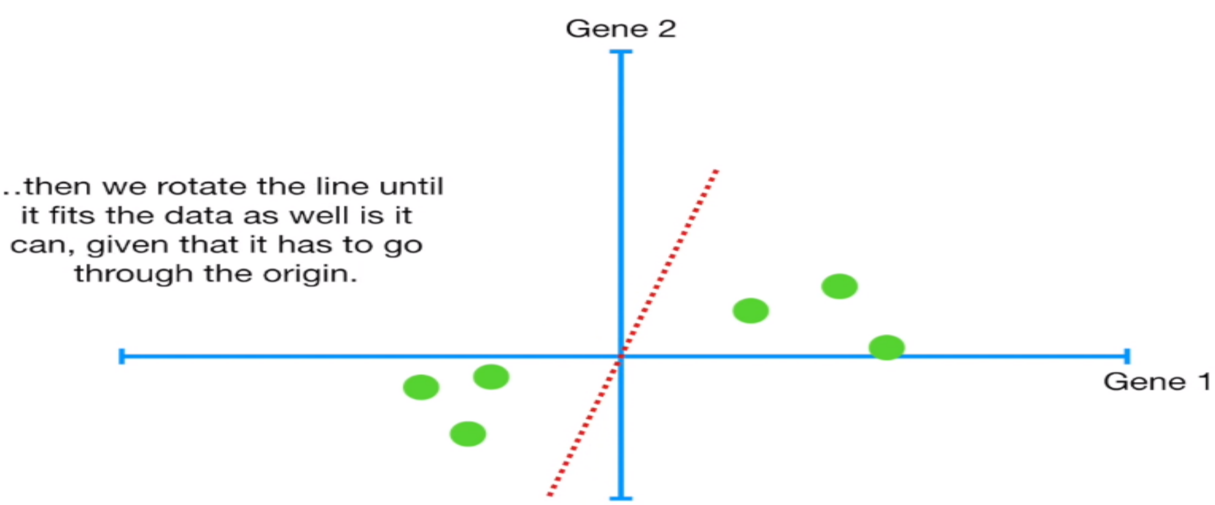
Column standardization: - Shifting to the origin

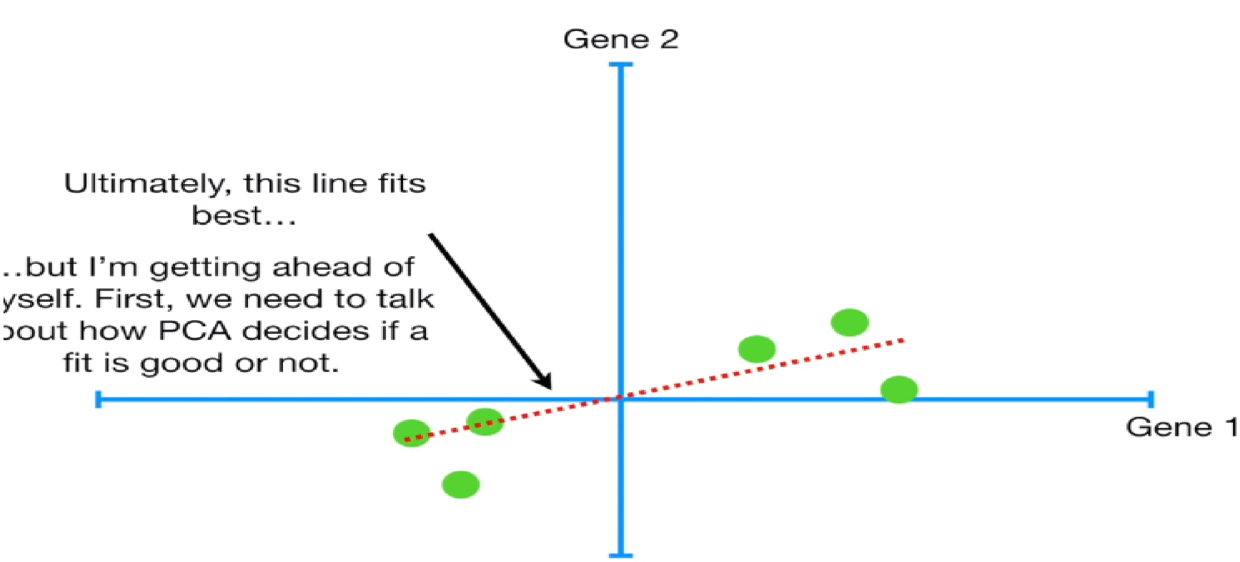
NOTE: Shifting the data did not change how the data points are positioned *relative to each other*.



