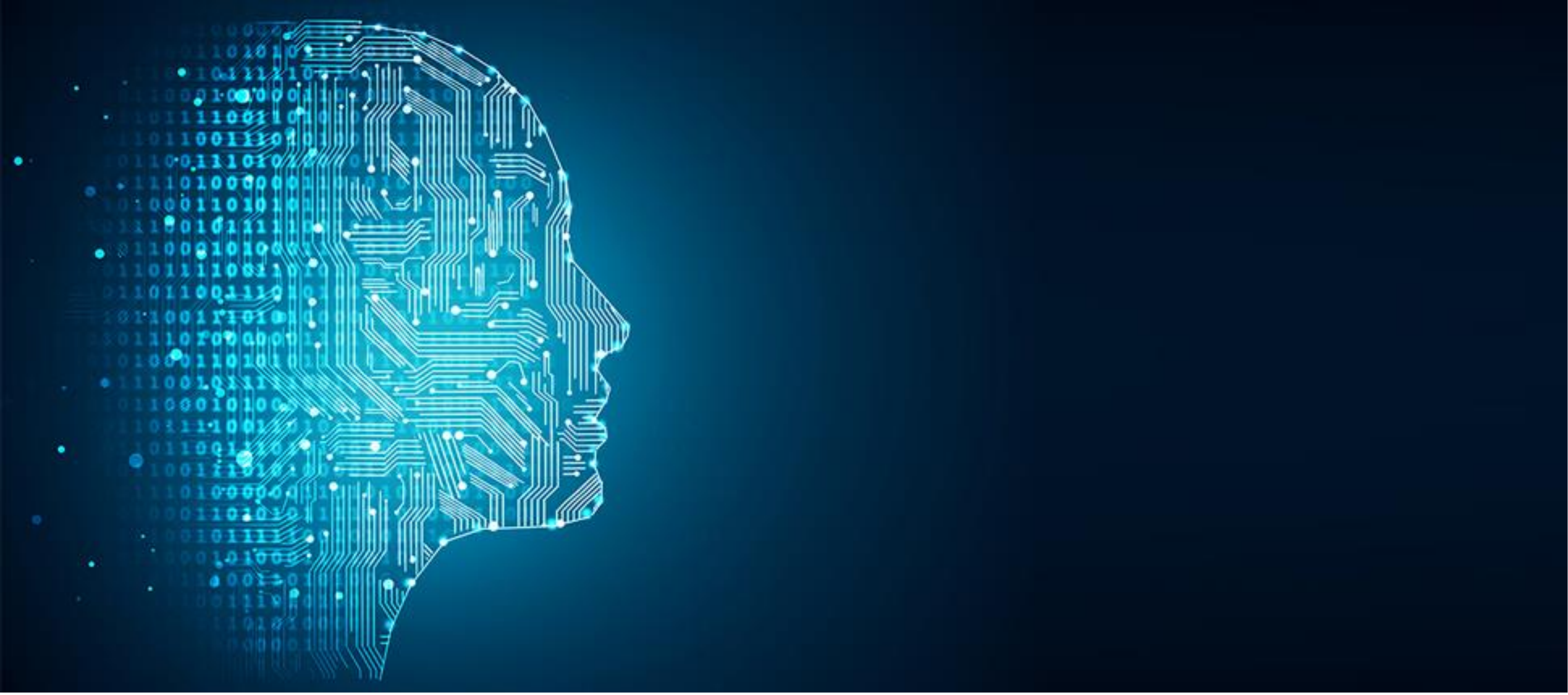


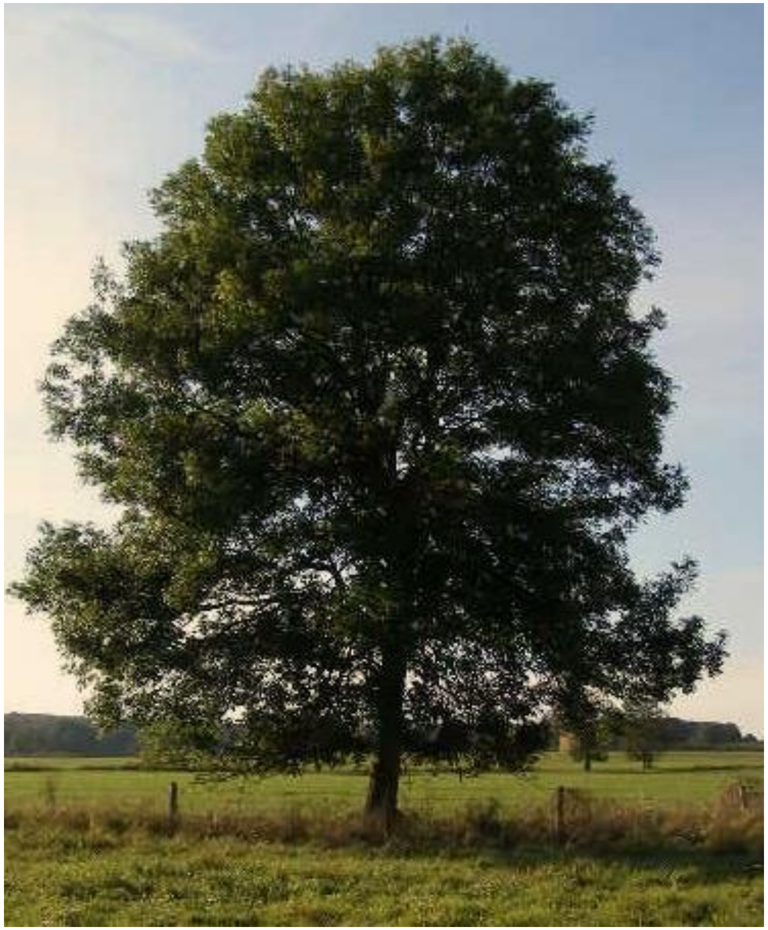
Concepts in Artificial Intelligence & Machine Learning Technologies

