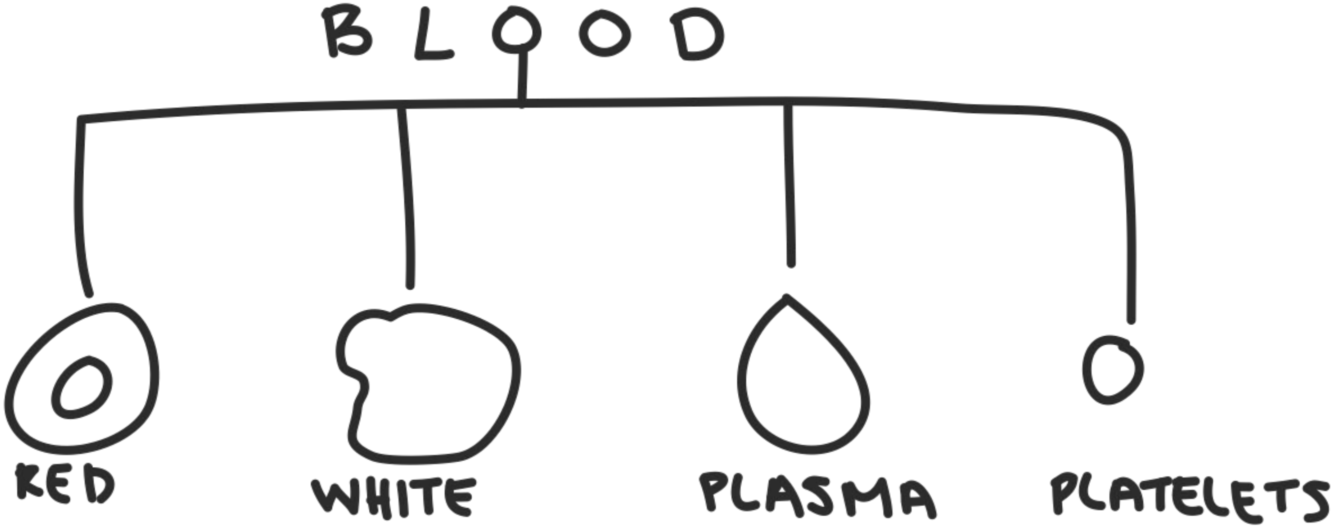
Hierarchical Clustering

Agglomerative
(Bottom up)

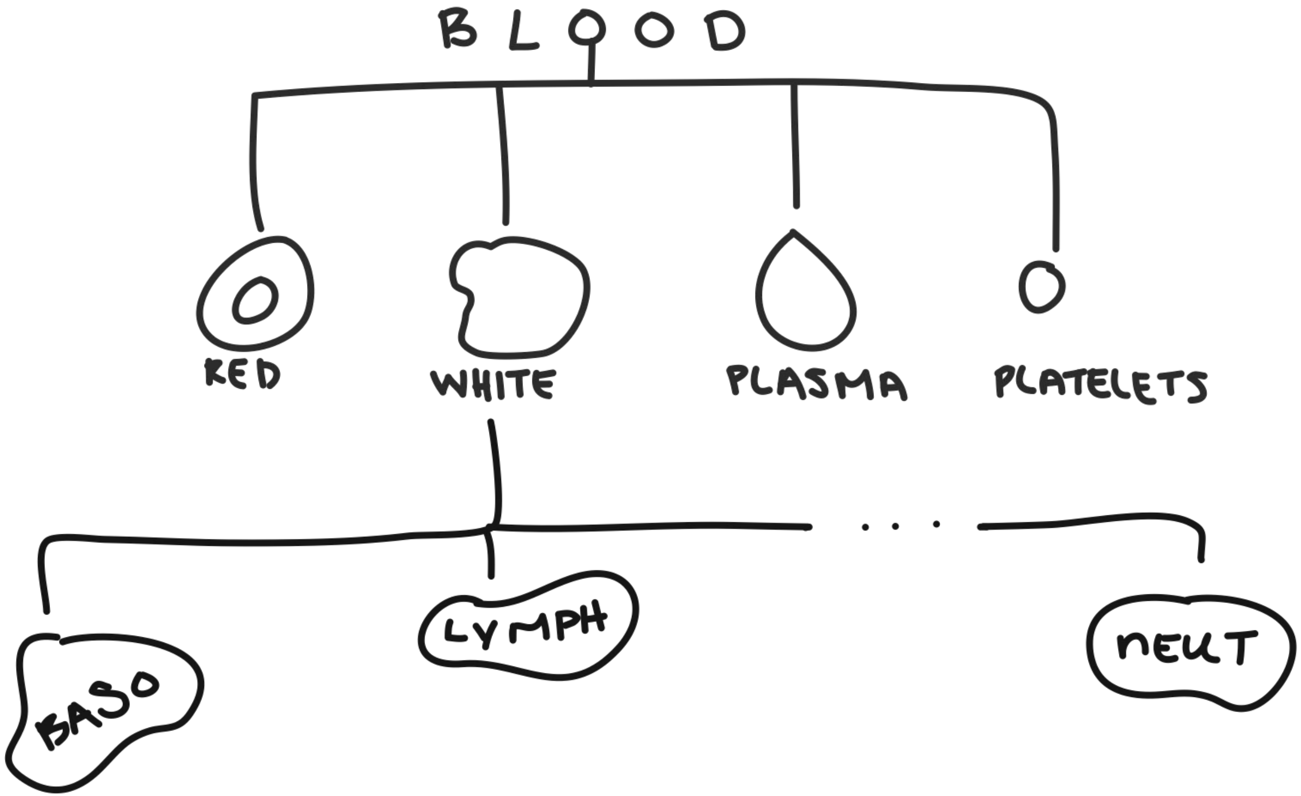
Divisive
(Top Down)