Estimating a best fit line: -

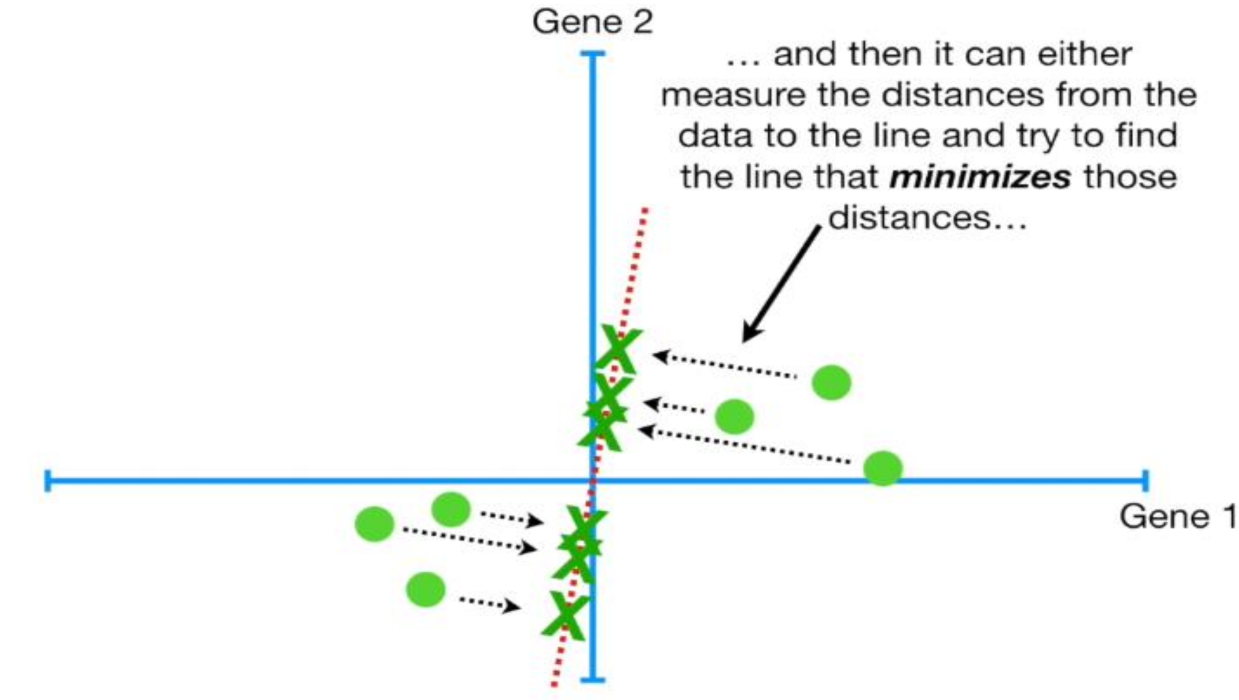




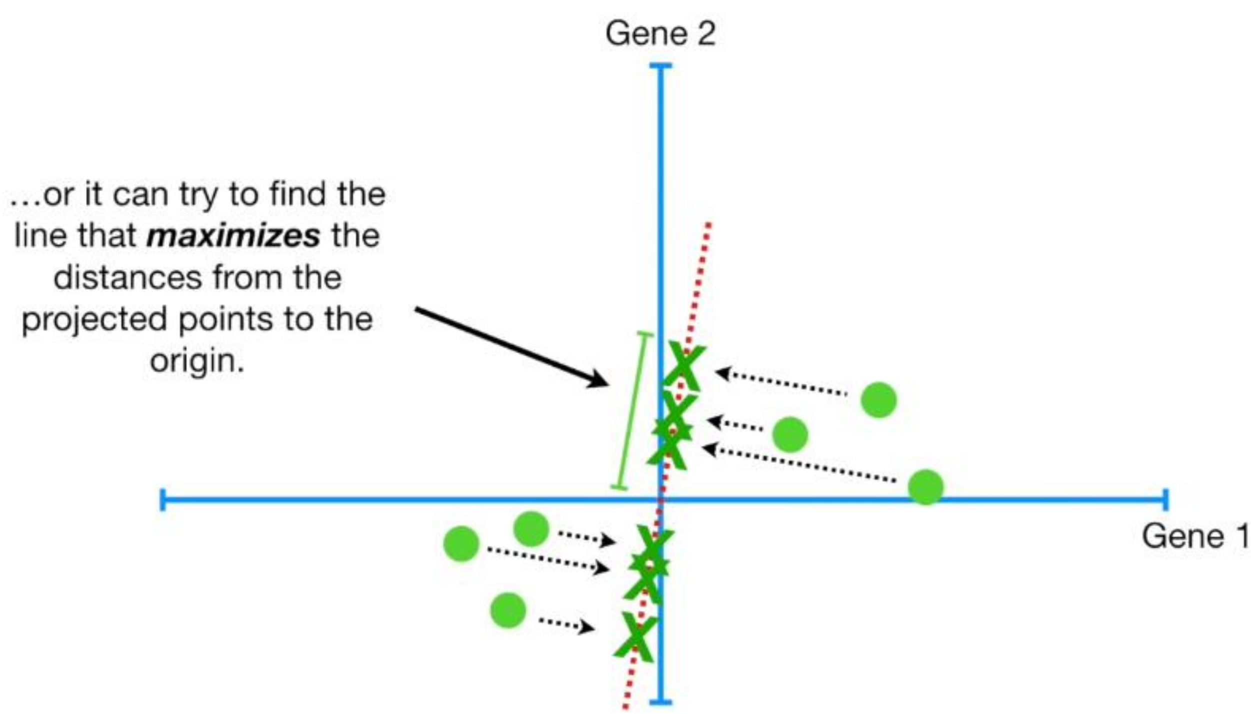


Method for finding the best fit line: - Projection of data onto the line

Optimization 1: Minimize sum of distances of the projected data from the line



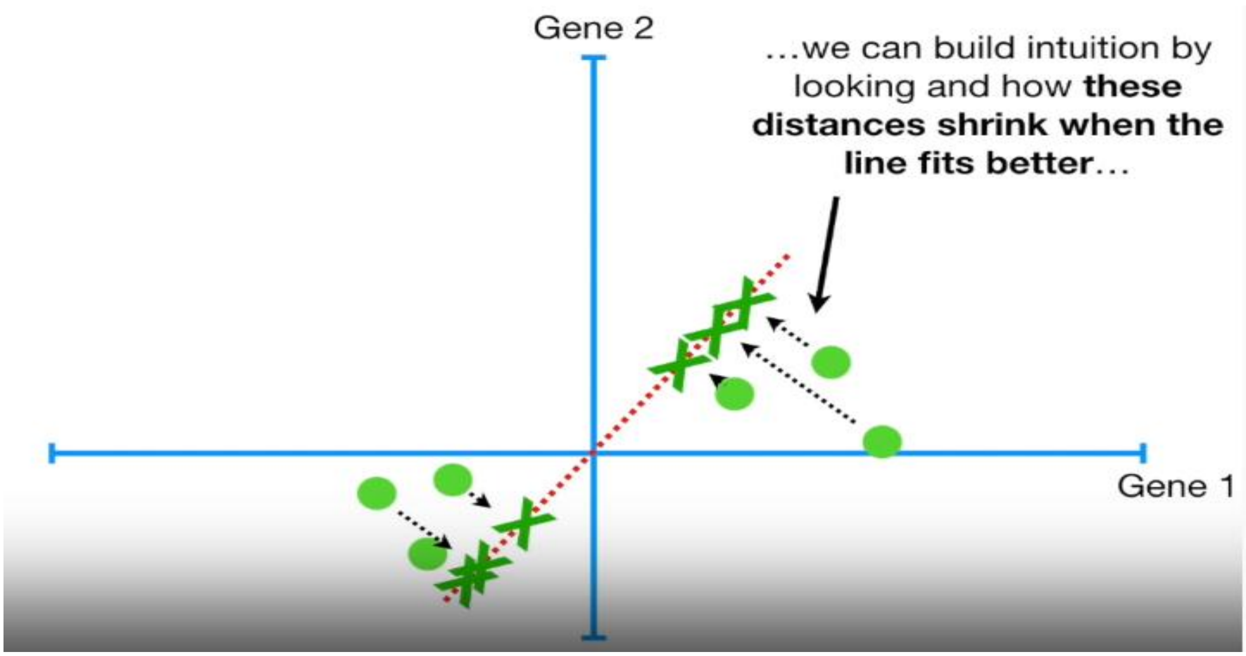
Optimization 2: Maximize the distance of projected data from the origin

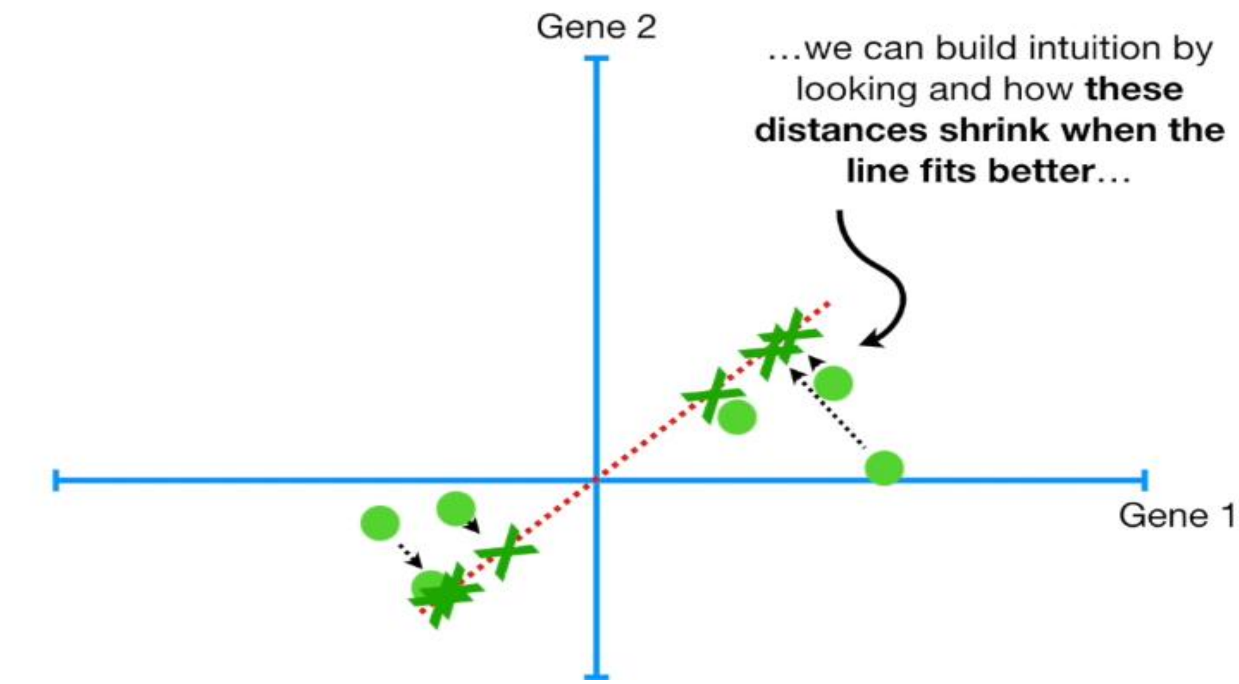


Both optimization 1 and 2 are equivalent, can be proved by below intuition

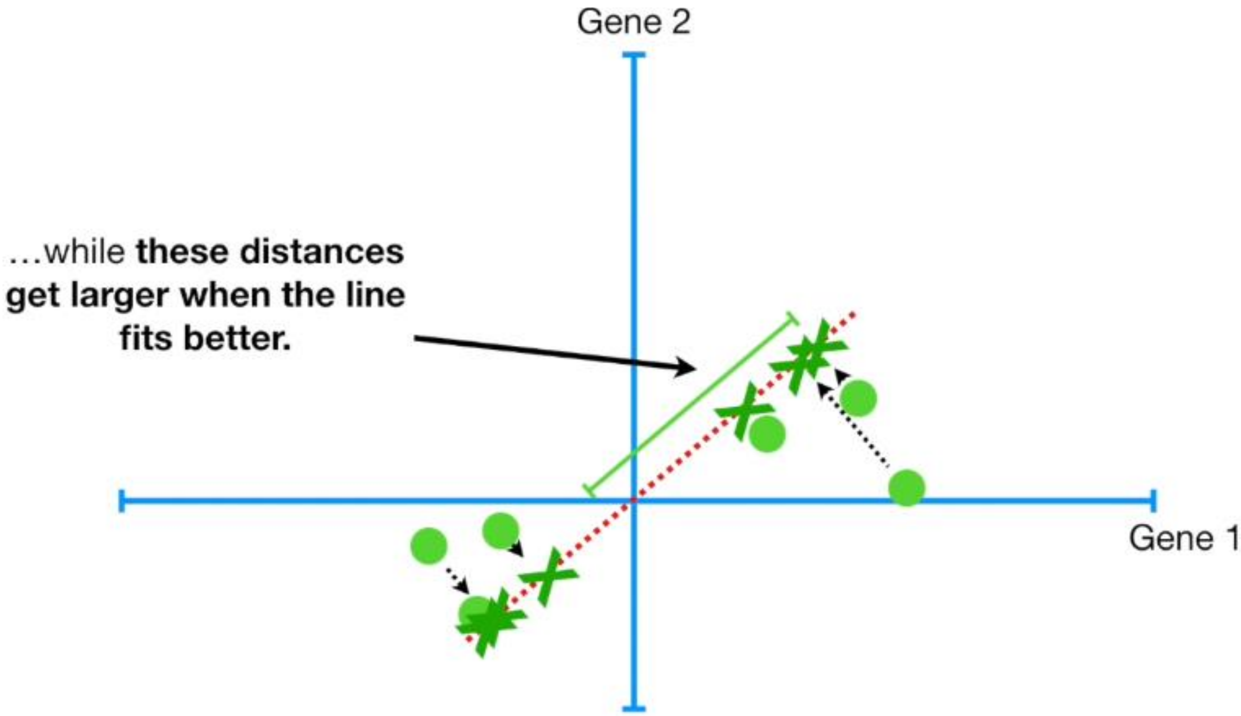
Visualize by rotation – observe the shrinking of the distances from the line