Machine Learning Basis – Decision tree, Kmeans, PCA

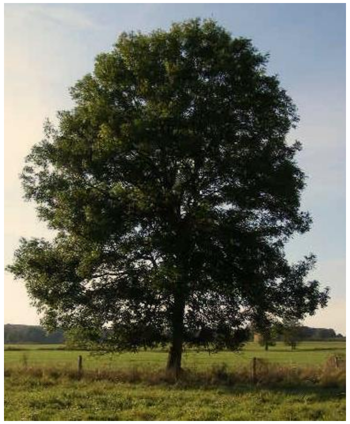
By Dr. Hu Wang, Dr. Wei Zhang

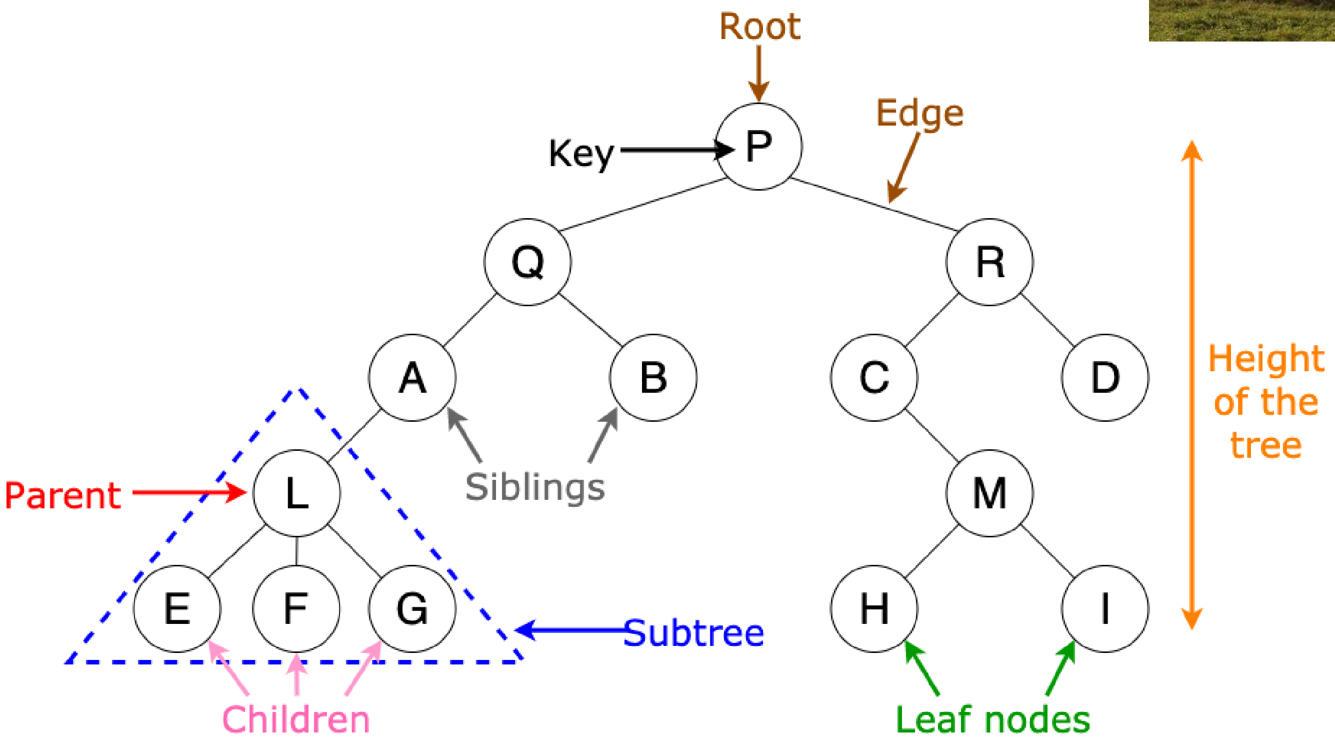
Decision Tree

- Decision Tree --- what is a tree

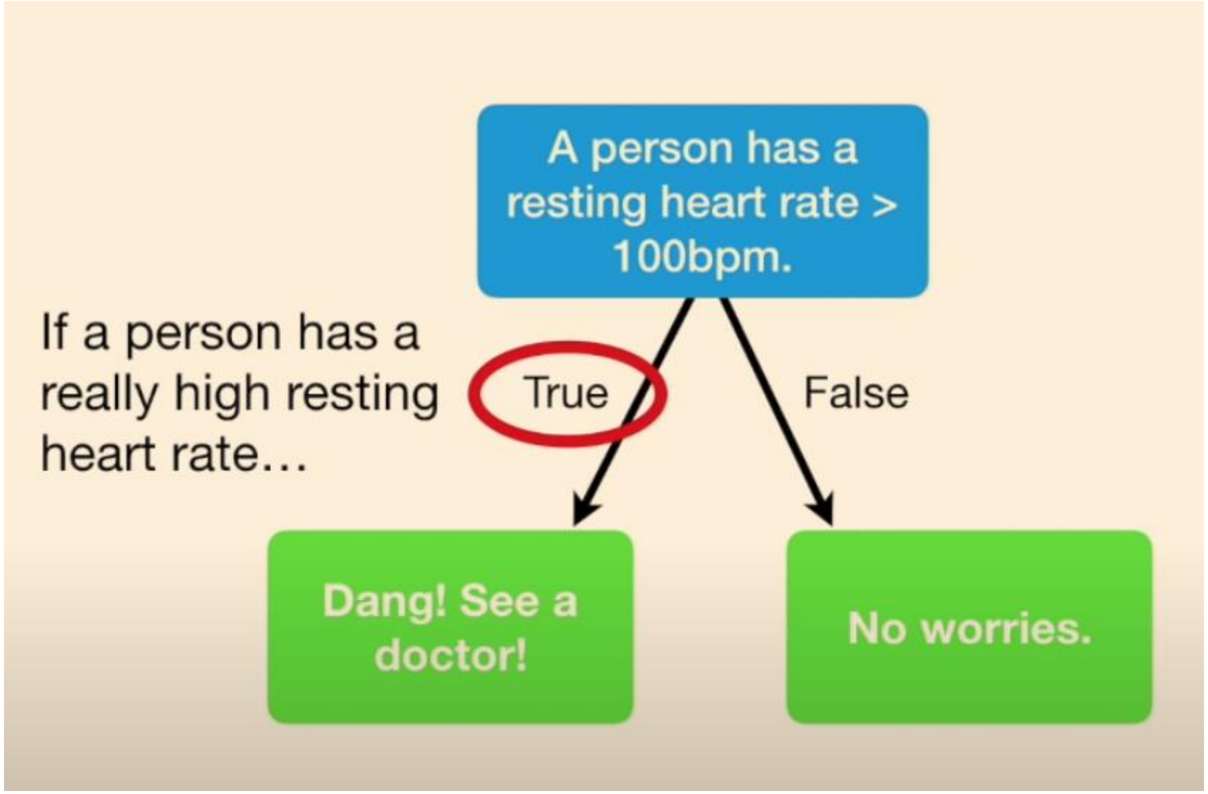


- Decision Tree --- what is a tree



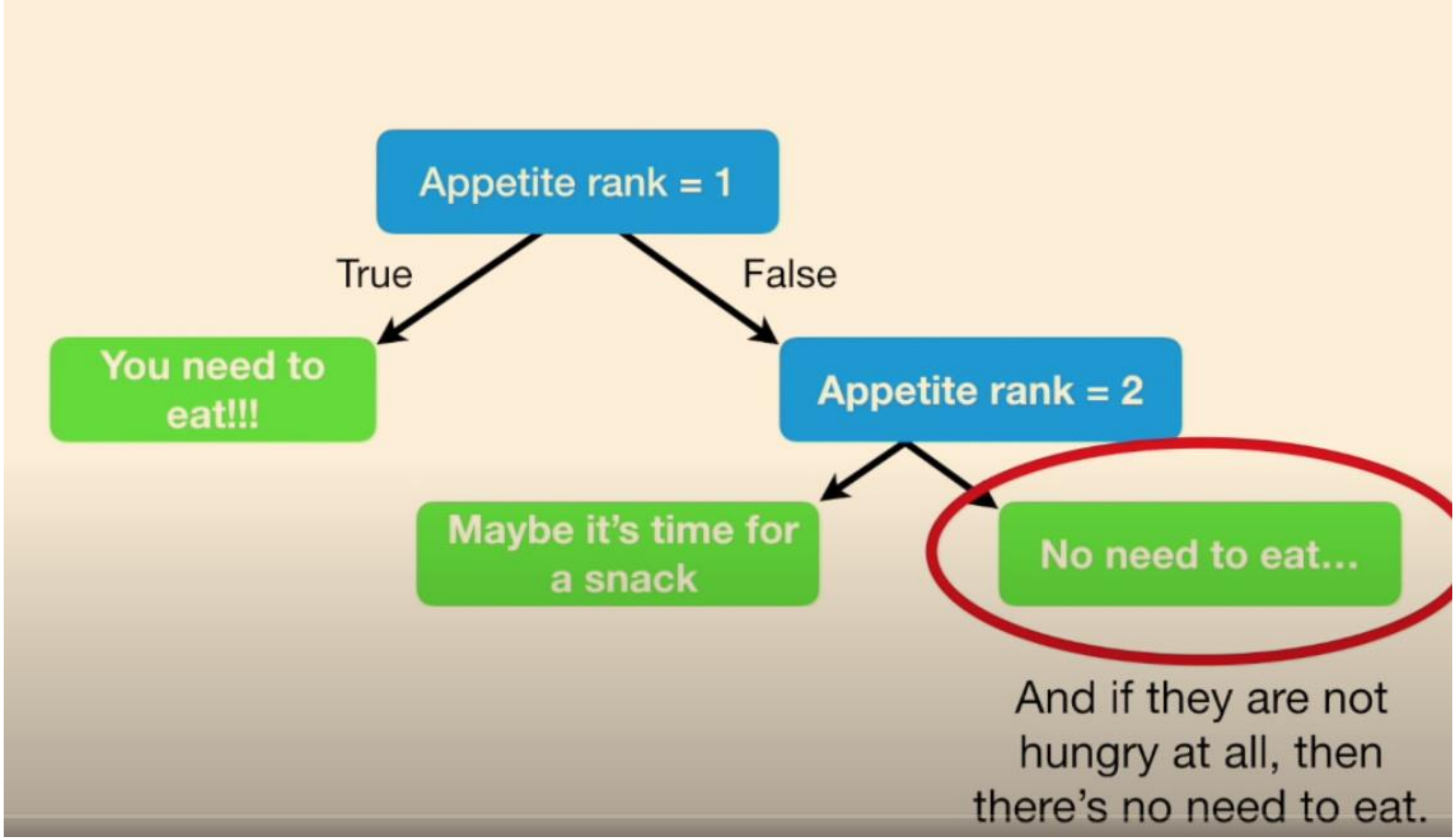


- # Decision Tree



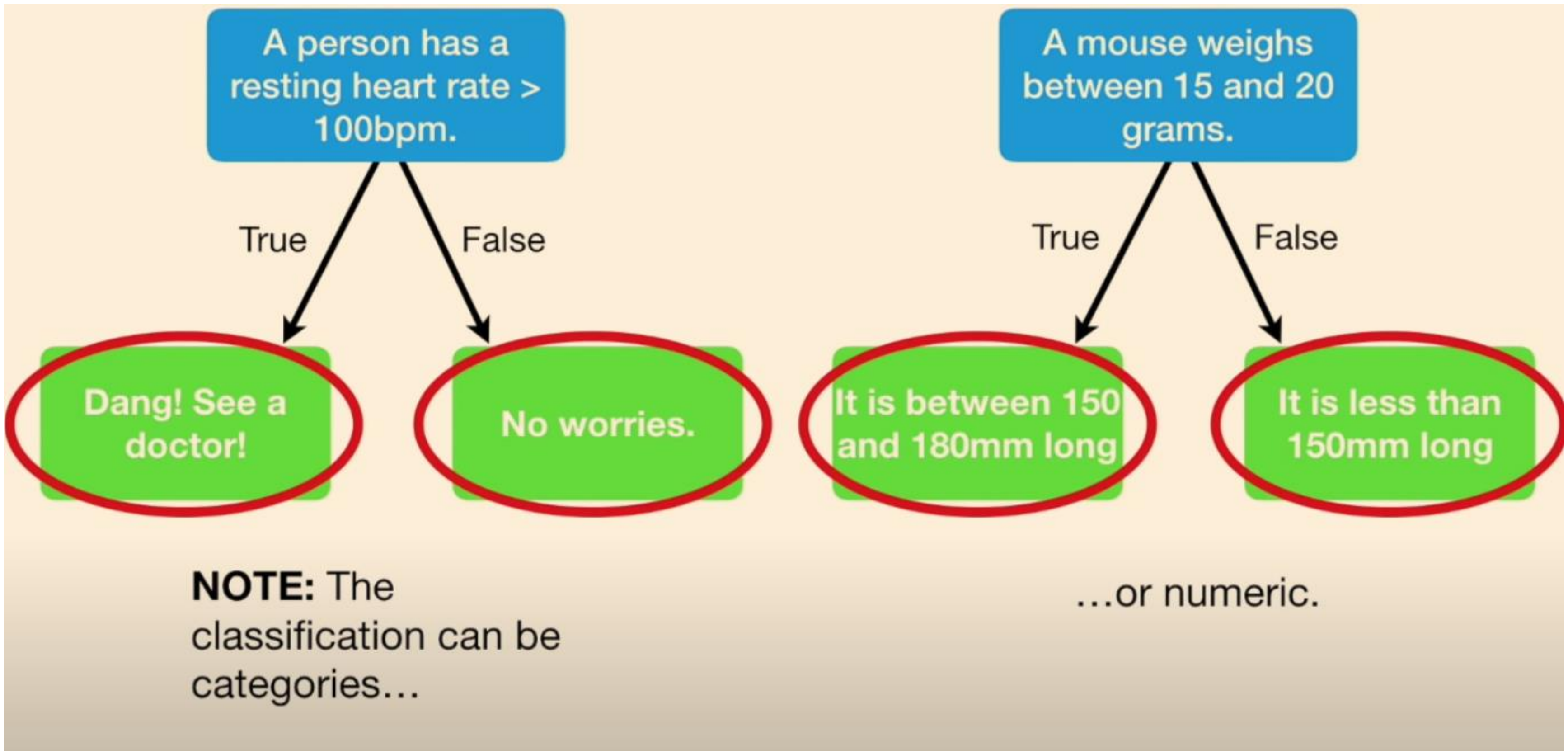
https://www.youtube.com/watch?v=7VeUPuFGJHk

- # Decision Tree



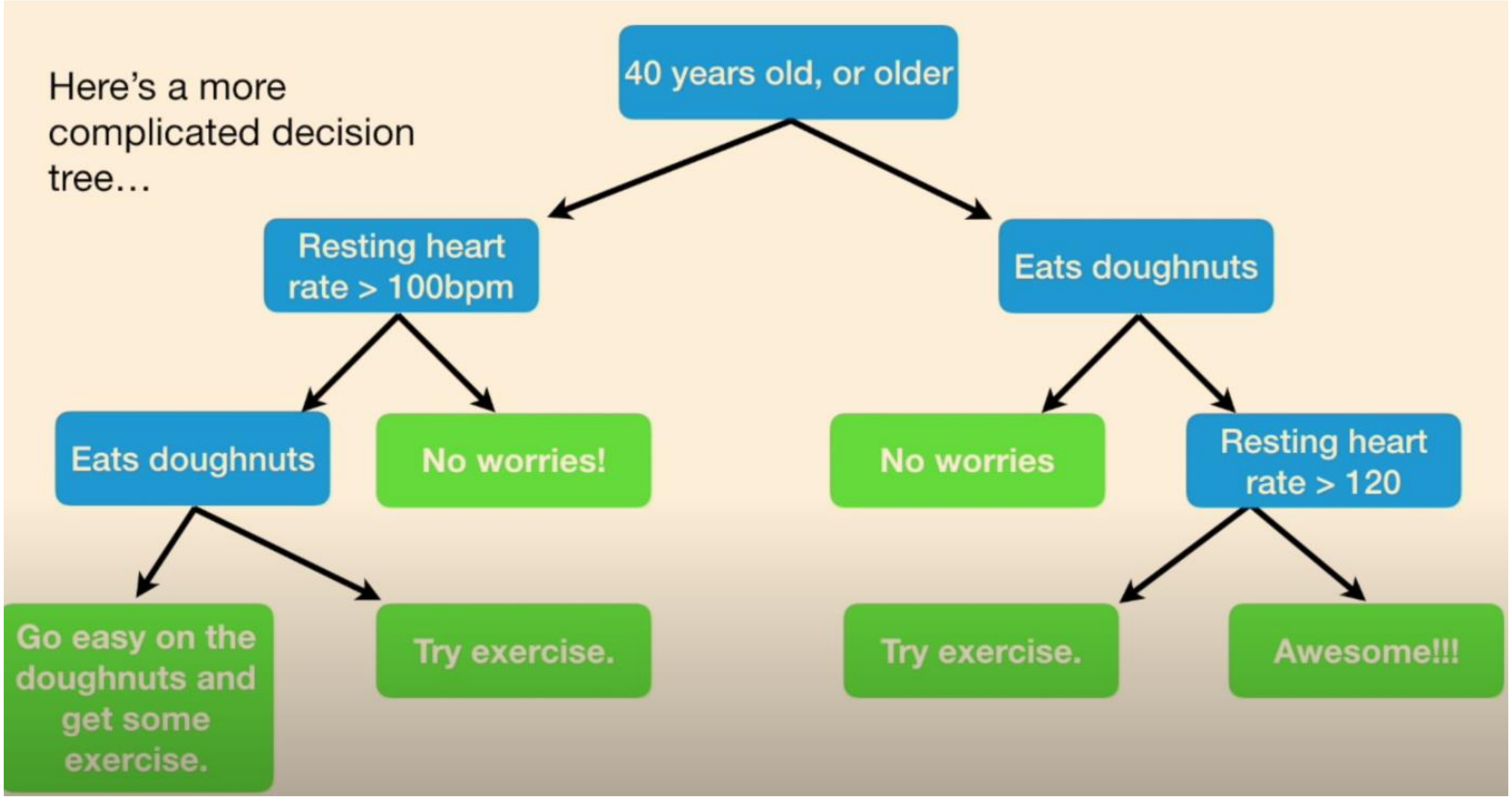
https://www.youtube.com/watch?v=7VeUPuFGJHk

- # Decision Tree



https://www.youtube.com/watch?v=7VeUPuFGJHk

Decision Tree



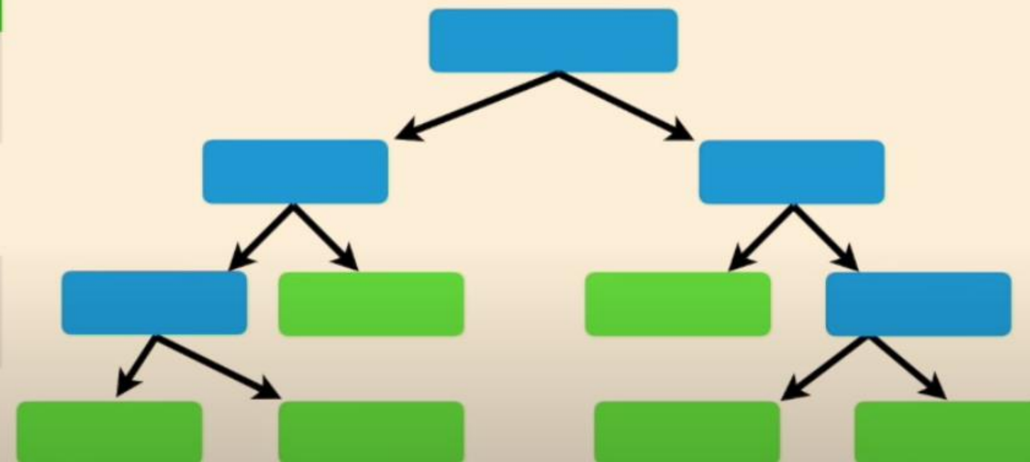
https://www.youtube.com/watch?v=7VeUPuFGJHk

Decision Tree

Now we are ready to talk about how to go from a raw table of data…

Chest Pain	Good Blood Circulation	Blocked Arteries	Heart Disease
No	No	No	No
Yes	Yes	Yes	Yes
Yes	Yes	No	No
Yes	No	???	Yes
etc…	etc…	etc…	etc…

…to a decision tree!!!



