Principal Component Analysis :-

→

Gene \ Sample →	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

To represent this we require 4D which is not possible to visualize.

- So we are going to talk abt how PCA can take 4 or more gene measurements (ie the 4 or more dimensions of data) ie make a 2-D PCA plot.

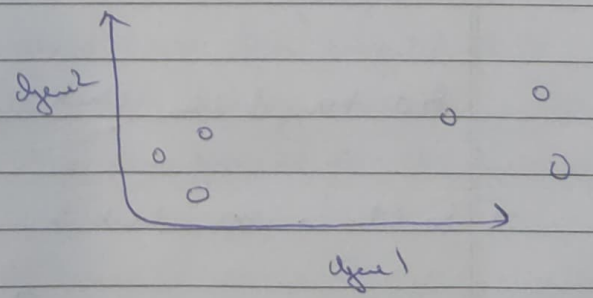
- We also talk abt how PCA can tell which variable (gene) is most valuable for clustering data

working of PCA

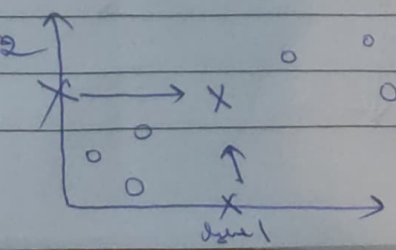
→

Gene \ Sample →	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

1) Plot the data



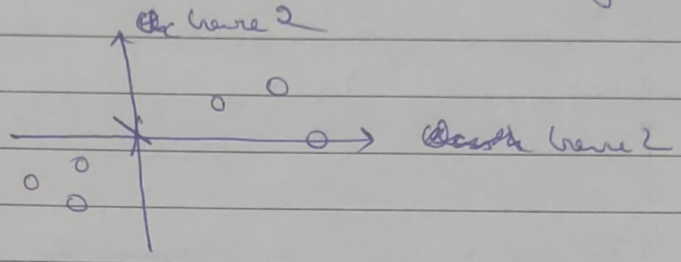
2) Calc the avg measurements of Gene 1 ie avg value of Gene 2



with avg values we can calculate the center of data

Now we will focus on graph, no longer need the original data

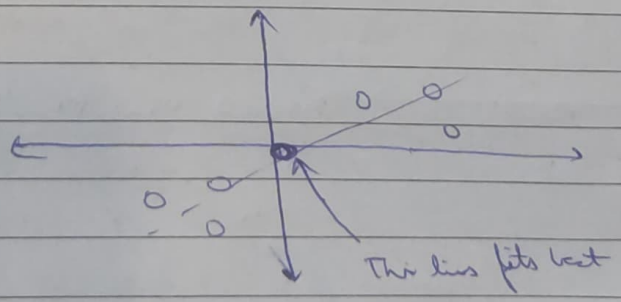
3) Now shift the data that centre is on origin



Note: Shifting the data did not change how the data pts are positioned relative to each other

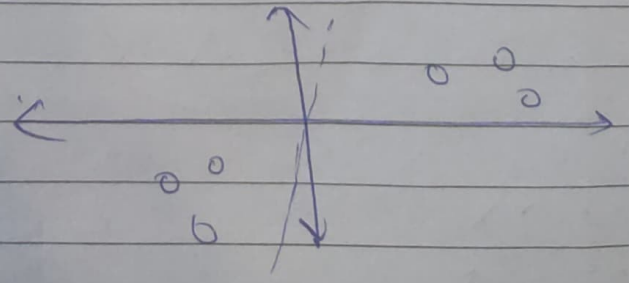
4) Try to fit a line

We start by drawing a random line that goes through origin & then we rotate the line until it fits the data as well as it can, given that it has to go through origin