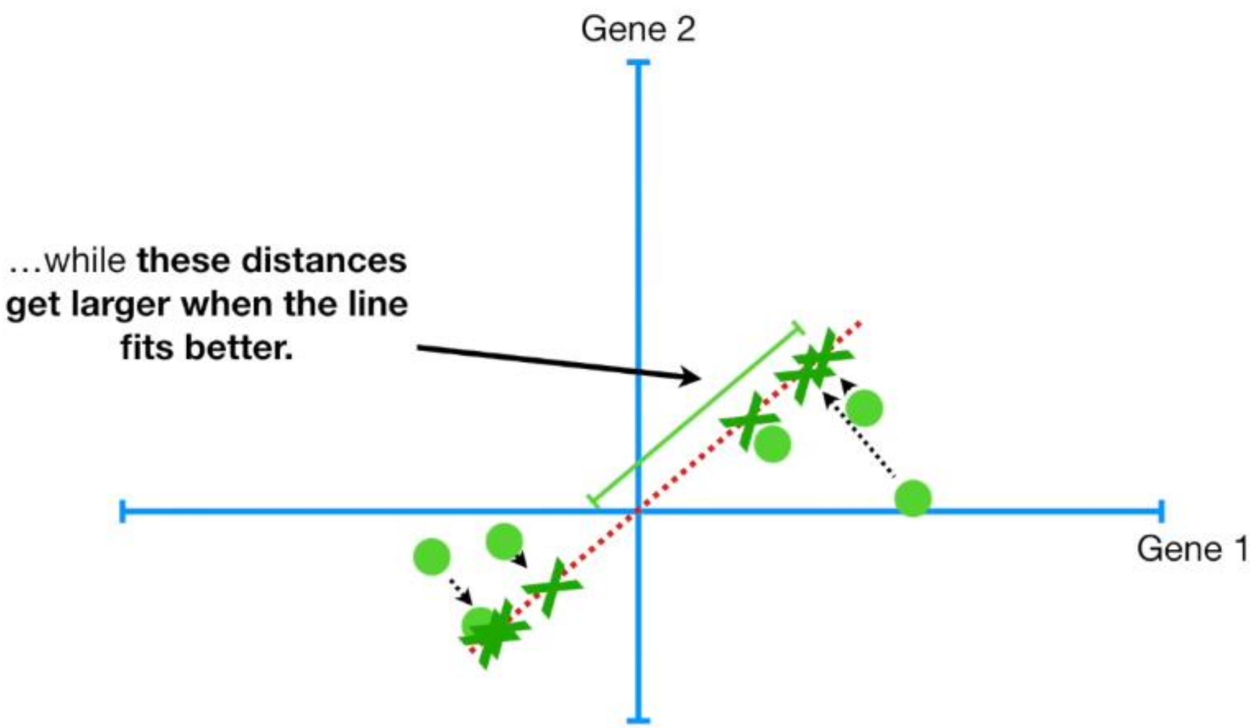
- **Optimization 1**



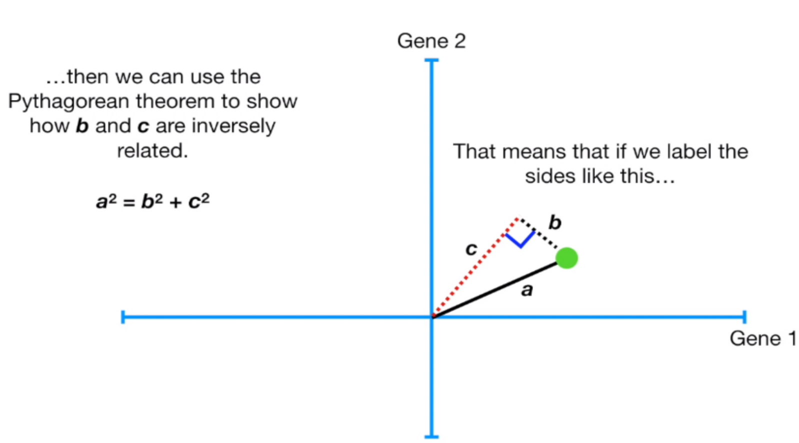


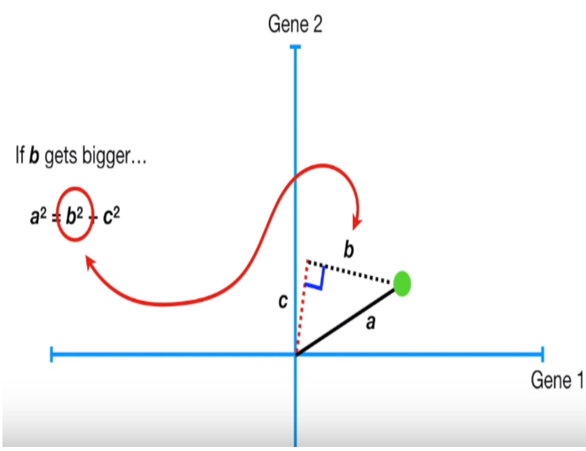
- **Optimization 2**

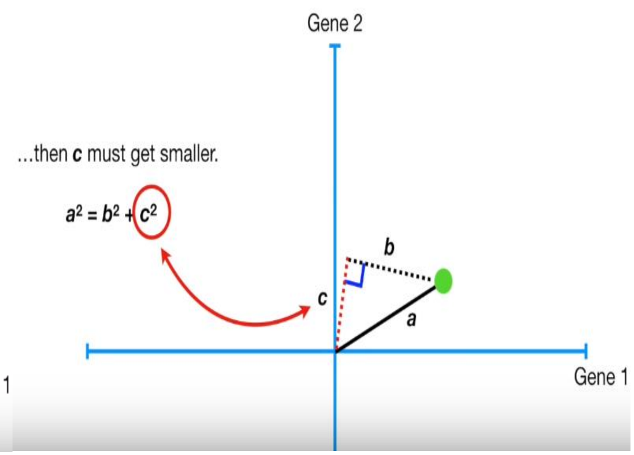


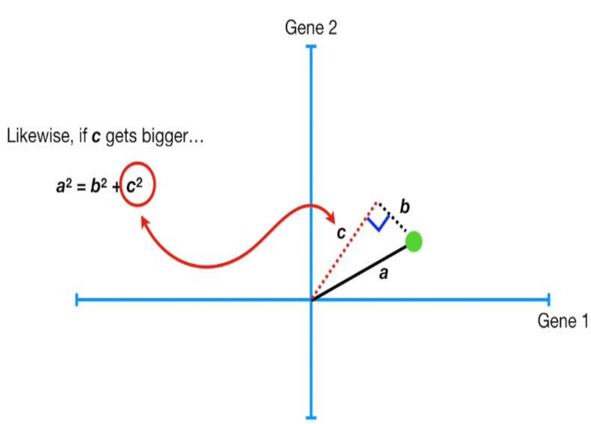


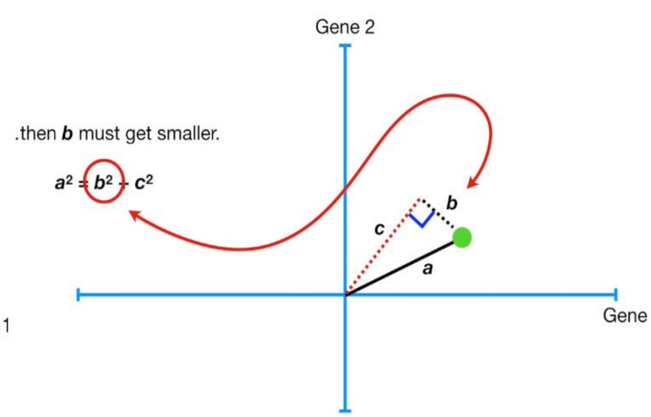
Geometric proof - optimization 1 and 2 are equivalent