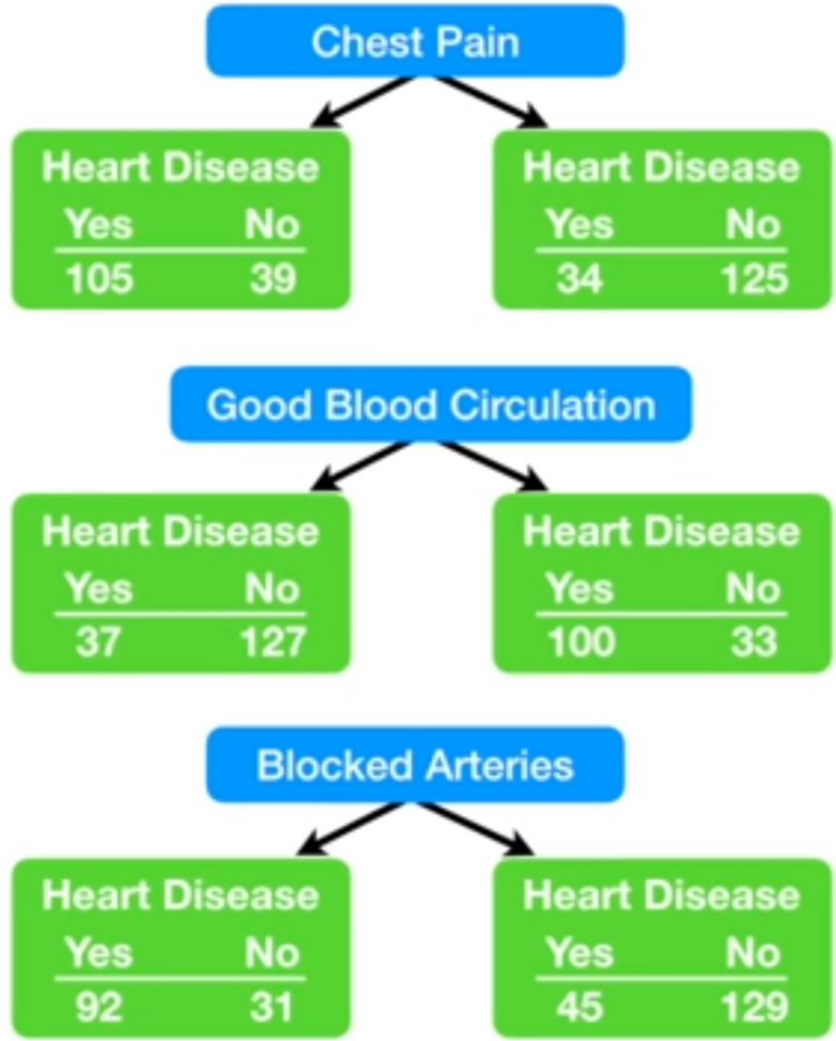
https://www.youtube.com/watch?v=7VeUPuFGJHk

- ## Decision Tree – Root Node

Gini impurity for Chest Pain = 0.364

Gini impurity for Good Blood Circulation = 0.360

Gini impurity for Blocked Arteries = 0.381

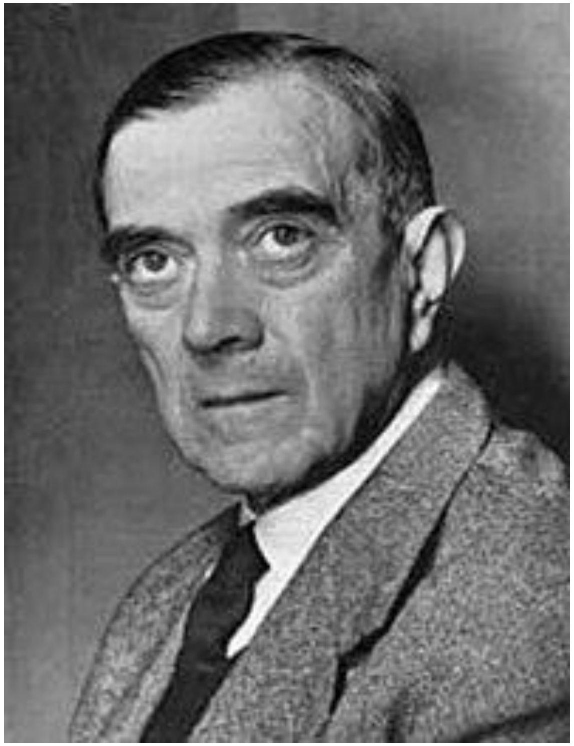


Decision Tree – Gini Impurity

- Then the Gini Impurity of the dataset D is defined as:

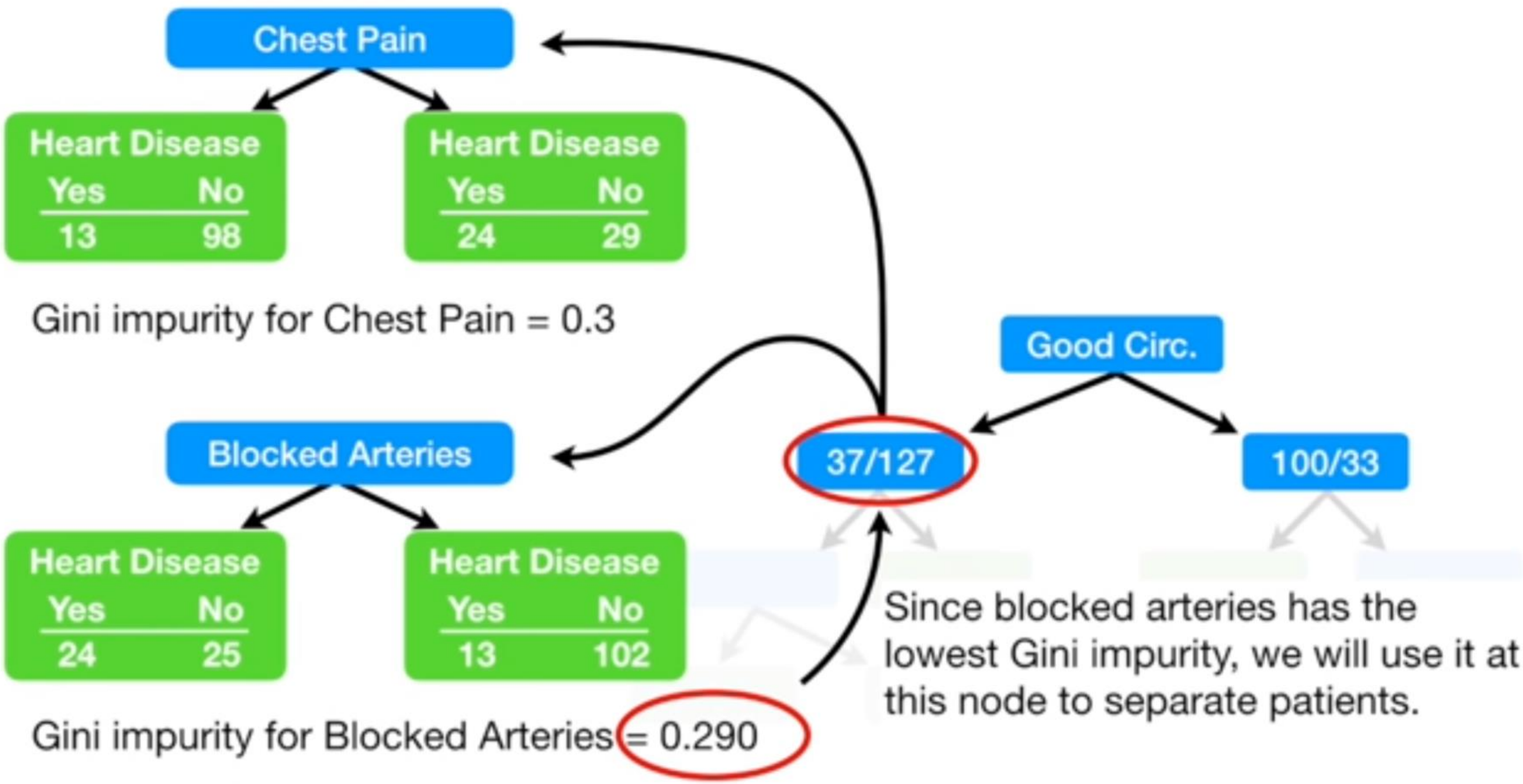
$$Gini(D) = 1 - \sum_{i=1}^{k} p_i^2$$

  - D: the data set
  - k: number of classes
  - p_i: The probability of samples belonging to class i at a given node.

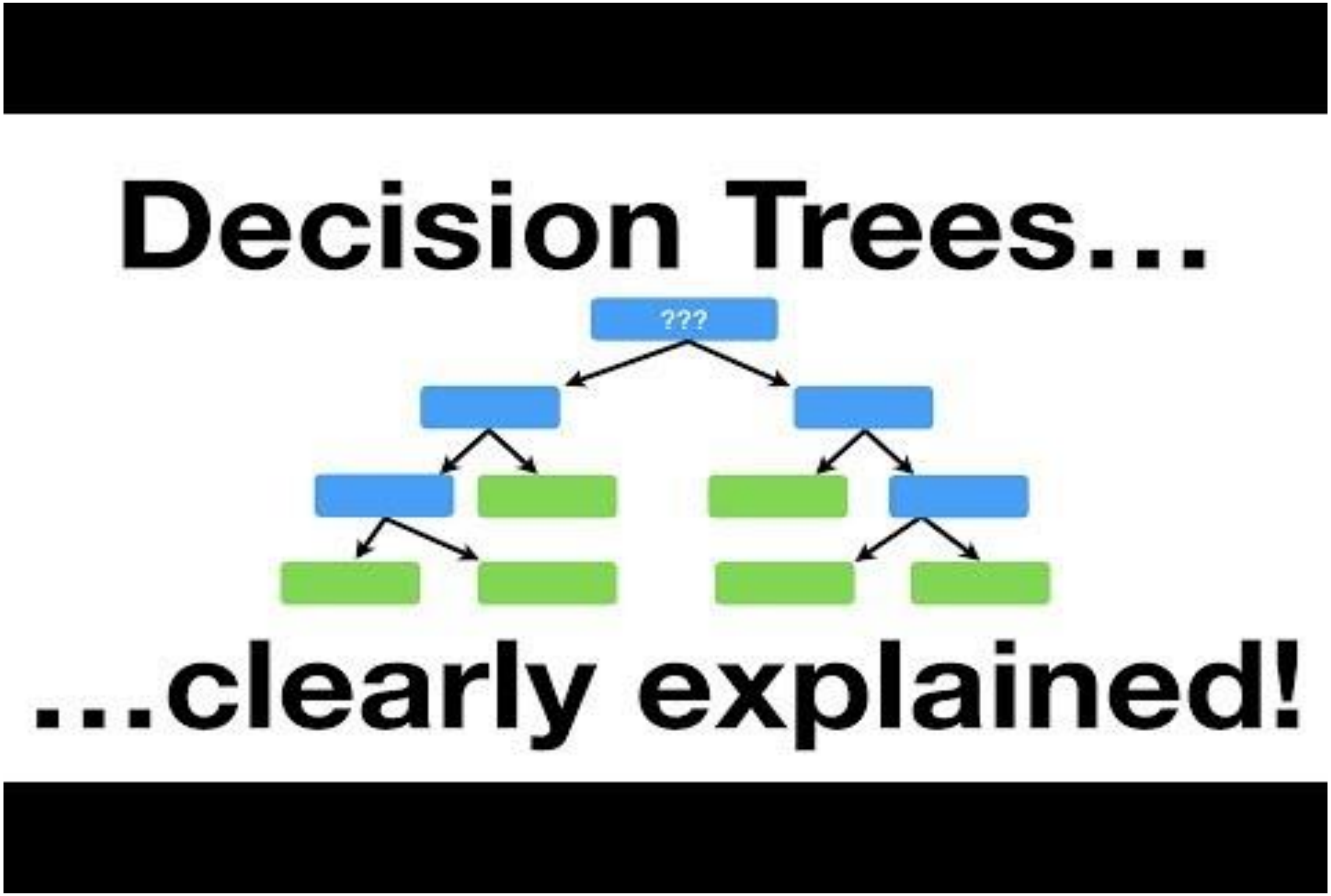


It was developed by the Italian statistician and sociologist Corrado Gini and published in his 1912 paper *Variability and Mutability*