HAC

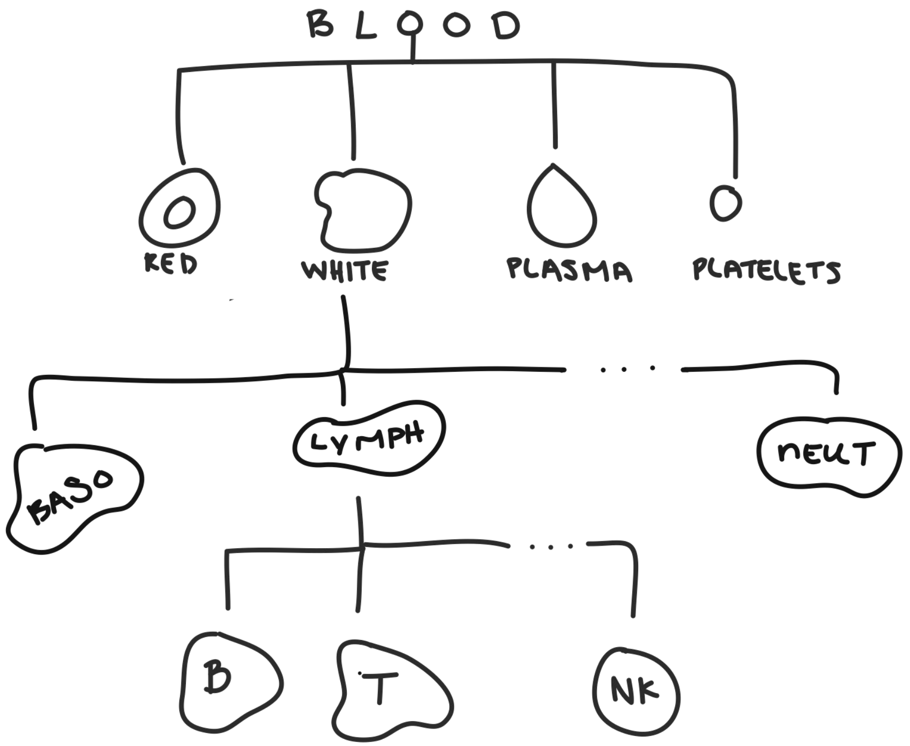
Hierarchical
Agg. (BO)
Clustering



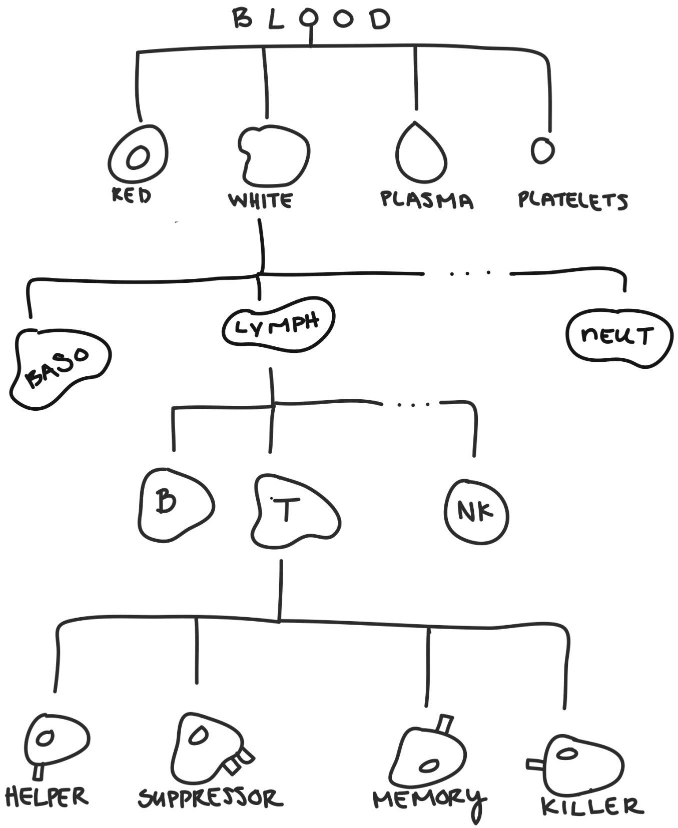