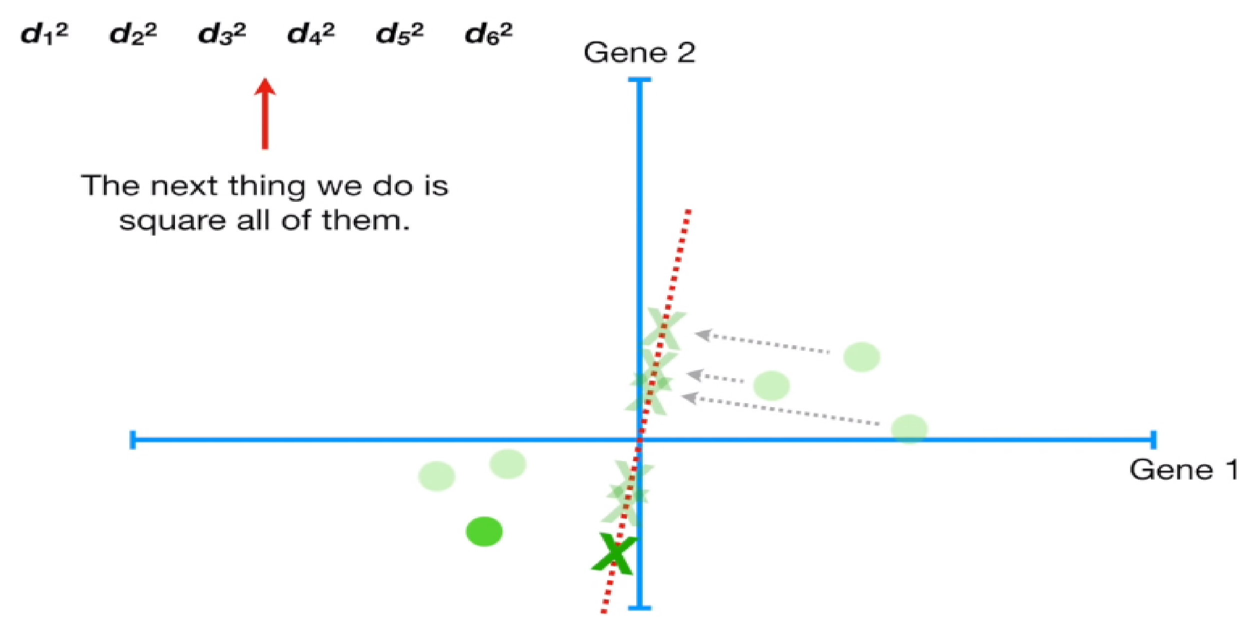


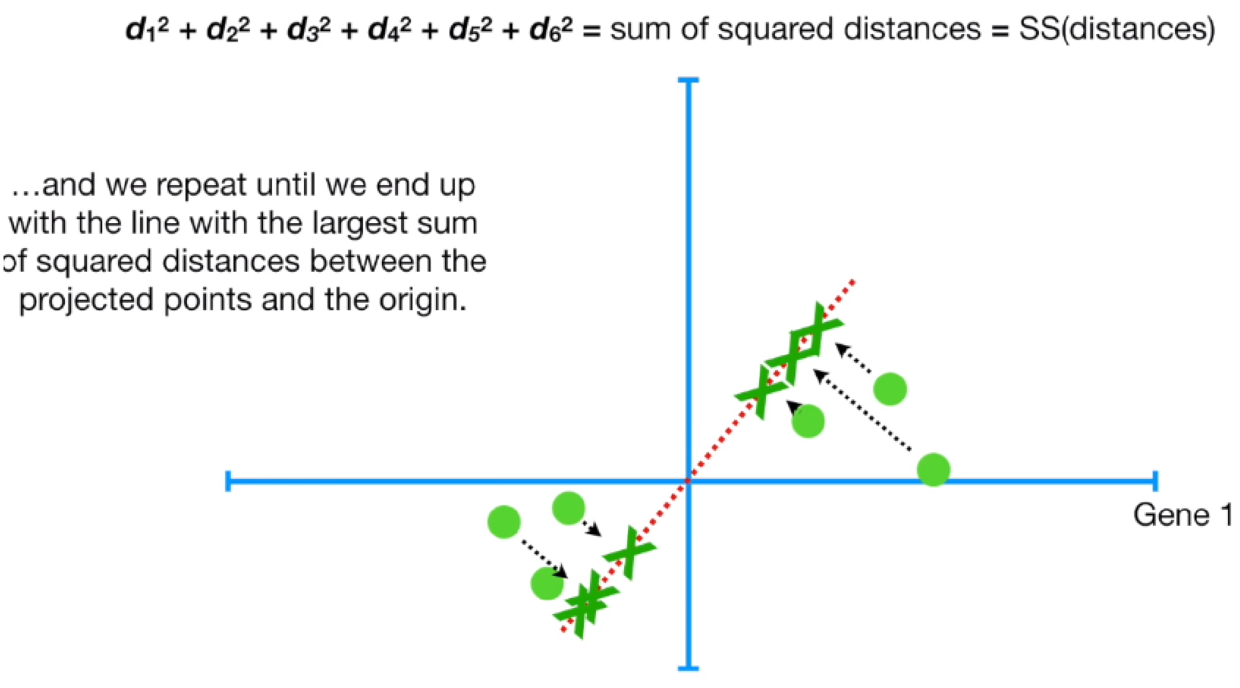


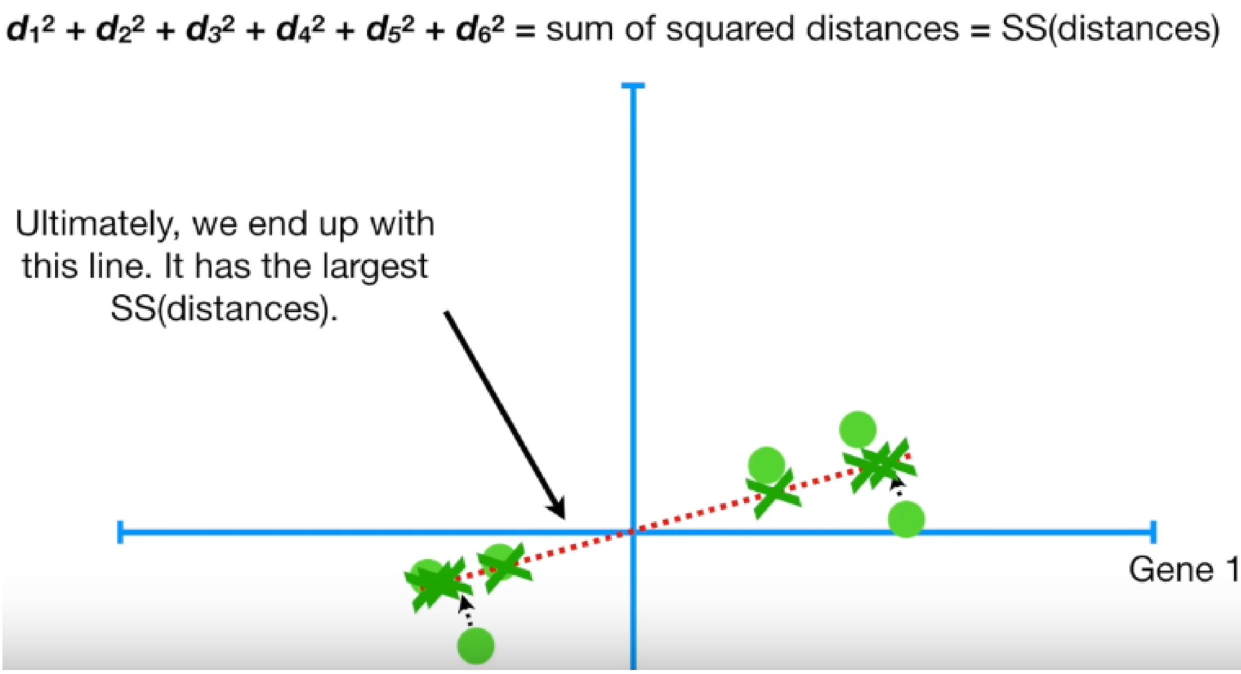


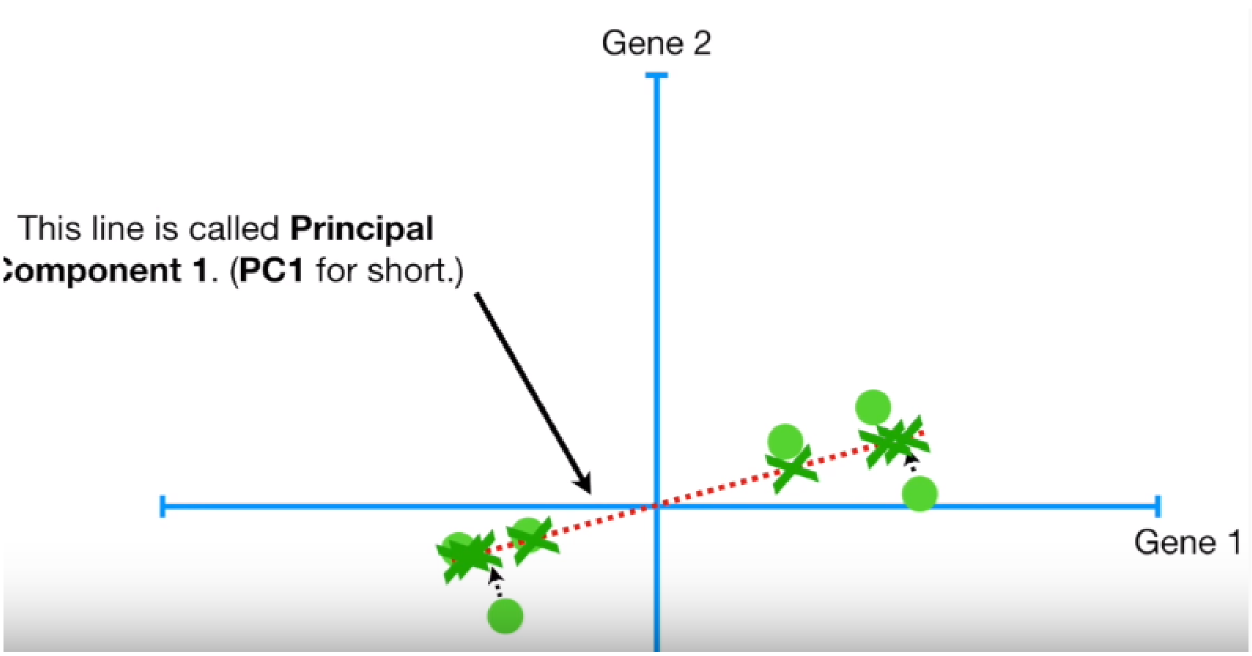
Optimization 1: Minimize sum of distances of the projected data from the line