Decision Tree – Further Split

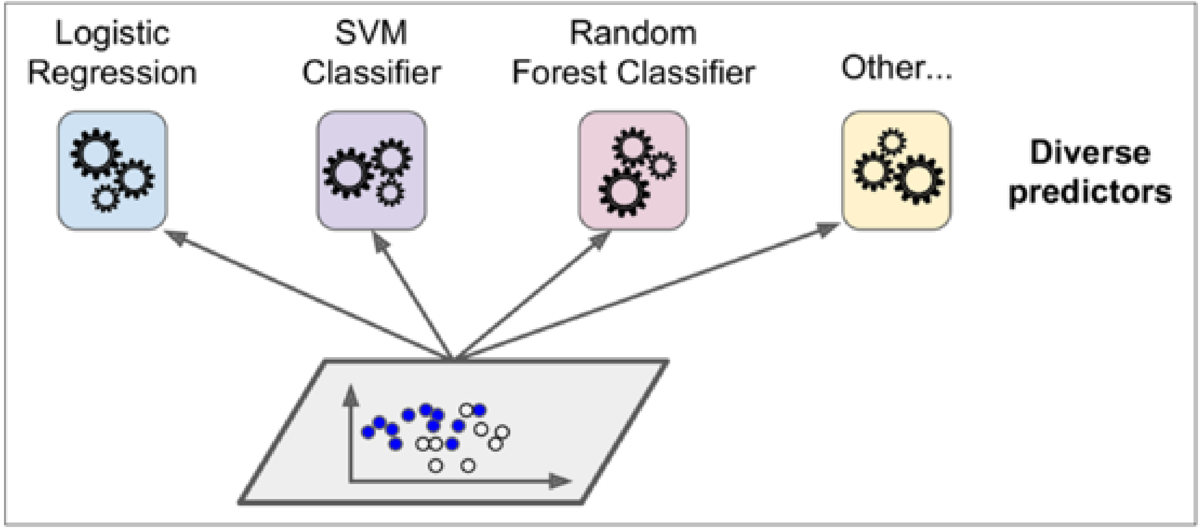


- Decision Tree

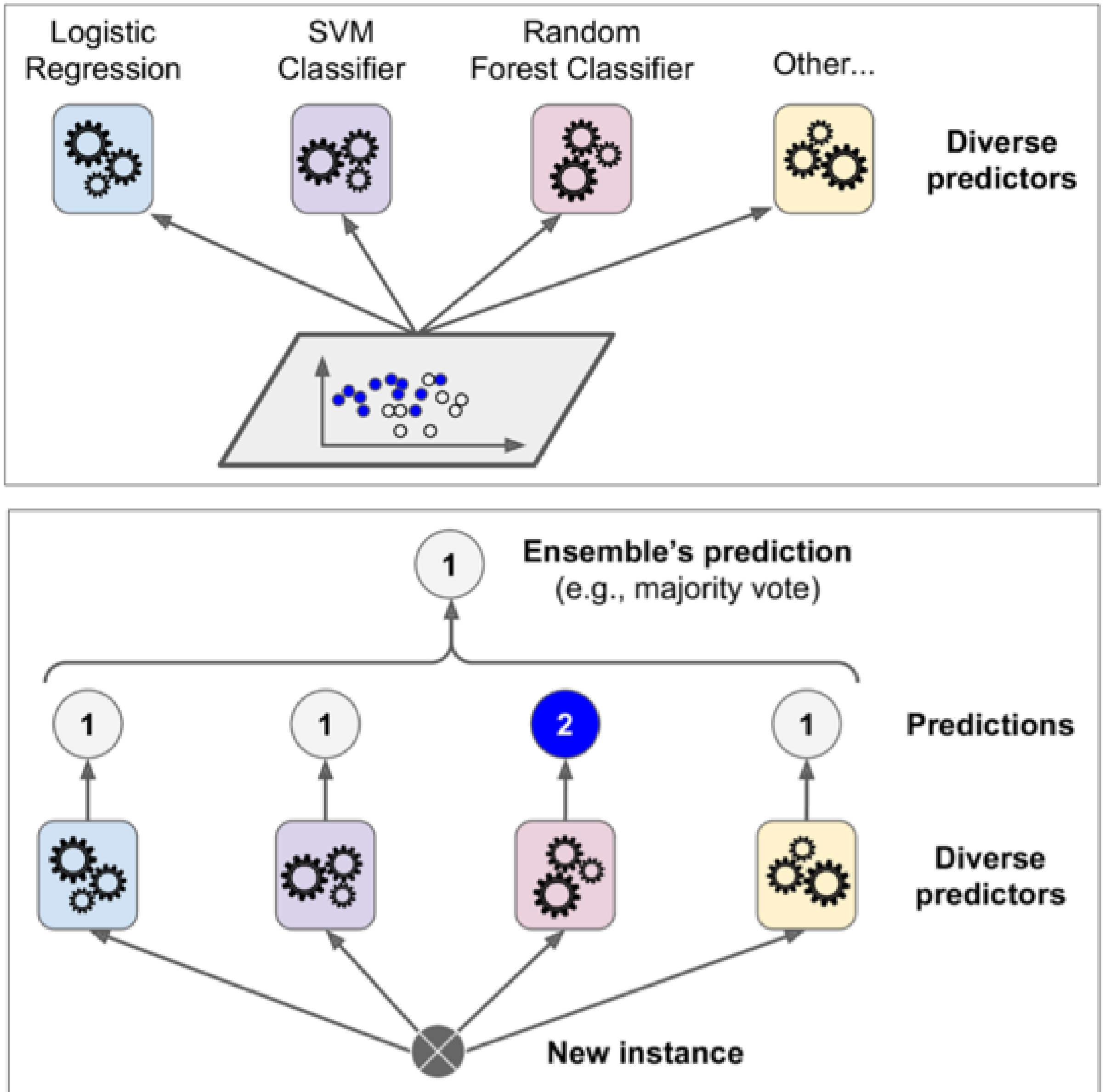


https://www.youtube.com/watch?v=7VeUPuFGJHk

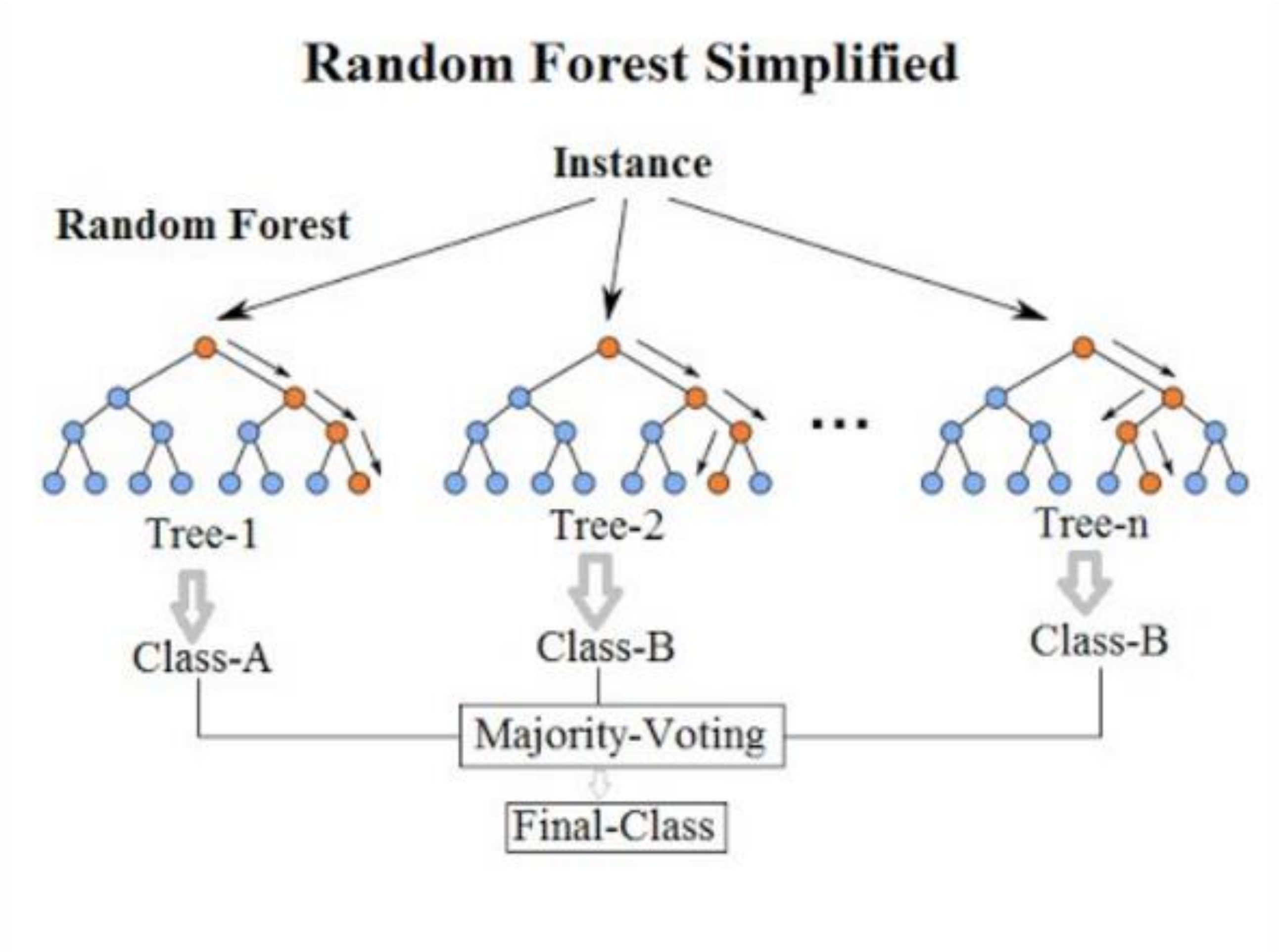
- # Ensemble Learning



- # Ensemble Learning

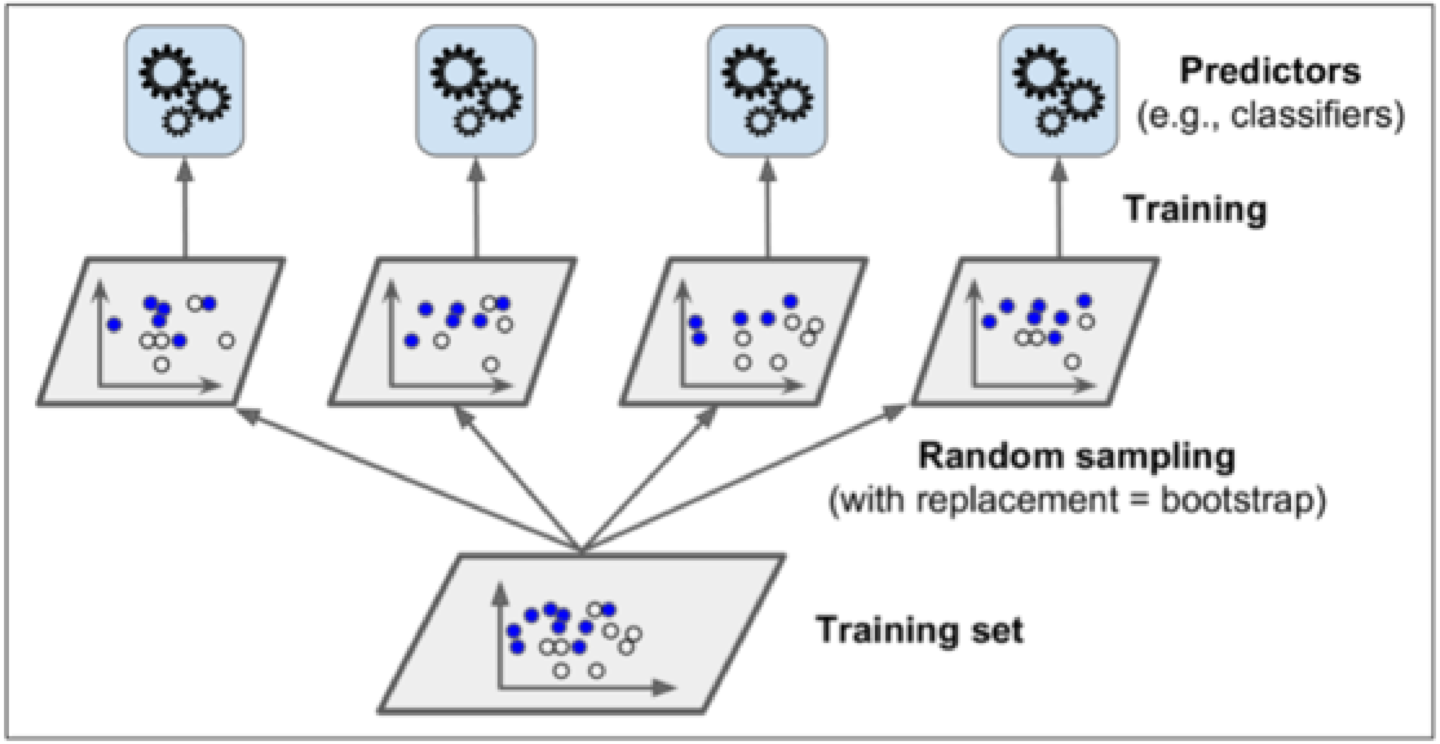


- ## Random Forest



Random forest can limit the instability by the average predicted value of multiple trees