HAC



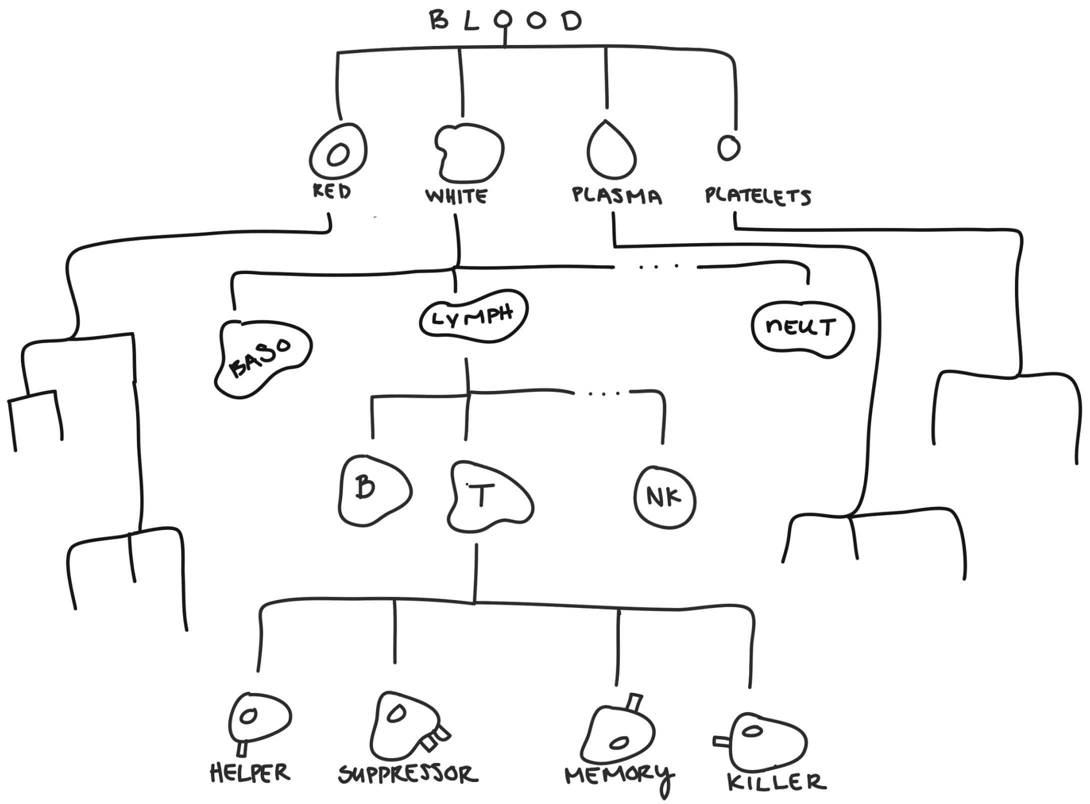
HAC



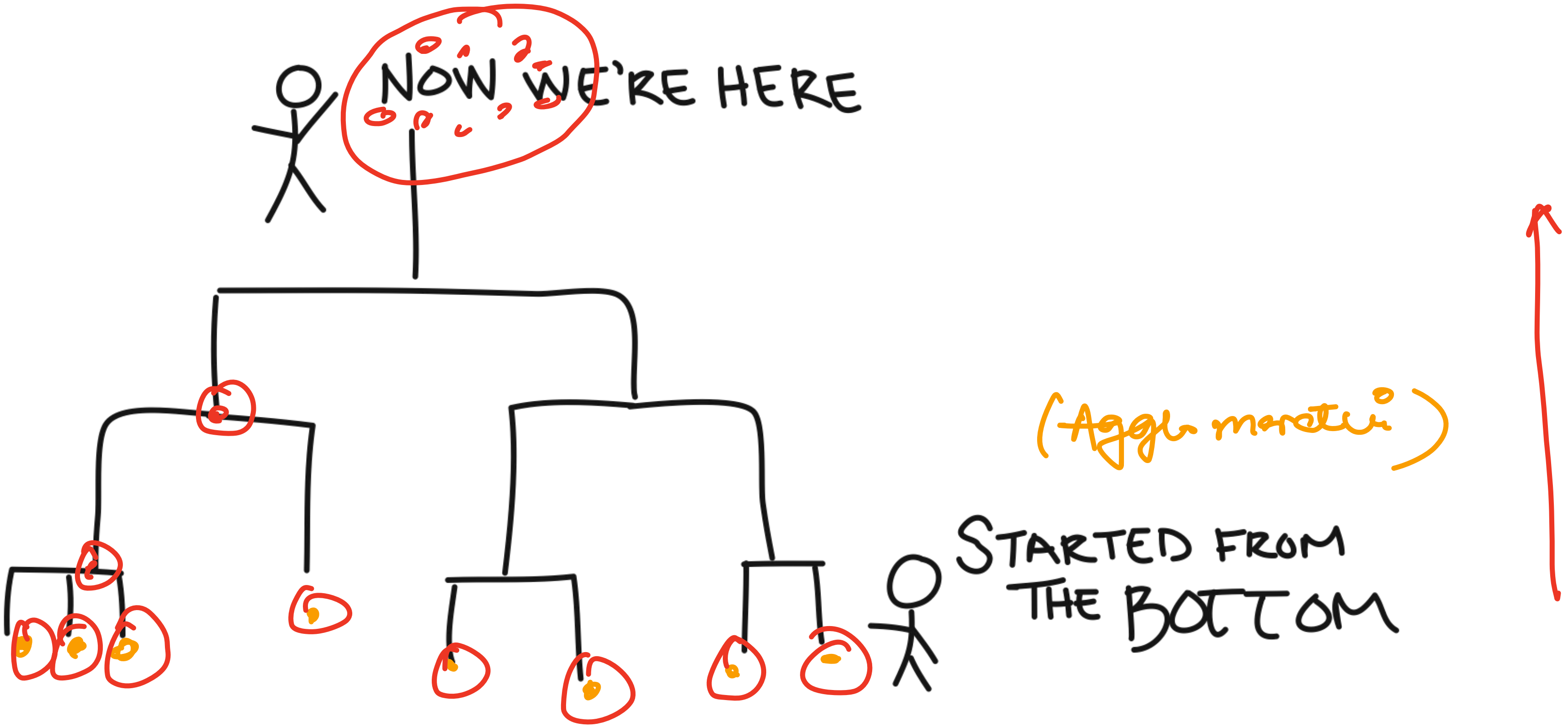
HAC