Find: first principle component

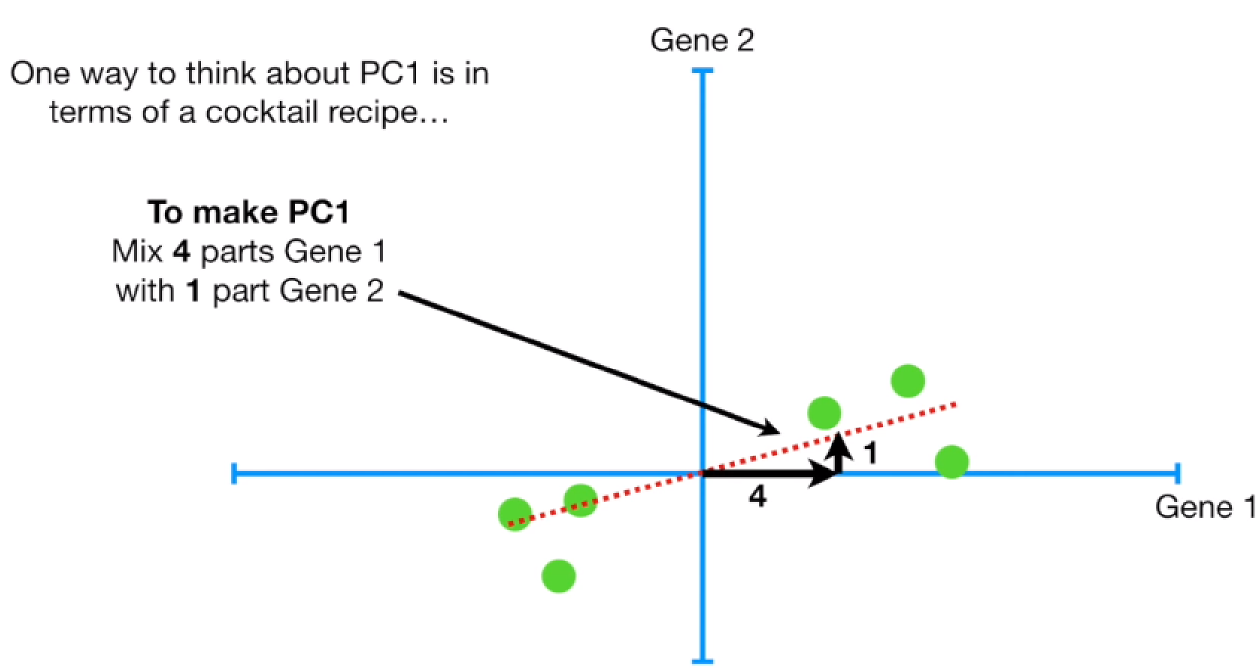


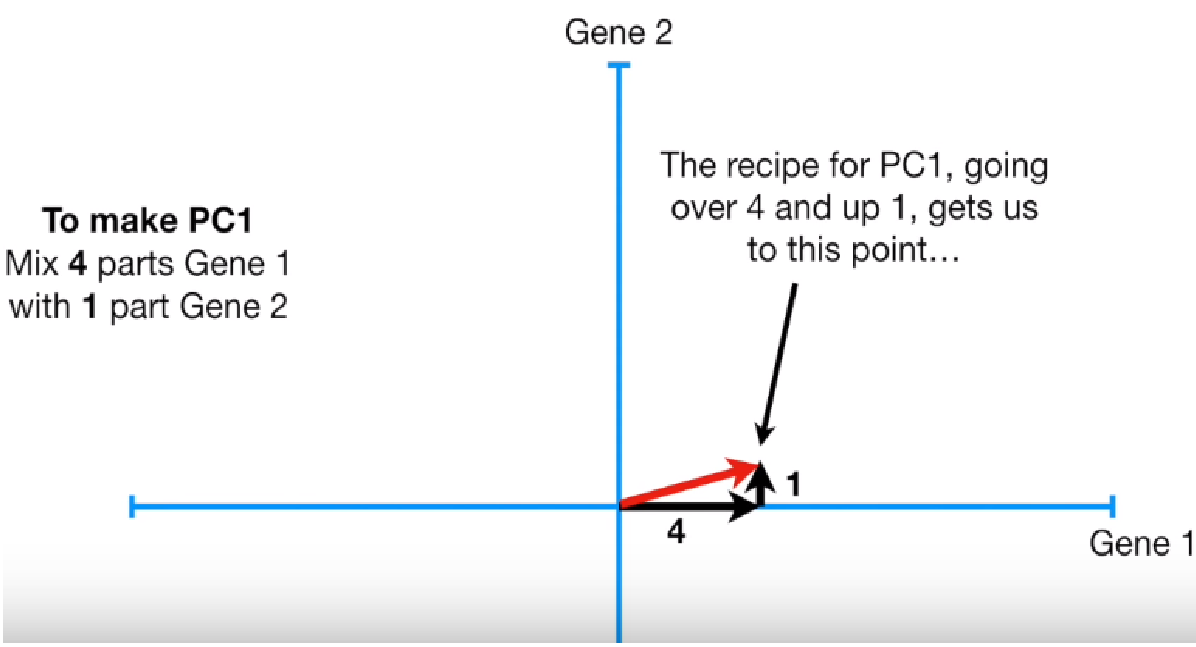



$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2$ = sum of squared distances = SS(distances)
Ultimately, we end up with this line. It has the largest SS(distances).
Gene 1

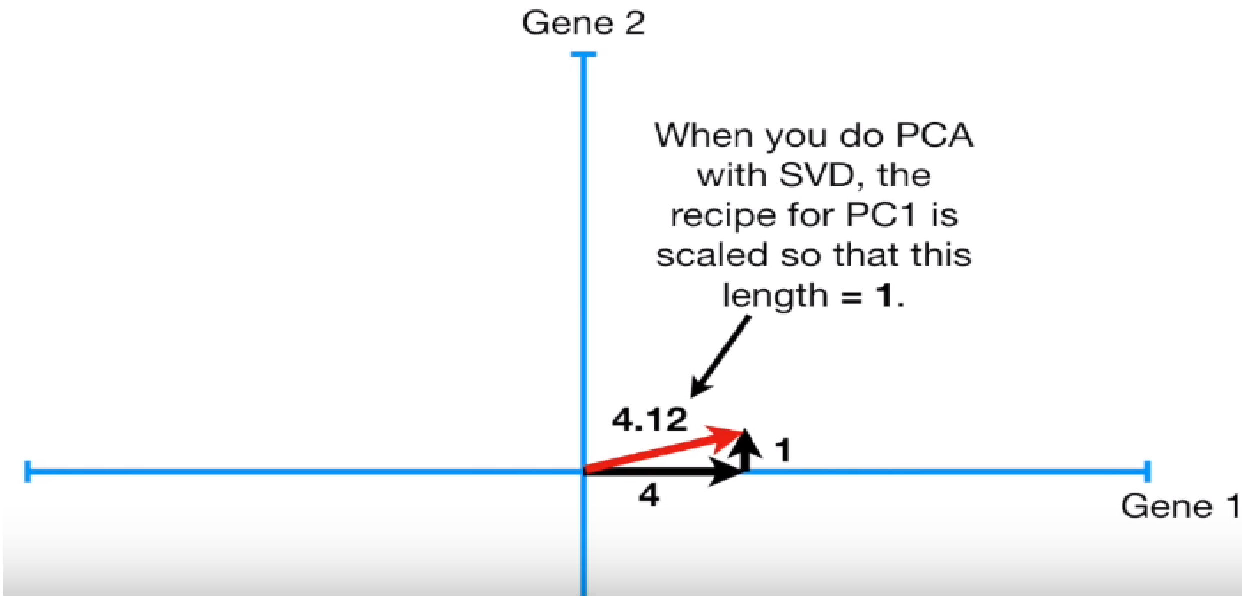

Gene 2
This line is called **Principal**
omponent 1. (**PC1** for short.)
Gene 1

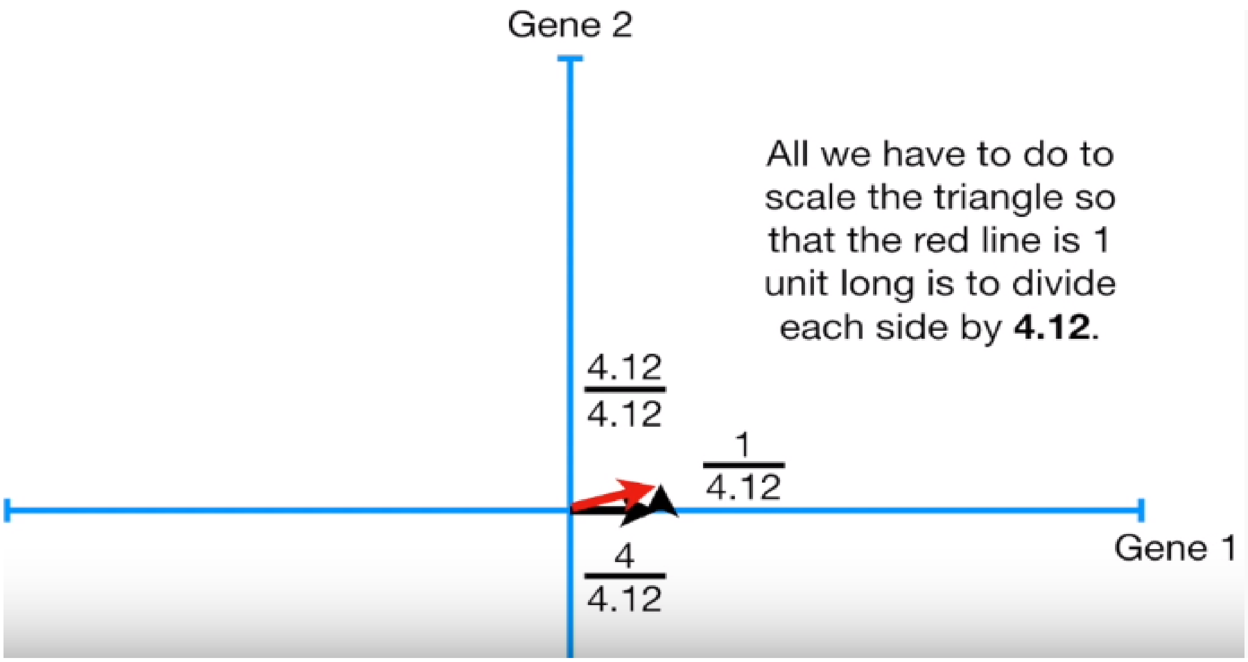
Geometric interpretation of projected data on principle component