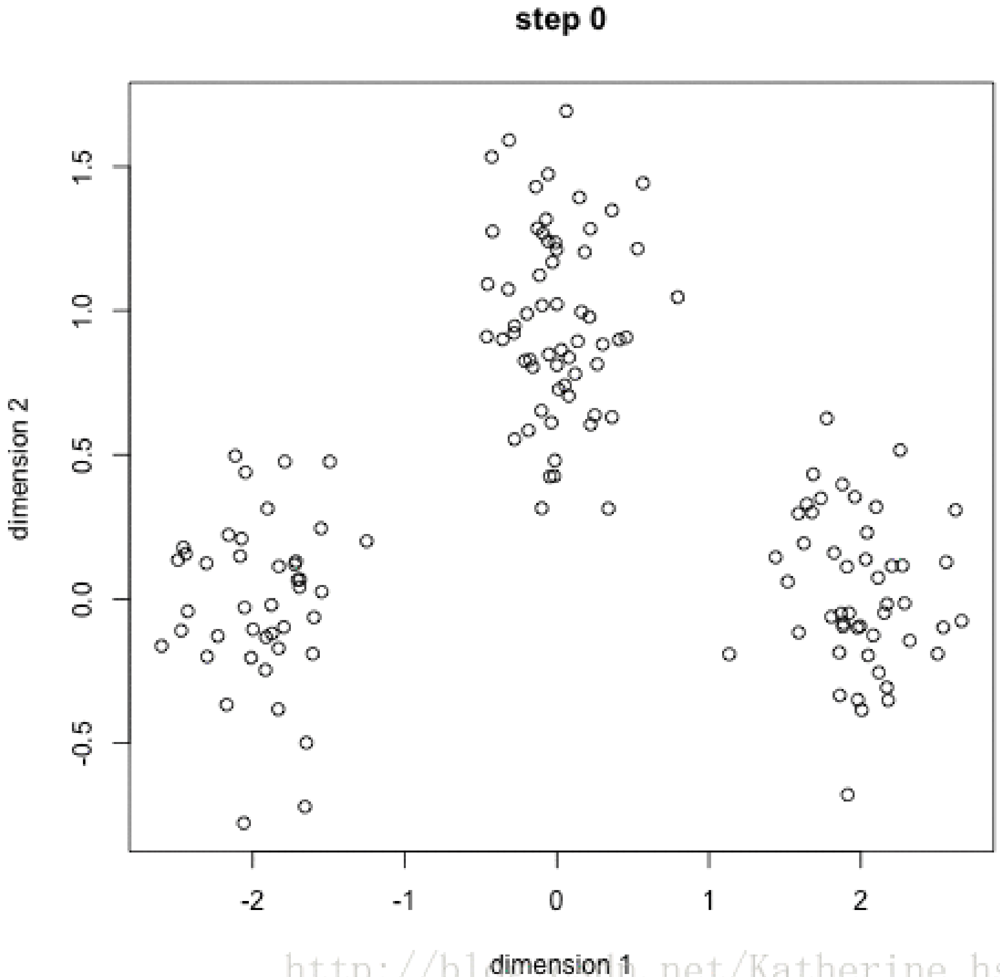
- ## Random Forest



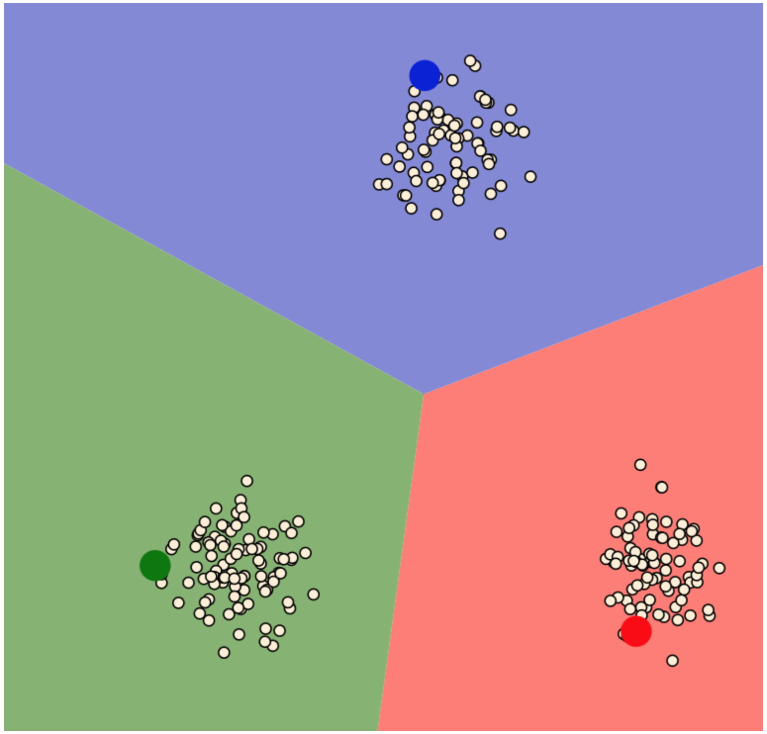
Random forest can limit the instability by the average predicted value of multiple trees

K-Means

K-Means --- clustering



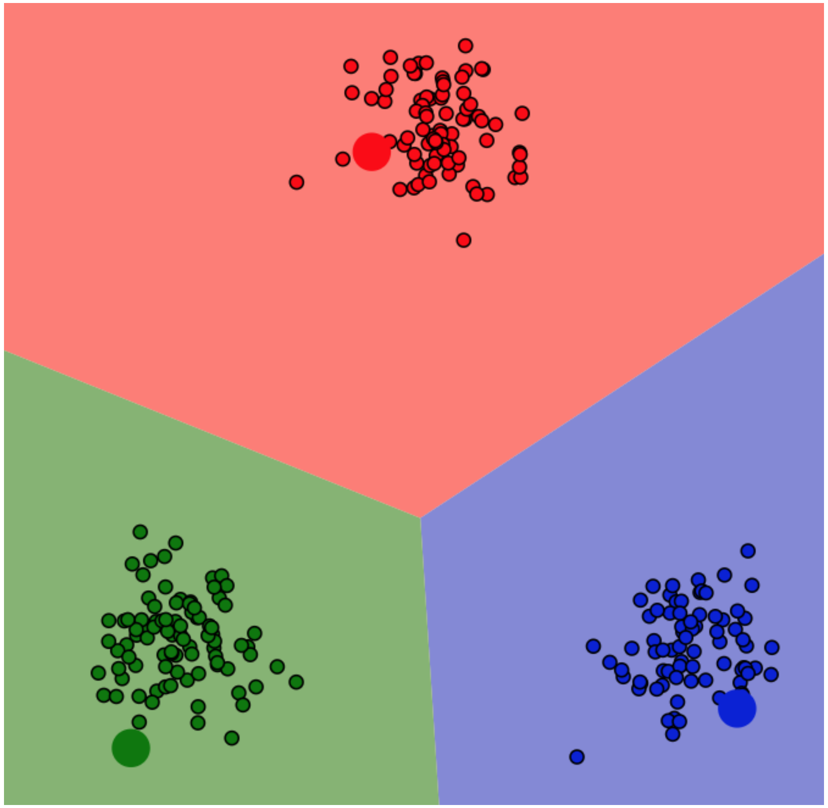
K-Means --- process



1. First enter the value of k, that is, we specify that we want to get k groups through clustering.

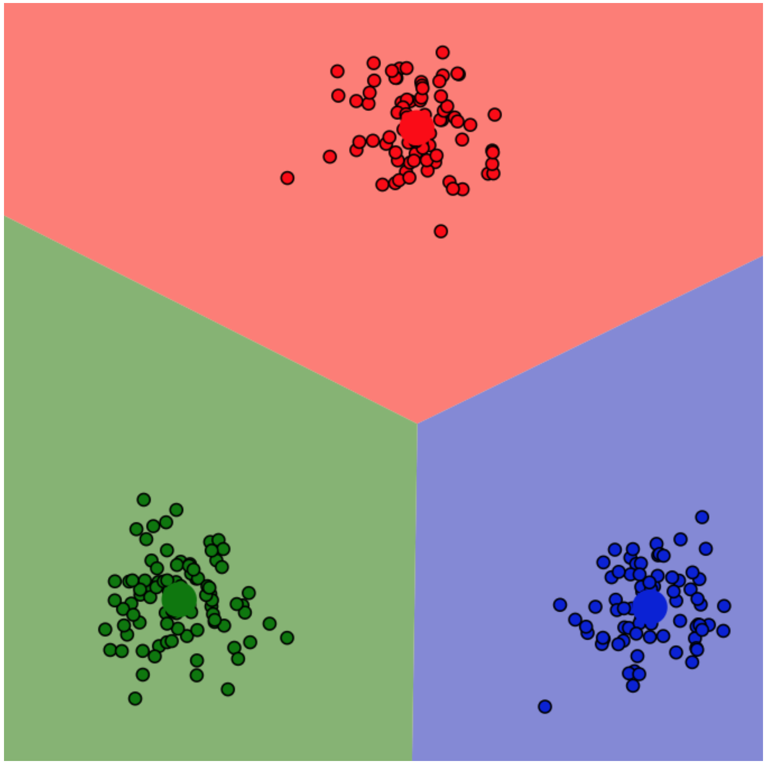
2. Randomly select k data points from the data set as the initial centroid.

K-Means --- process



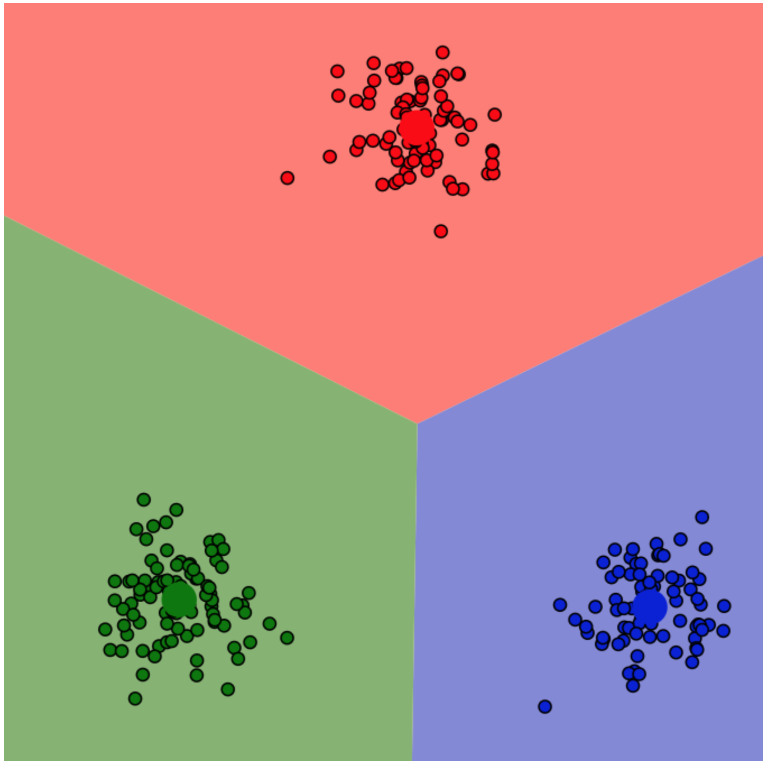
3. For each point in the set, calculate the distance to each centroid, and decide which centroid is the closest to this point. Then assign this group (represented by the centroid) to the point.

K-Means --- process



4. At this time, each of the centroids has a set of points associated with it. Then a new centroid was selected based on the distances between each points in the group to its centroid.

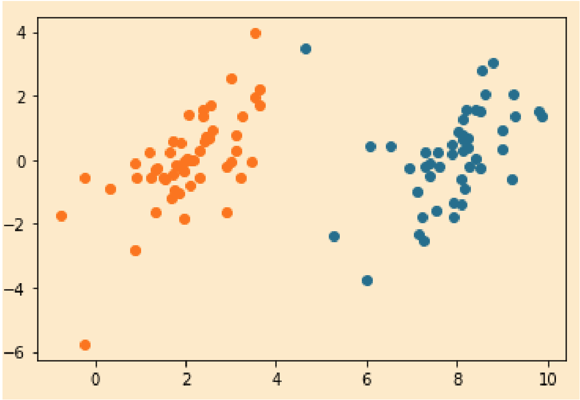
K-Means --- process



5. If the distance between the new boss and the old boss is less than a certain set threshold (indicating that the position of the recalculated centroid has not changed much, tends to stabilize, or converges), we can consider that the clustering we have performed has reached the desired result , The algorithm terminates.