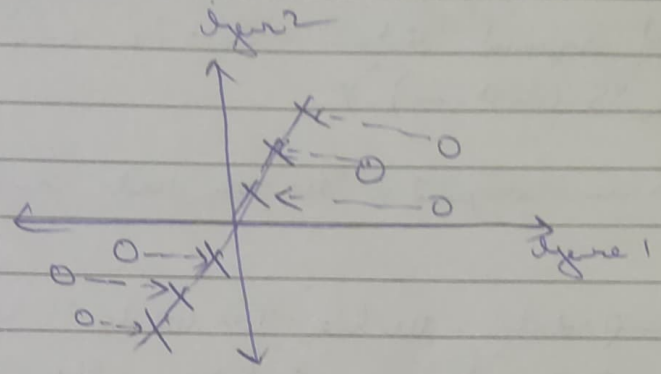
But how decide which line is the best?

→ So let's go back to "random line that goes through origin"



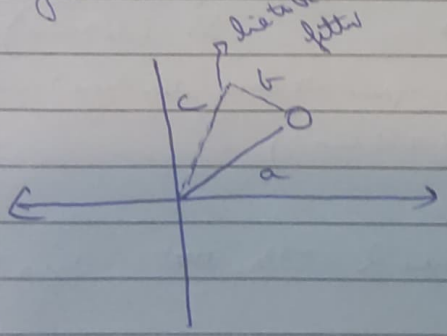
PAGE EDGE
Date : / /

To quantify how good the line fits the data, PCA projects data onto it



Then it can either measure the distances from the data to the line and try to find the line that minimizes those distances or it can try to find the line that maximizes the distance from the projected pts to the origin

The projected pts distance on line gets longer when the line fits better

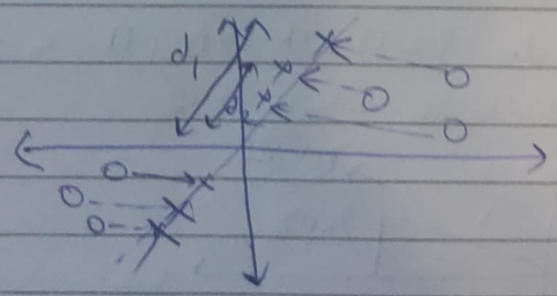


a is the distance from pt to origin (fixed)

$a^2 = b^2 + c^2$ (Phythagorean thm)

In order to minimize b, we need to increase c

Minimizing b, the distance from pt to line but its easier to calc c, the distance from projected pt to origin, so PCA finds best line by maximizing the sum of squared distances from the projected pts to the origin

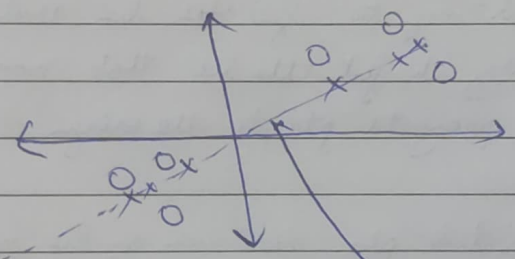


So 6 distances $d_1, d_2, d_3, d_4, d_5, d_6$

Sum of squared distances $= d_1^2 + d_2^2 + d_3^2 + d_4^2 + d_5^2 + d_6^2$
(SS (distances)) =

↑
Distance is squared so the -ve don't cancel +ve.

We keep rotating the line till we don't get a line with the largest sum of squared distances b/w the projected points and the origin

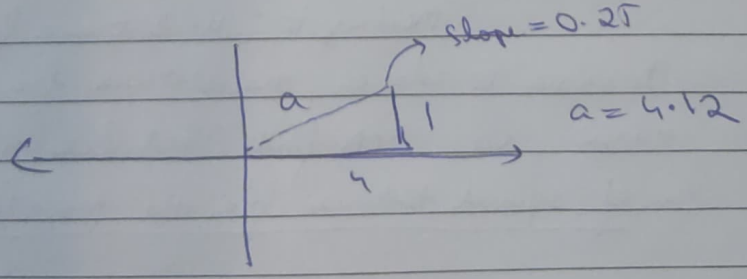


↑
PC1 has a slope of 0.25

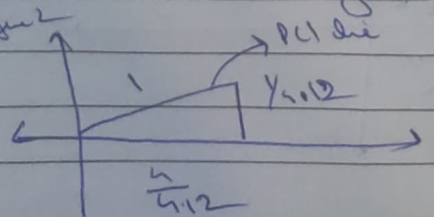
At last we end up with line having largest SS (distances)

(This means that data mostly spread out along the axis 1 axis & only a little bit spread out along axis 2 axis)

The ratio of axis 1 to axis 2 with that axis 1 is more imp



When you do PCA with SVD, a must be scaled to 1.
So ∴ we divide all sides by 4.12.