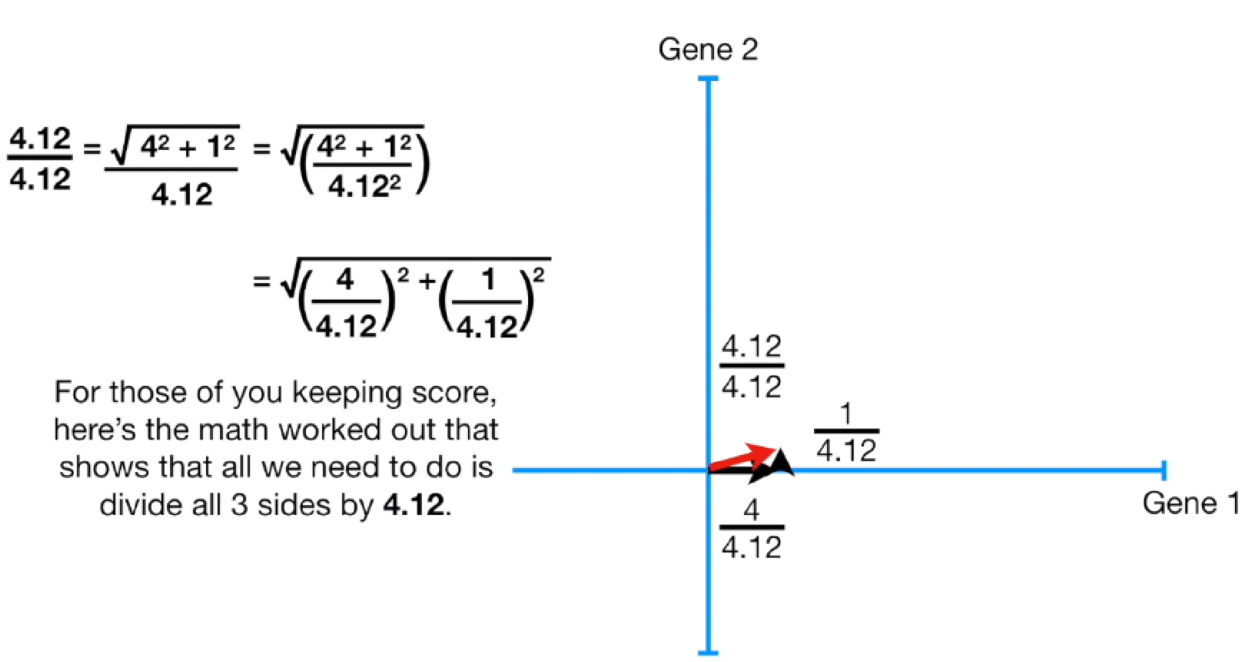




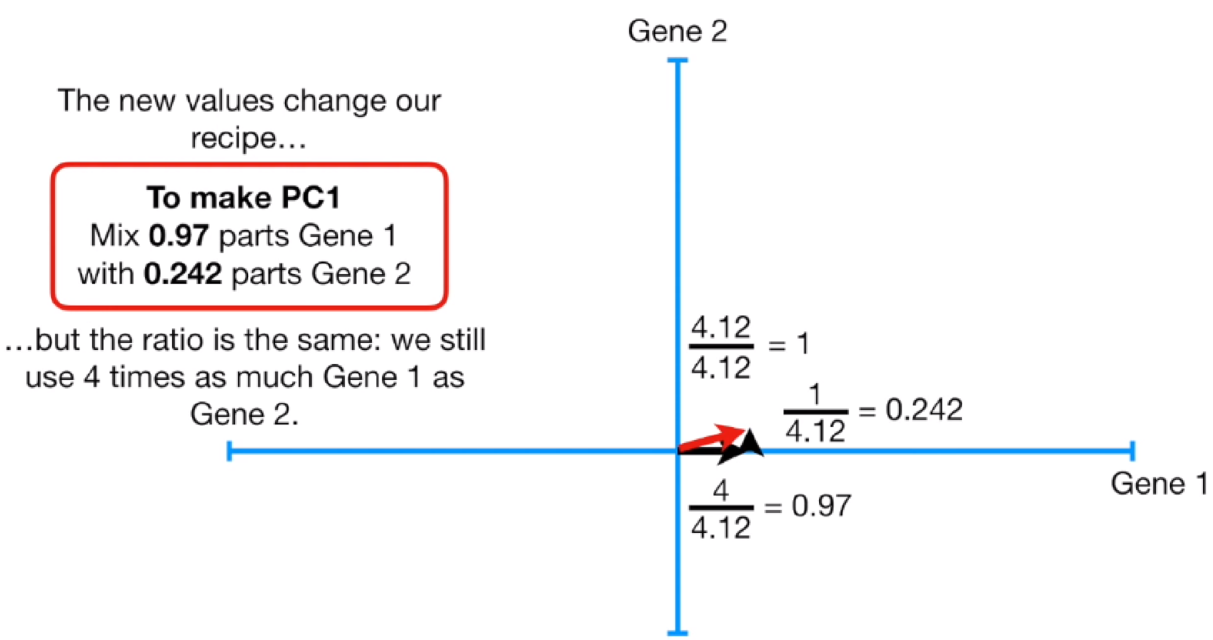
Projected line as a unit vector (eigen vector)



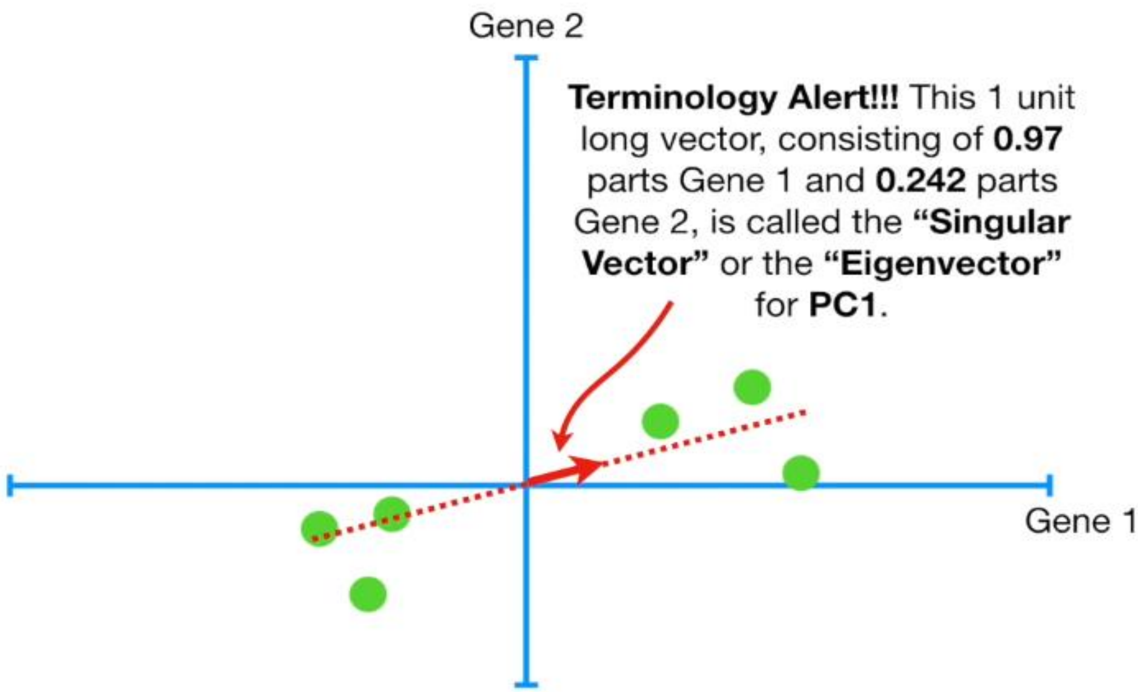


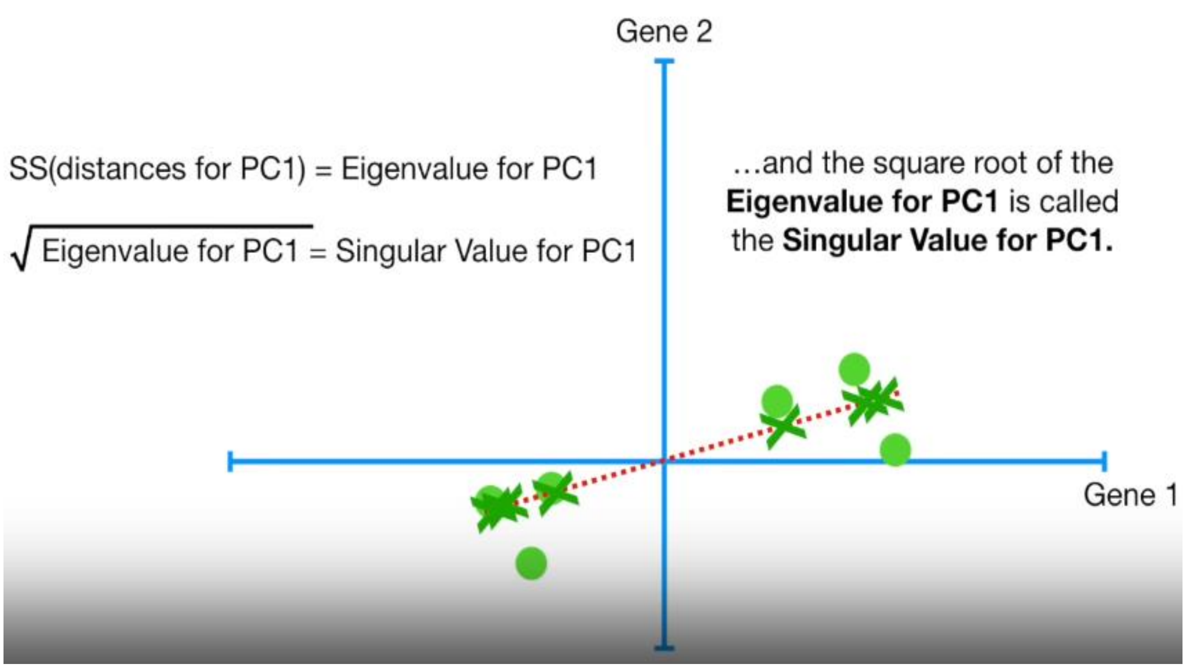


Percentage of variance explained for two samples in the direction of unit vector (line or eigen vector)