If the distance between the new boss and the boss changes greatly, iterate step 3~5.

K-Means --- Implementation

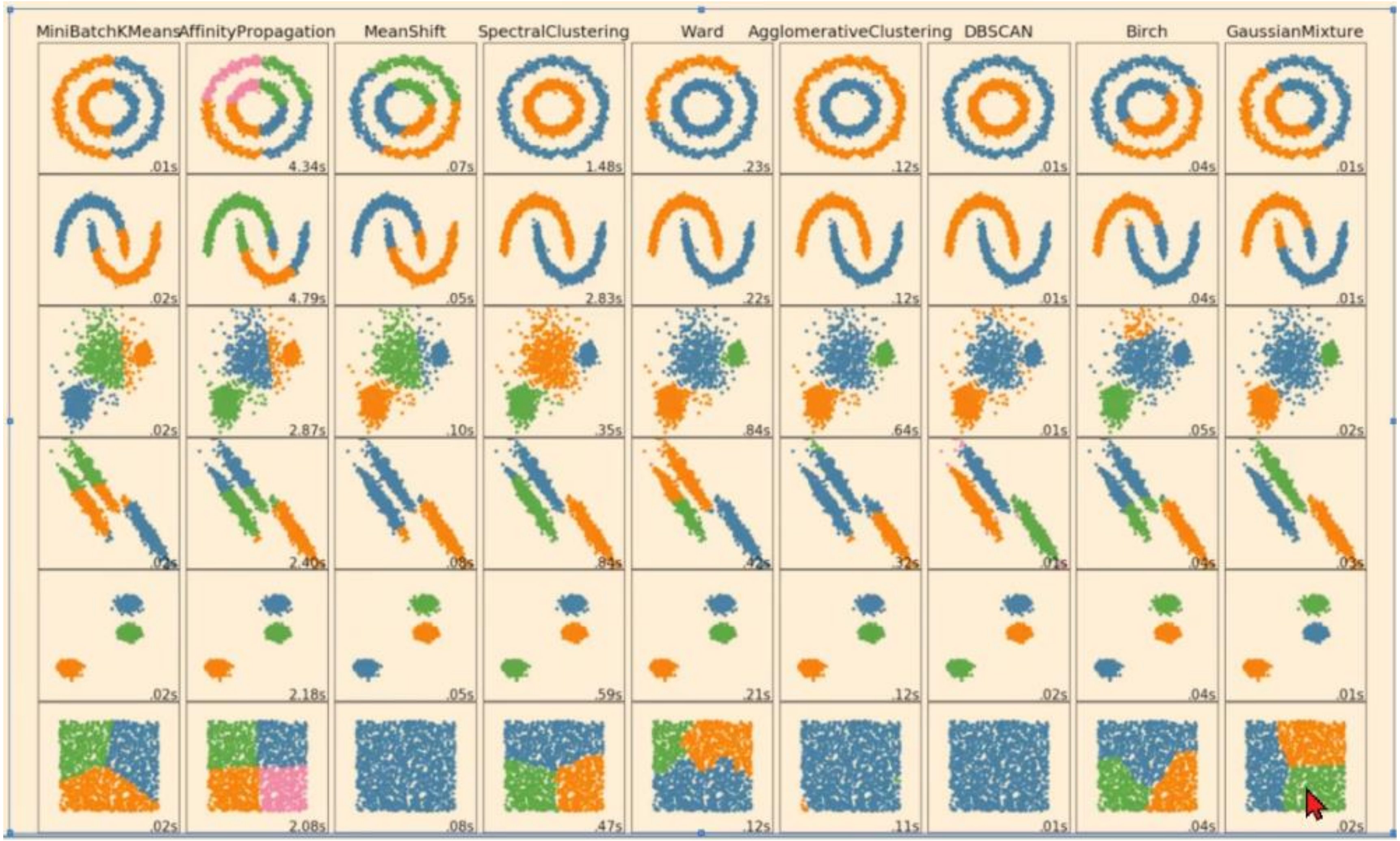



```
epoch = 5
for _ in range(epoch):
    for i in range(k):
        clusters[i]=[]

    # Calculate the distance from all points to the k cluster centers
    for i in range(x.shape[0]):
        xi = x[i]
        distances = np.sum((cluster_center-xi)**2,axis=1)
        # add the point to the cluster that is closer
        c = np.argmin(distances)
        clusters[c].append(i)

    # Recalculate the cluster centers of k clusters (all points in each cluster are added up and averaged)
    for i in range(k):
        cluster_center[i] = np.sum(x[clusters[i]],axis=0)/len(clusters[i])
```

Demo

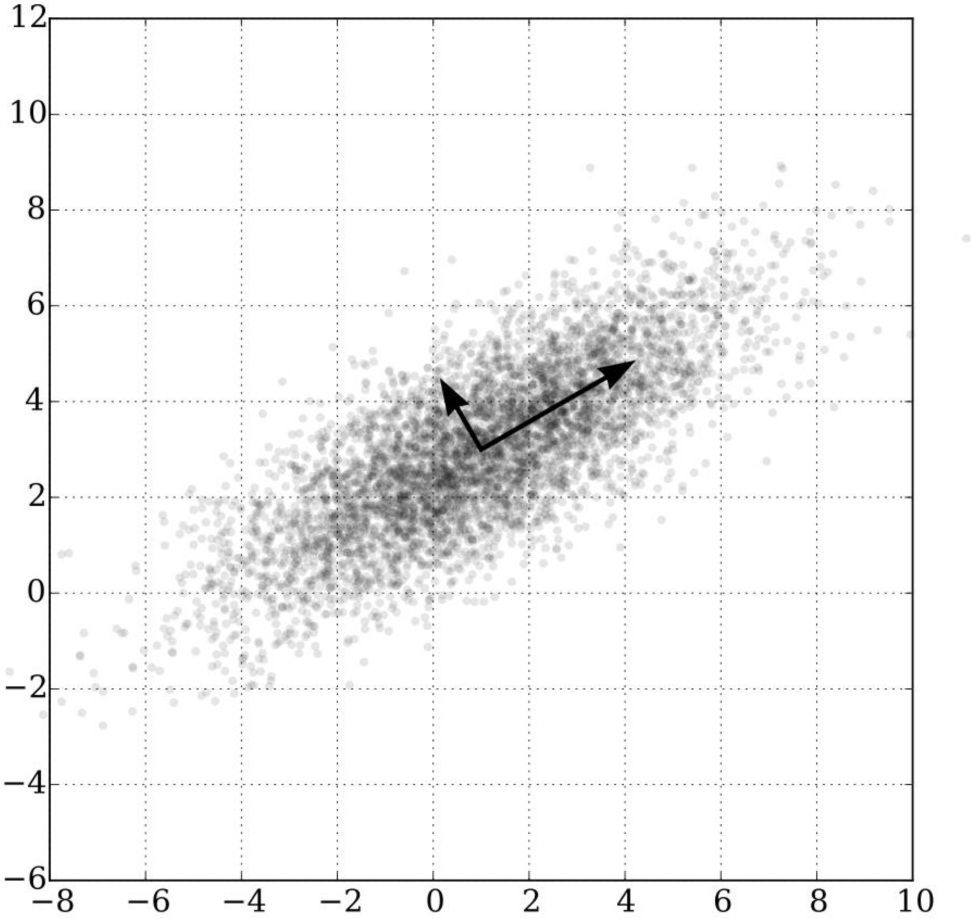
https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