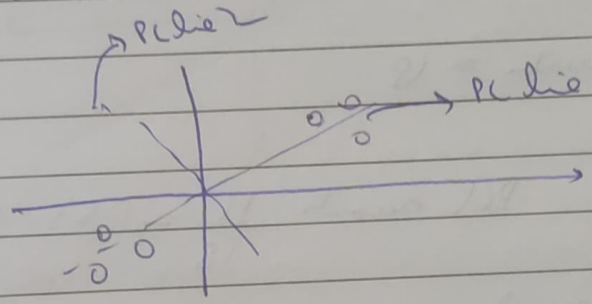


→ This unit long vector is called "Singular vector" or "Eigenvector" for PC1

→ $\frac{SS(\text{distance for PC1})}{n-1}$ = Eigenvalue for PC1

→ $\sqrt{SS(\text{distance for PC1})}$ = Singular value for PC1

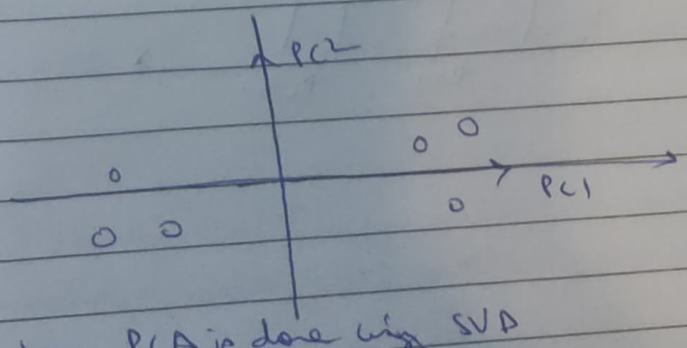
→ PC2 line is line to PC1



According to PC2 gene 2 is more up than gene 1.

$\frac{SS(\text{distance for PC2})}{n-1}$ = Eigenvalue for PC2

→ To get final plot.
We rotate everything so we get PC1 as horizontal



That is how PCA is done using SVD

Eigen values are just measure of variation

$\frac{SS(\text{distance for PC1})}{n-1}$ = variation for PC1

$\frac{SS(\text{distance for PC2})}{n-1}$ = variation for PC2

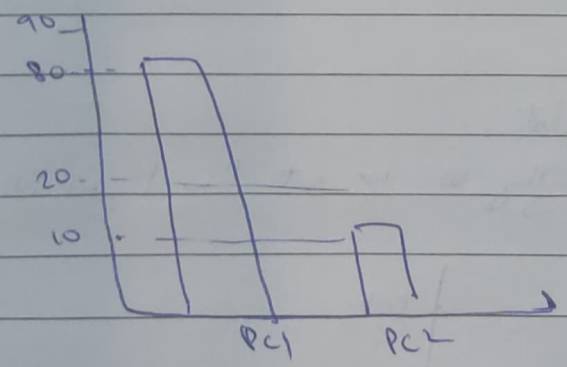
Eg variation PC1 = 15
variation PC2 = 3

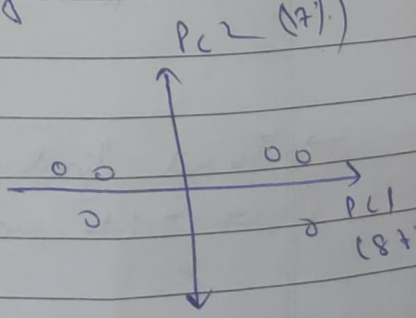
Total variation = 18

That means PC1 accounts $\frac{15}{18} = 0.83 = 83\%$ of total variation around PCs

PC2 accounts for $\frac{3}{18} = 0.17 = 17\%$ of total variation around PCs

→ A Scree Plot - graphical representation of % of variation that each PC accounts for.