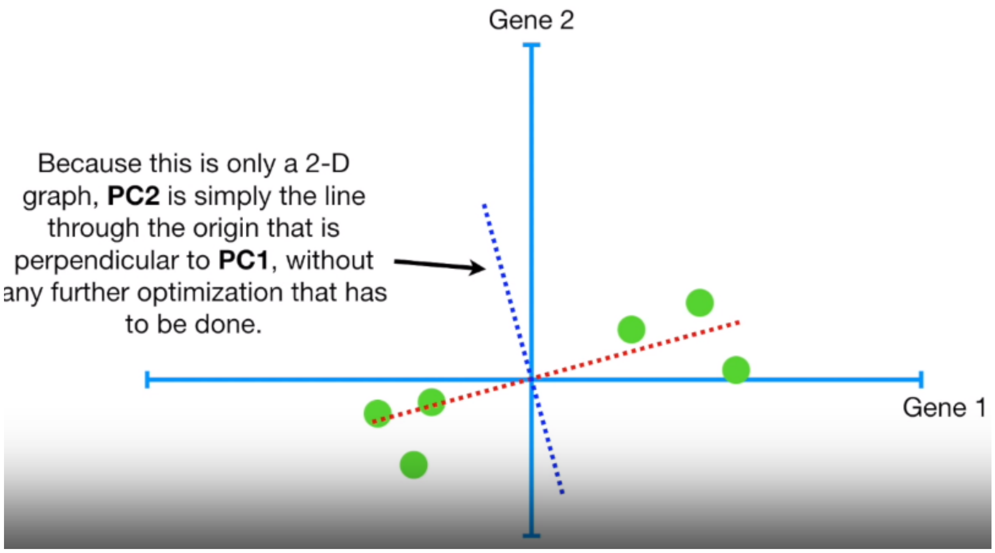


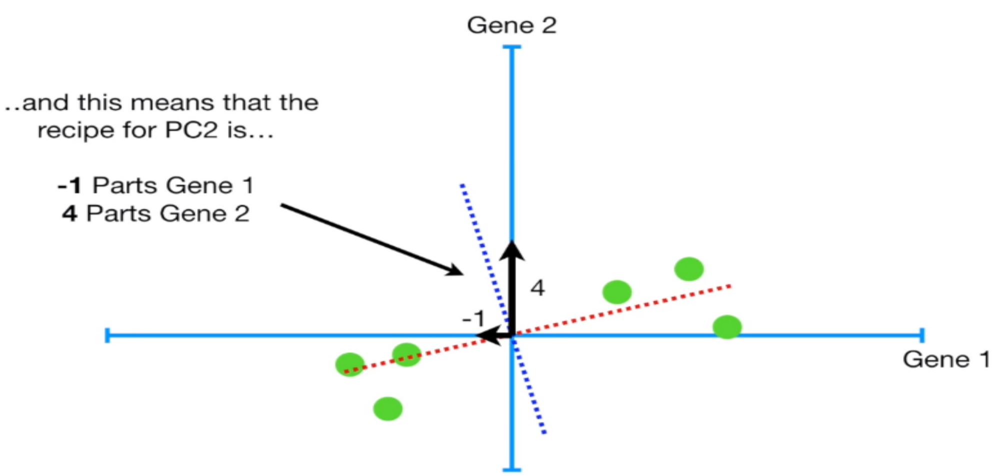
Eigen value interpretation

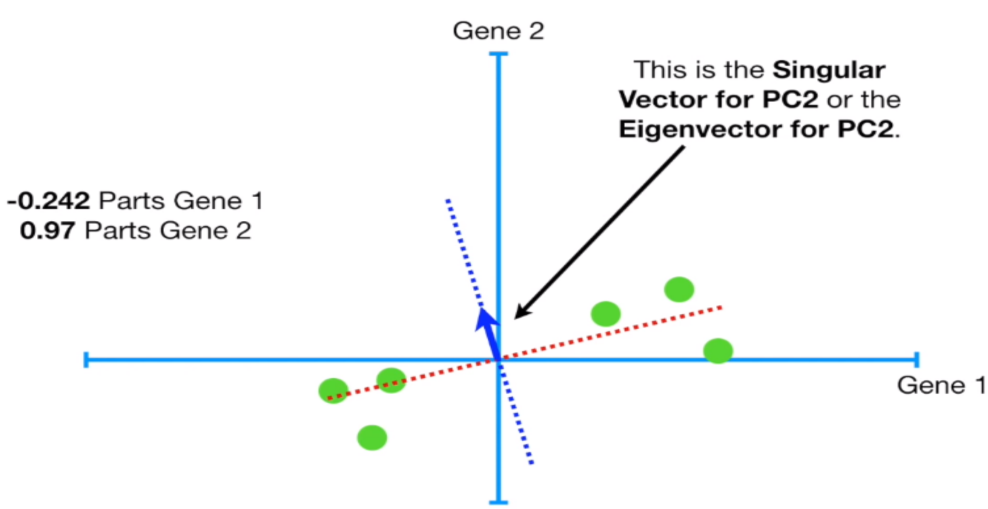


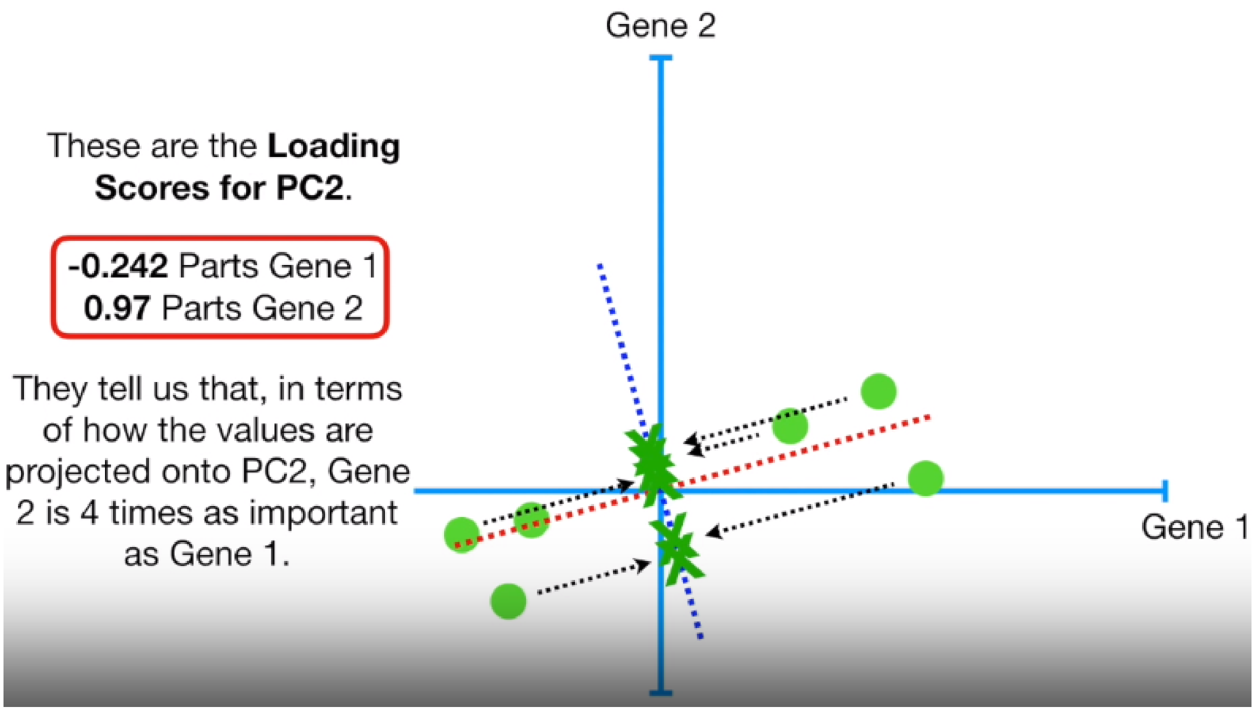


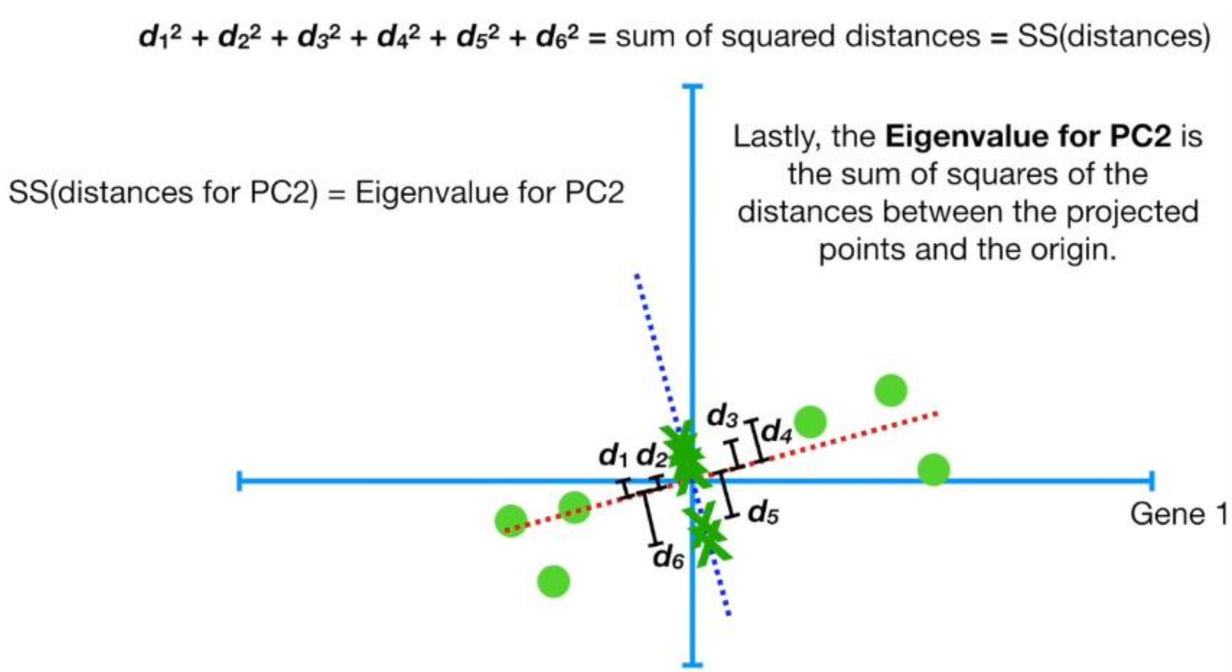
Deriving second principle components: -

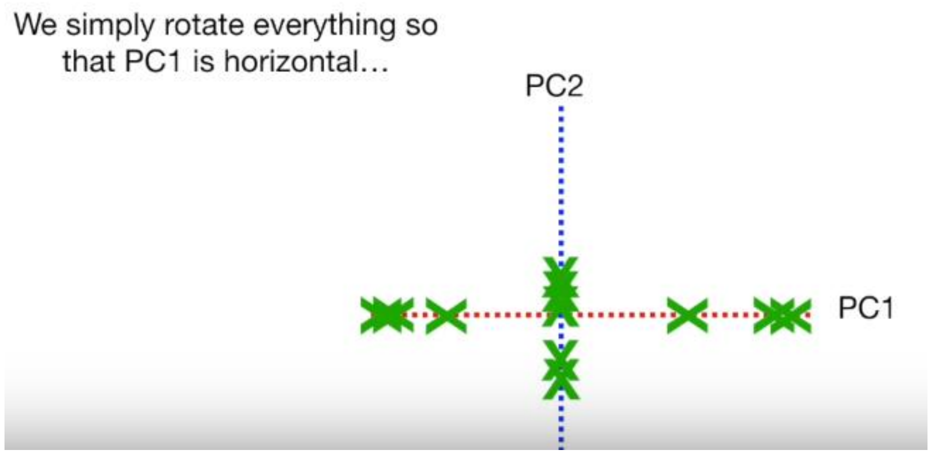


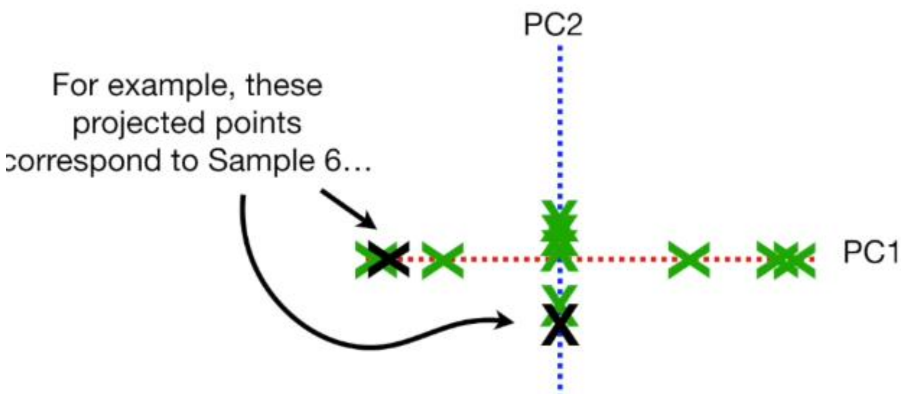


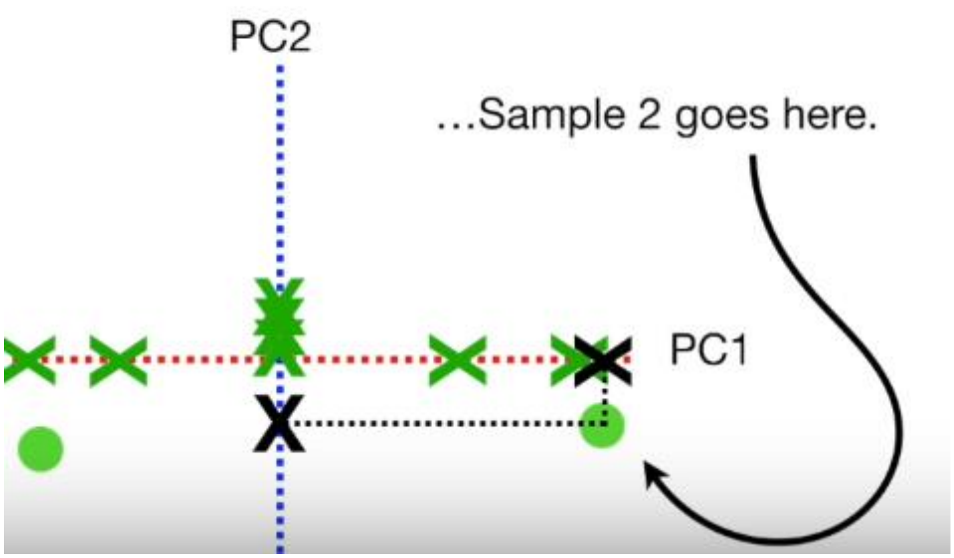



Gene 2
These are the **Loading Scores for PC2**.
-0.242 Parts Gene 1
0.97 Parts Gene 2
They tell us that, in terms of how the values are projected onto PC2, Gene 2 is 4 times as important as Gene 1.
Gene 1

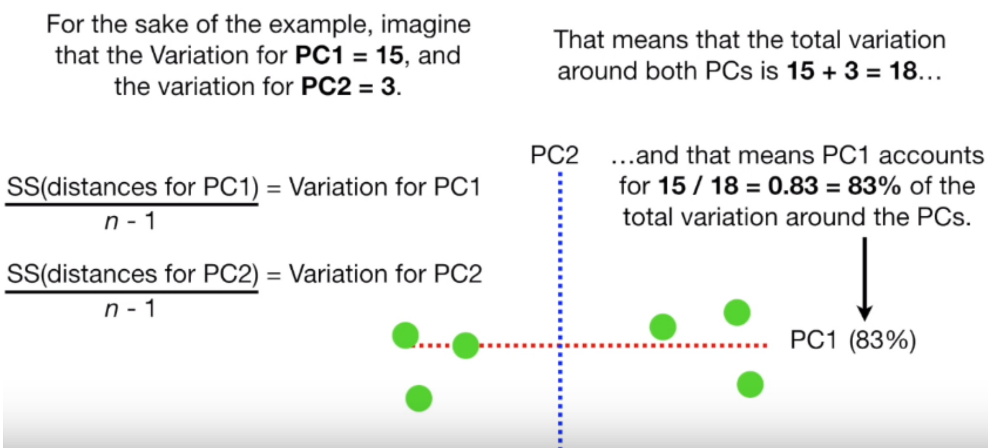