Other Clustering Algorithms

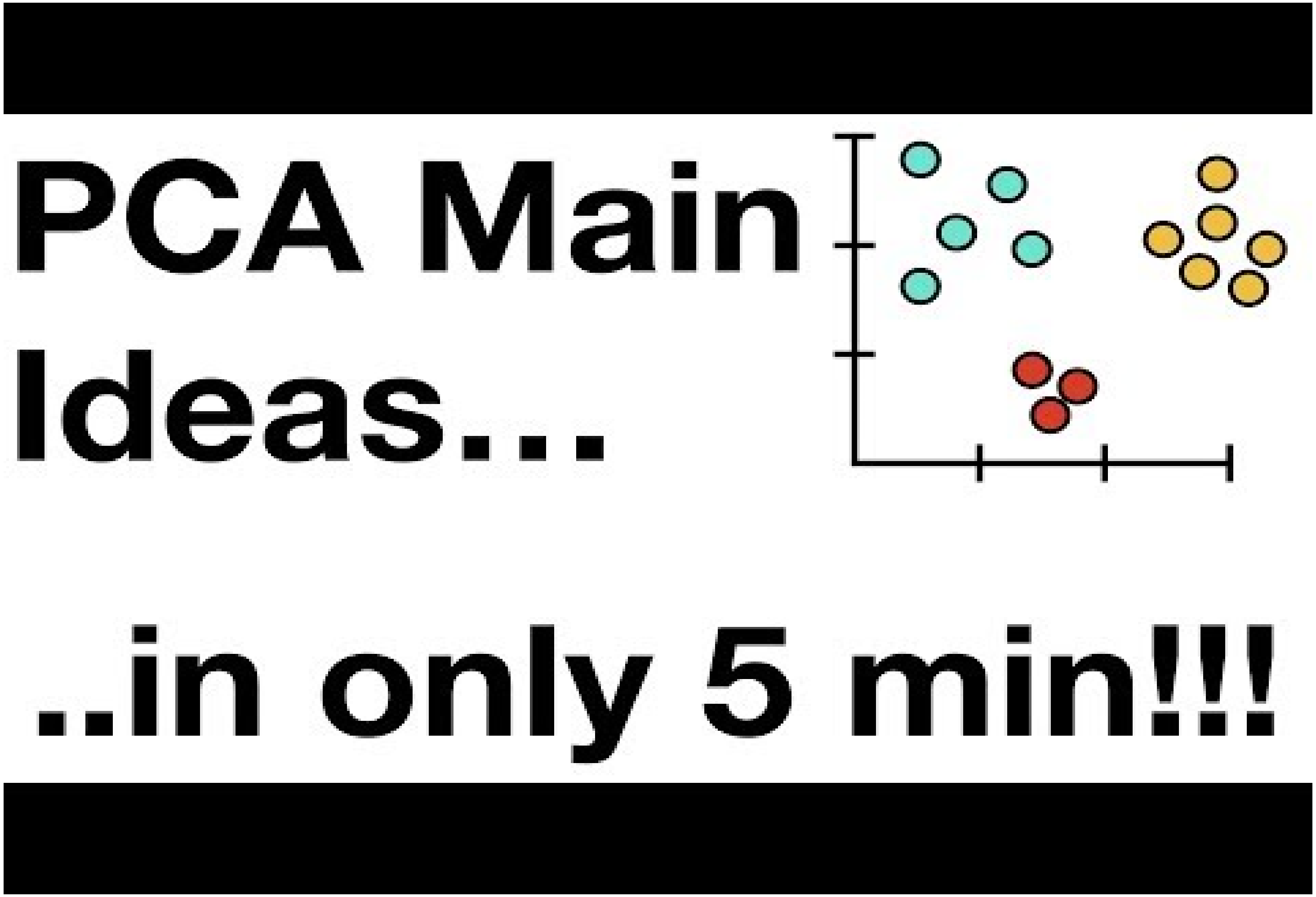


PCA --- Principal component analysis

PCA --- intuition



PCA --- Example



https://www.youtube.com/watch?v=HMOI_lkzW08

PCA --- Why Dimension Reduction

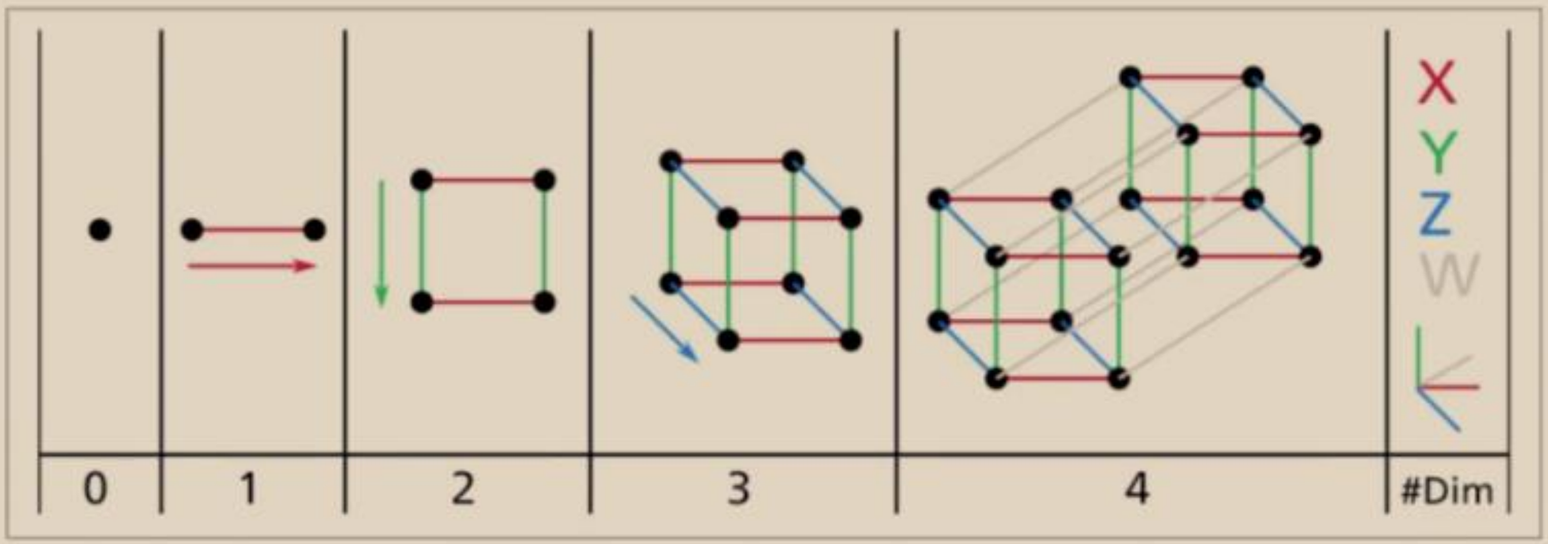
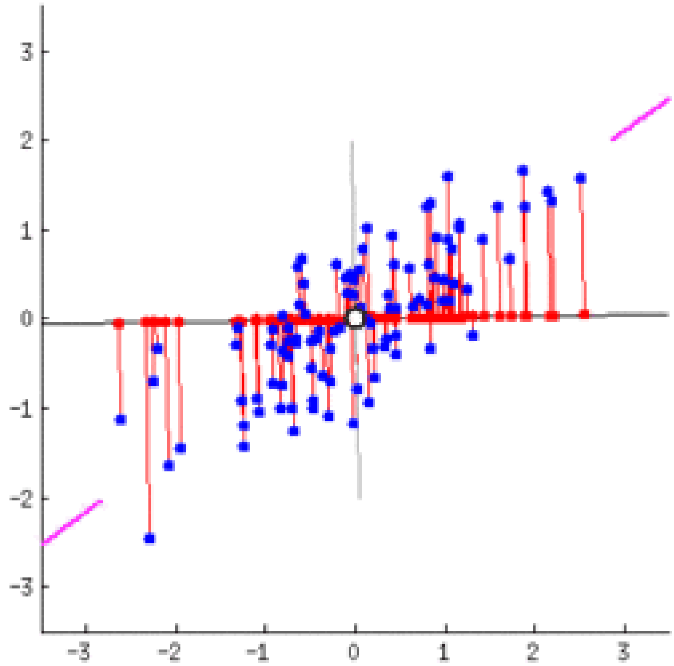


Figure 8-1. Point, segment, square, cube, and tesseract (0D to 4D hypercubes)[2]

PCA --- Process

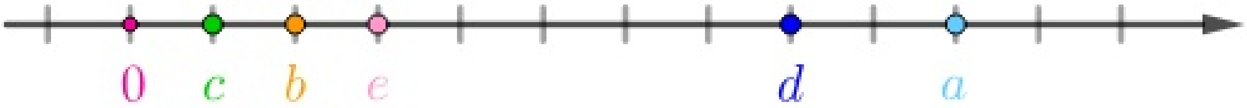


PCA --- Example

	House Price (in million)
a	10
b	2
c	1
d	7
e	3

PCA --- Example

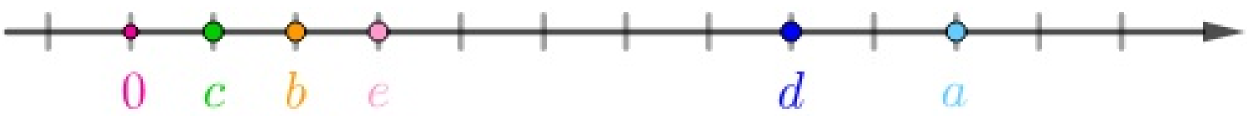
	House Price (in million)
a	10
b	2
c	1
d	7
e	3



If plotted on x axis...

PCA --- Example

	House Price (in million)
a	10
b	2
c	1
d	7
e	3

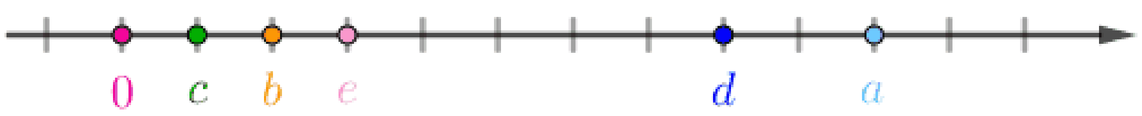


$$\overline{X} = \frac{X_1 + X_2 + X_3 + X_4 + X_5}{5} = \frac{10+2+1+7+3}{5} = 4.6$$

The mean is ...

PCA --- Example

	House Price (in million)
a	10
b	2
c	1
d	7
e	3



$$\overline{X} = \frac{X_1 + X_2 + X_3 + X_4 + X_5}{5} = \frac{10+2+1+7+3}{5} = 4.6$$

Let mean point be the origin point...

PCA --- Example Centralization

	House Price (in million)
a	10
b	2
c	1
d	7
e	3

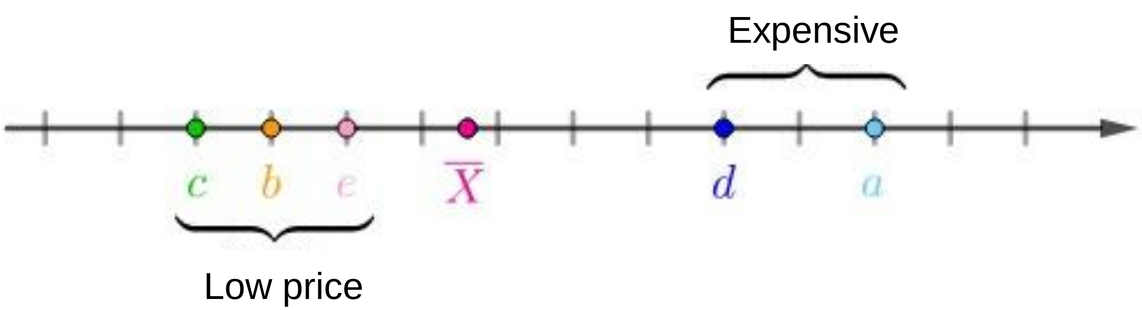
	House Price (in million)
a	$10 - \overline{X} = 5.4$
b	$2 - \overline{X} = -2.6$
c	$1 - \overline{X} = -3.6$
d	$7 - \overline{X} = 2.4$
e	$3 - \overline{X} = -1.6$

So the table is changed to ...

PCA --- Example Centralization

	House Price (in million)
a	$10 - \overline{X} = 5.4$
b	$2 - \overline{X} = -2.6$
c	$1 - \overline{X} = -3.6$
d	$7 - \overline{X} = 2.4$
e	$3 - \overline{X} = -1.6$

After centralization ...



PCA --- Example Sample variance

	House Price (in million)
a	$10 - \overline{X} = 5.4$
b	$2 - \overline{X} = -2.6$
c	$1 - \overline{X} = -3.6$
d	$7 - \overline{X} = 2.4$
e	$3 - \overline{X} = -1.6$

$$Var(X) = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

So the sample variance is ...

$$Var(X) = \frac{1}{n}\left(5.4^2 + (-2.6)^2 + (-3.6)^2 + 2.4^2 + (-1.6)^2\right)$$

PCA --- Example2

	House Price (in million)	House Space (in m^2)
a	10	10
b	2	2
c	1	1
d	7	7
e	3	3

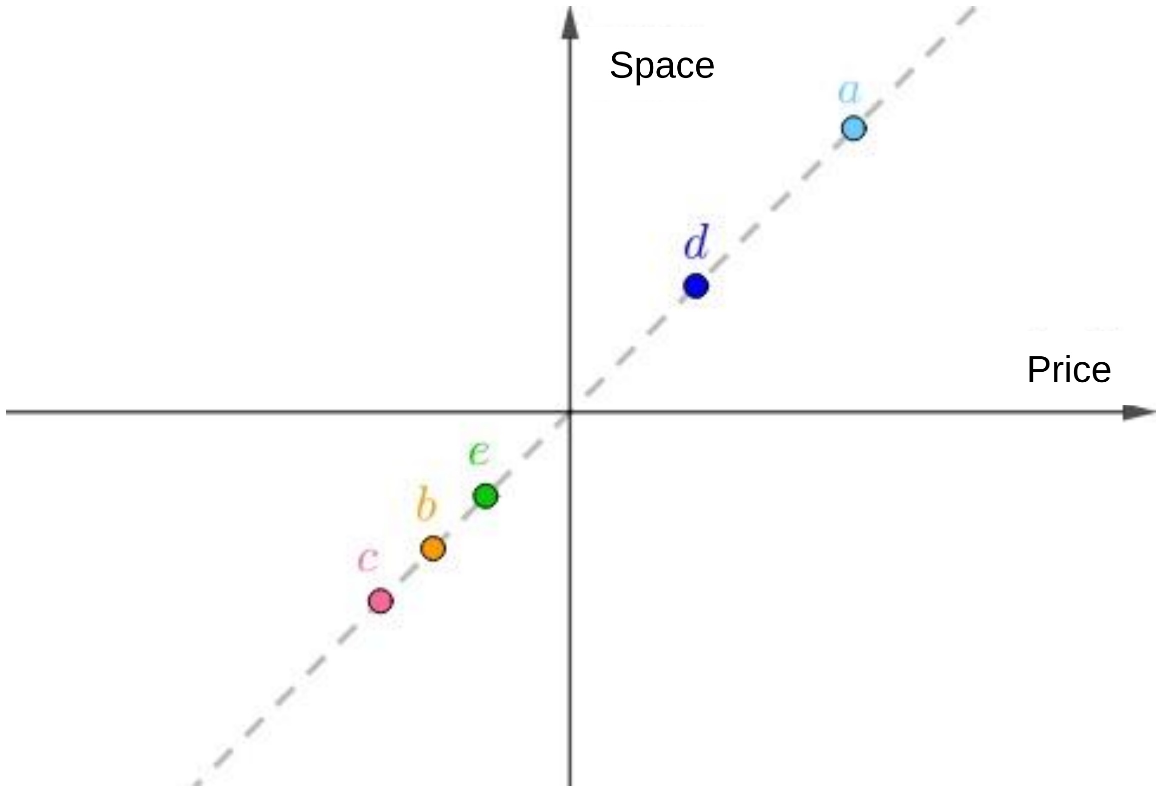
PCA --- Example2 Centralization

	House Price (in million)	House Space (in m^2)
a	10	10
b	2	2
c	1	1
d	7	7
e	3	3

	House Price (in million)	House Space (in m^2)
a	5.4	5.4
b	-2.6	-2.6
c	-3.6	-3.6
d	2.4	2.4
e	-1.6	-1.6

PCA --- Example2 Centralization

After centralization ...