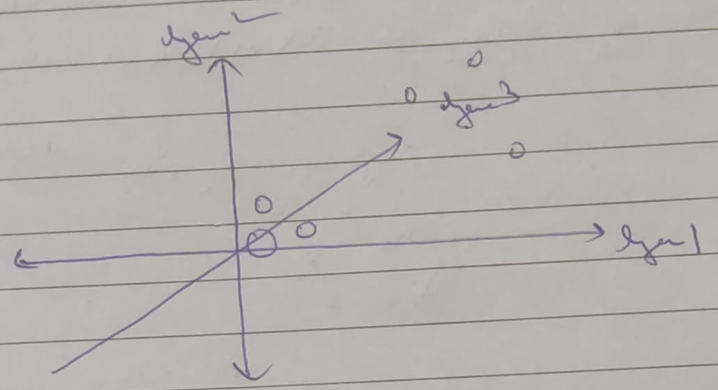




PAGE EDGE
Date : / /

→ PCA with 3 variables

	Mark1	Mark2	Mark3	Mark4	Mark5	Mark6
Gene1	10	4	8	3	2	1
Gene2	6	4	5	3	2.8	1
Gene3	12	9	10	2.5	1.3	2



- You enter the data
- Fit best fitting line PC1

0.62 Parts Gene1
0.15 Parts Gene2
0.77 Parts Gene3 ∴ Gene3 - most imp for PC1

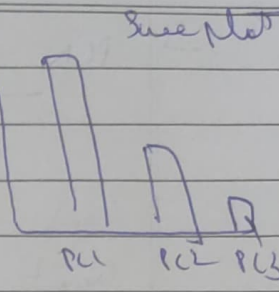
- Fit PC2 ⊥ to PC1
- Fit PC3 ⊥ to PC1 & PC2 & all go through origin

✗ In theory, No of PC = No of variables / Samples (whichever is less)

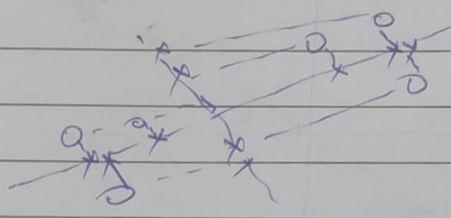
- Once you have all PC, you can use eigenvalues (i.e SS(distance)) to determine proportion of variation that each PC accounts for

Eg PC1 - 74% variation, PC2 - 15%, PC3 - 6%.

∴ It runs a 2D graph using just PC1 & PC2 good approximation of 3D graph since it would account for 94% of variation in data.

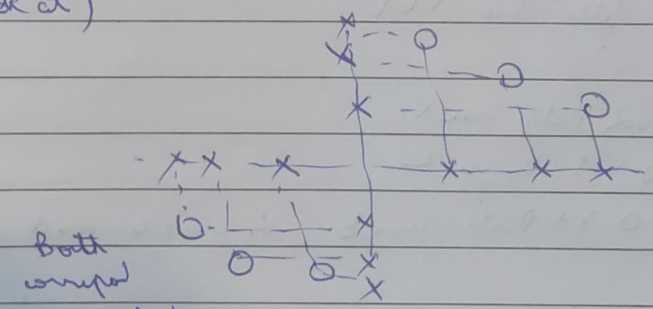


- To convert the 3D graph into a 2D PCA, we just strip away everything but PC1 & PC2 & data.



Project the samples onto PC1 & PC2

Then we rotate such that PC1 - horizontal, PC2 - vertical (easier to look at)



→ If we have higher dimension, we get scree plot & depends on the data from scree plot we choose the PC.