$d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2$ = sum of squared distances = SS(distances)
SS(distances for PC2) = Eigenvalue for PC2
Lastly, the **Eigenvalue for PC2** is the sum of squares of the distances between the projected points and the origin.
d_1 d_2 d_3 d_4 d_5 d_6
Gene 1

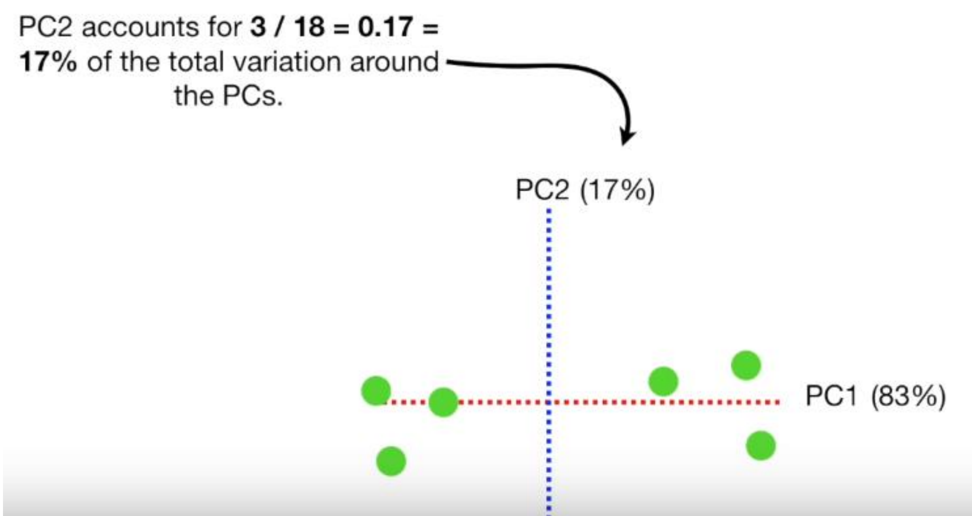

We simply rotate everything so that PC1 is horizontal…
PC2
PC1

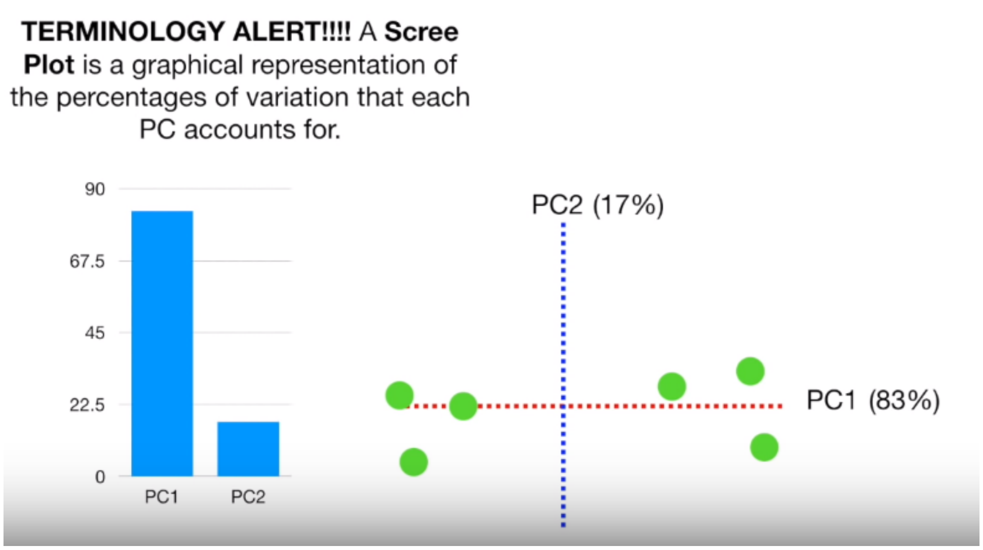

PC2
For example, these projected points correspond to Sample 6…
PC1


PC2
…Sample 2 goes here.
PC1

Variance explained in the direction of projected line (eigen vector)







Principle components and variance explained for more than two features: -