PCA --- Example2 Sample covariance

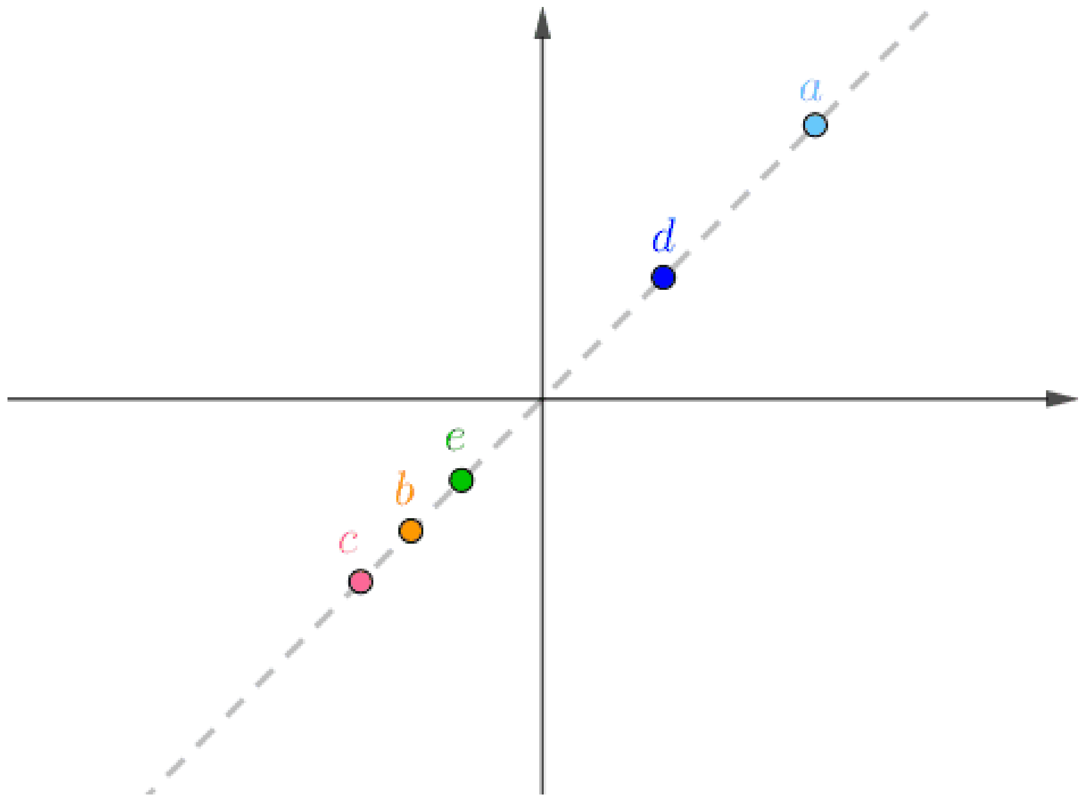
	House Price (in million)	House Space (in m^2)
a	5.4	5.4
b	-2.6	-2.6
c	-3.6	-3.6
d	2.4	2.4
e	-1.6	-1.6

The sample covariance of house price (X) and area (Y) is ...

$$Cov(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})$$

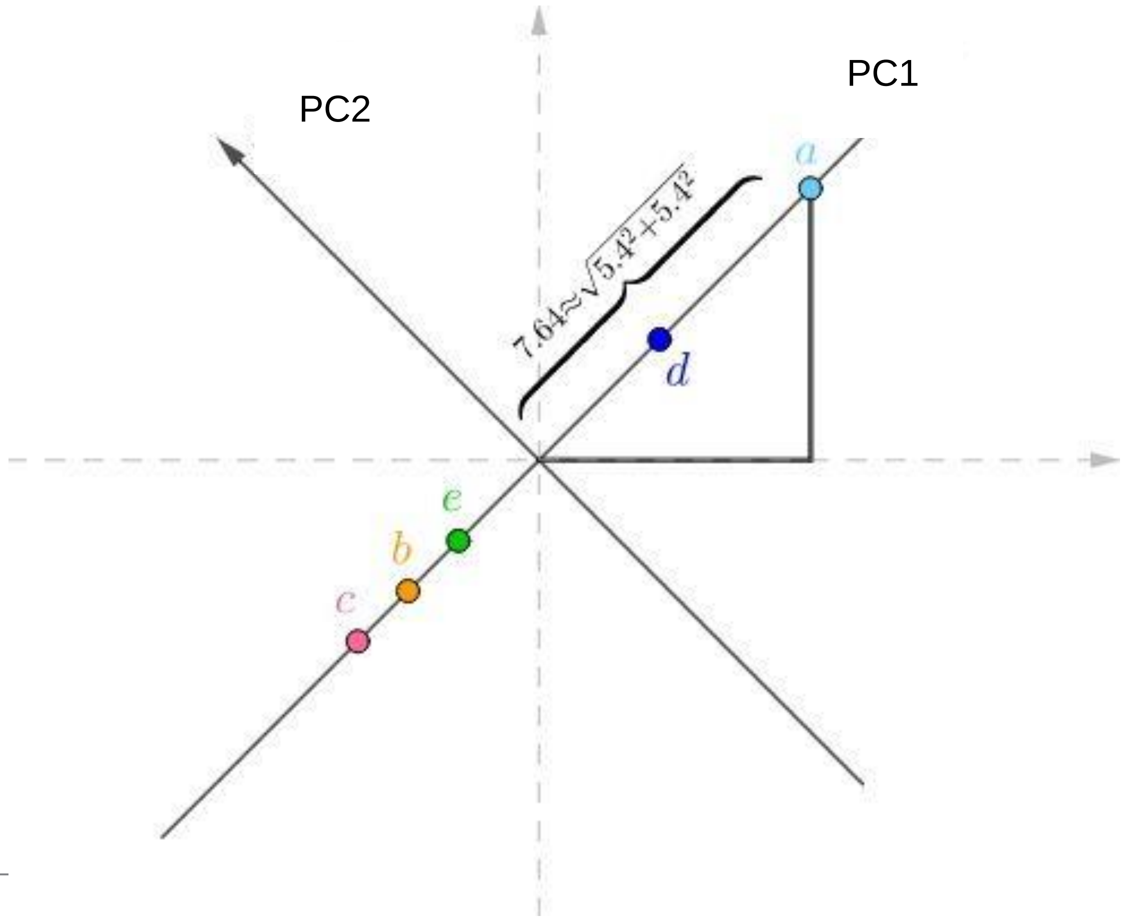
PCA --- Example2 Centralization

If we rotate the coordinate system...



PCA --- Example2 Principal component

In the rotated coordinate system, the horizontal and vertical coordinates no longer represent "house price" and "space", but a mixture of the two (the term is **linear combination**). Here they are called "principal component 1" and "principal component 2" , The coordinate value is easily calculated using the Pythagorean theorem:

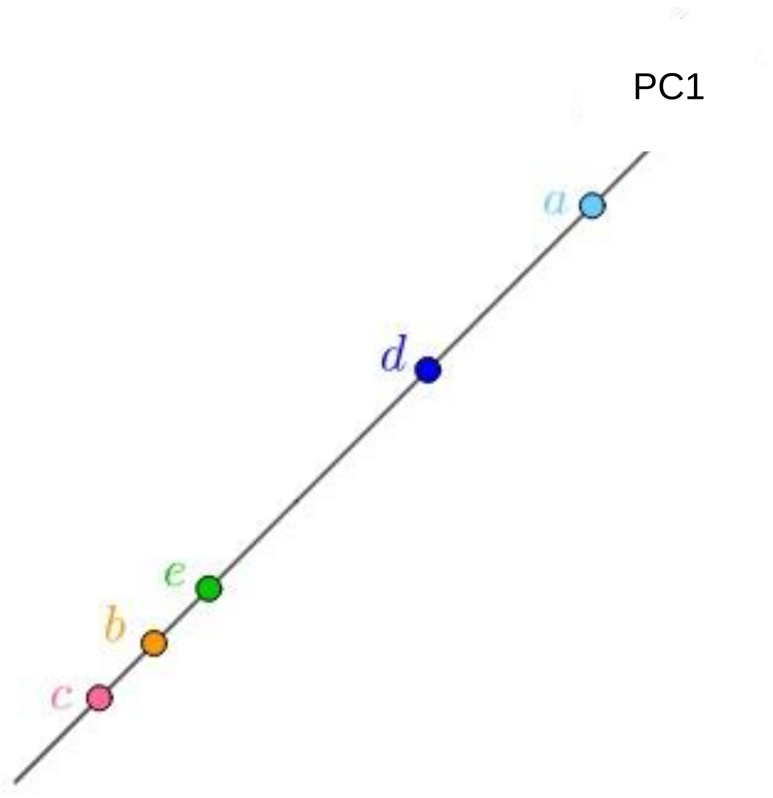


PCA --- Example2 Principal component

	PC1	PC2
a	7.64	0
b	-3.68	0
c	-5.09	0
d	3.39	0
e	-2.26	0

PCA --- Example2 Principal component

Because "PC 2" is all 0, it is completely redundant. We only need "PC 1" to reduce the data to one dimension without losing any information:

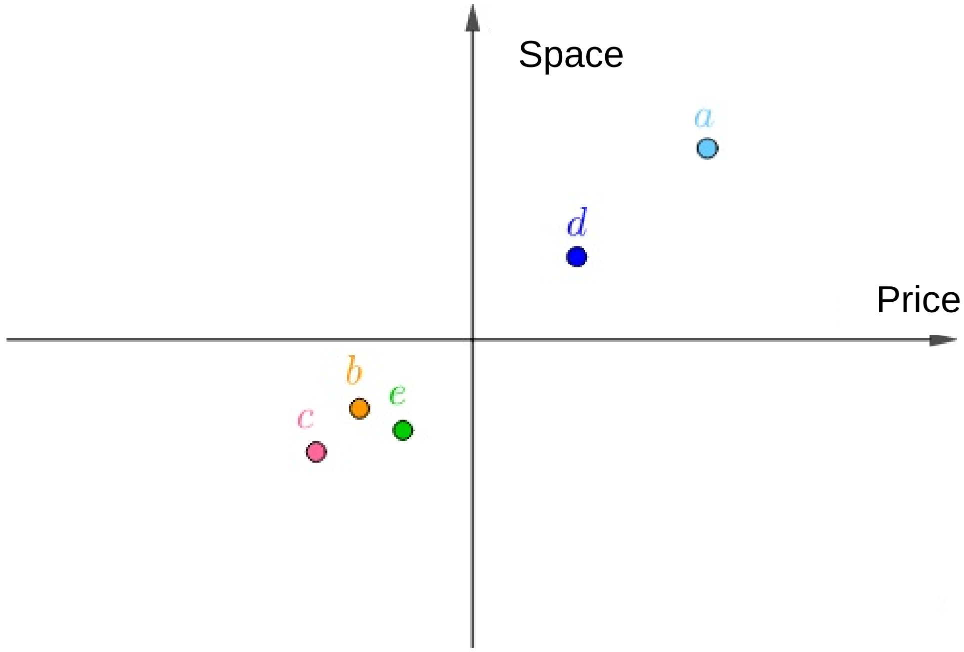


PCA --- Example2 Non-ideal case

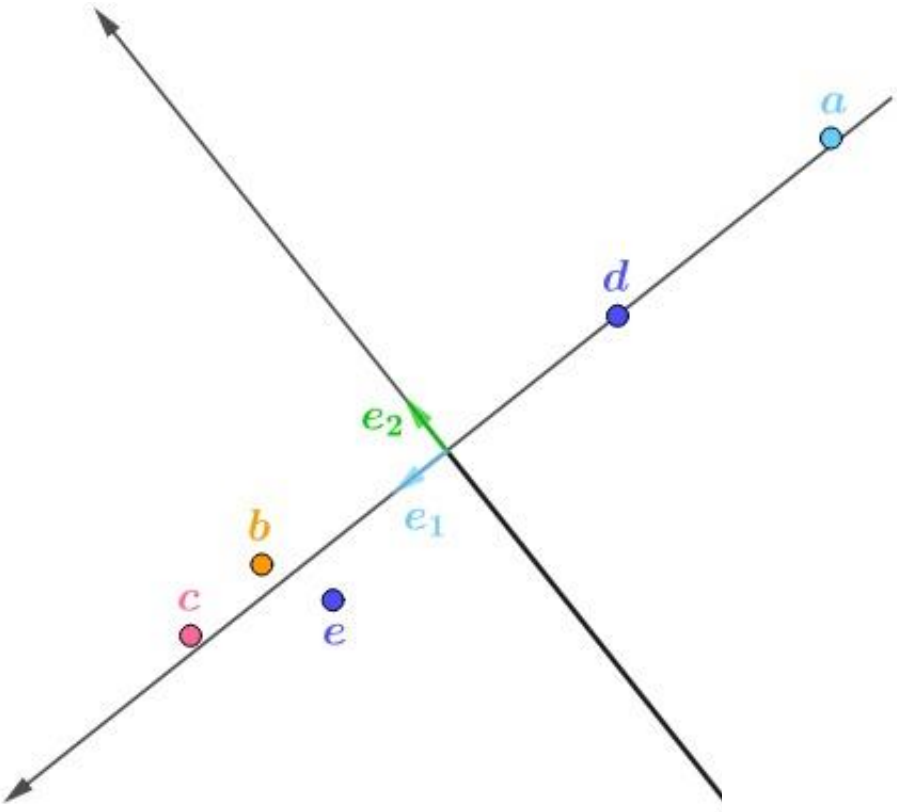
	Price	Space
a	10	9
b	2	3
c	1	2
d	7	6.5
e	3	2.5

$\longrightarrow$

	Price	Space
a	5.4	4.4
b	-2.6	-1.6
c	-3.6	-2.6
d	2.4	1.9
e	-1.6	-2.1



PCA --- Example2 Non-ideal case



PCA – Higher Dimensions

- Calculate the covariance matrix of the data for each pair of dimension values

- Calculate the eigenvalues and eigenvectors of the covariance matrix. The are the principle components of the data
  - an eigenvector is a non-zero vector which, when multiplied by a square matrix, yields a constant times the vector):

Matrix we are finding the eigenvector/eigenvalue of

eigenvalue

$$Ax = \lambda x$$

Demo

Reference

- PCA Wikipidia

https://en.wikipedia.org/wiki/Principal_component_analysis