BLOOD
RED
WHITE
PLASMA
PLATELETS
BASO
LYMPH
NEUT
B
T
NK
HELPER
SUPPRESSOR
MEMORY
KILLER

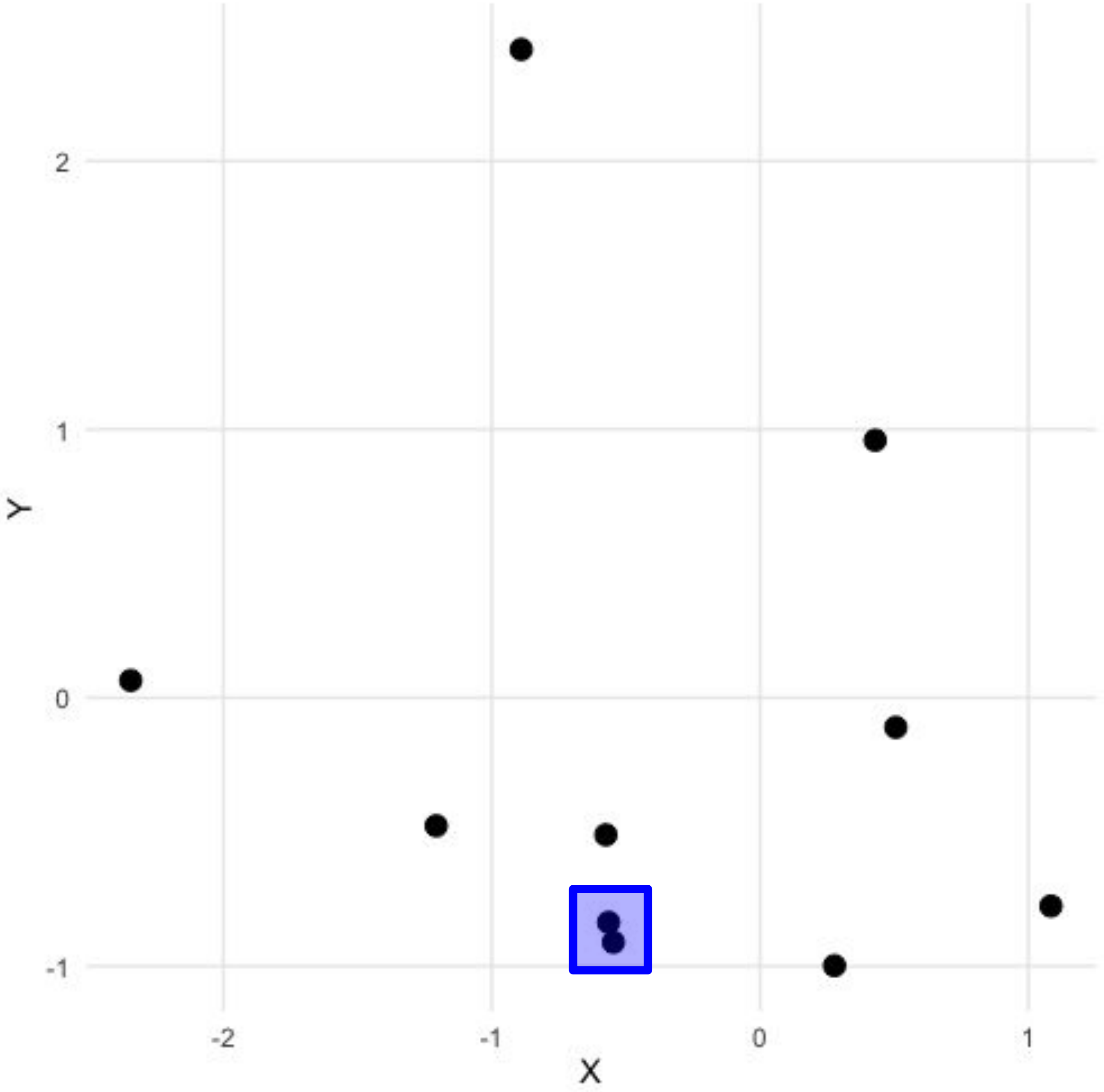
HAC


BLOOD
RED
WHITE
PLASMA
PLATELETS
BASO
LYMPH
NEUT
B
T
NK
HELPER
SUPPRESSOR
MEMORY
KILLER

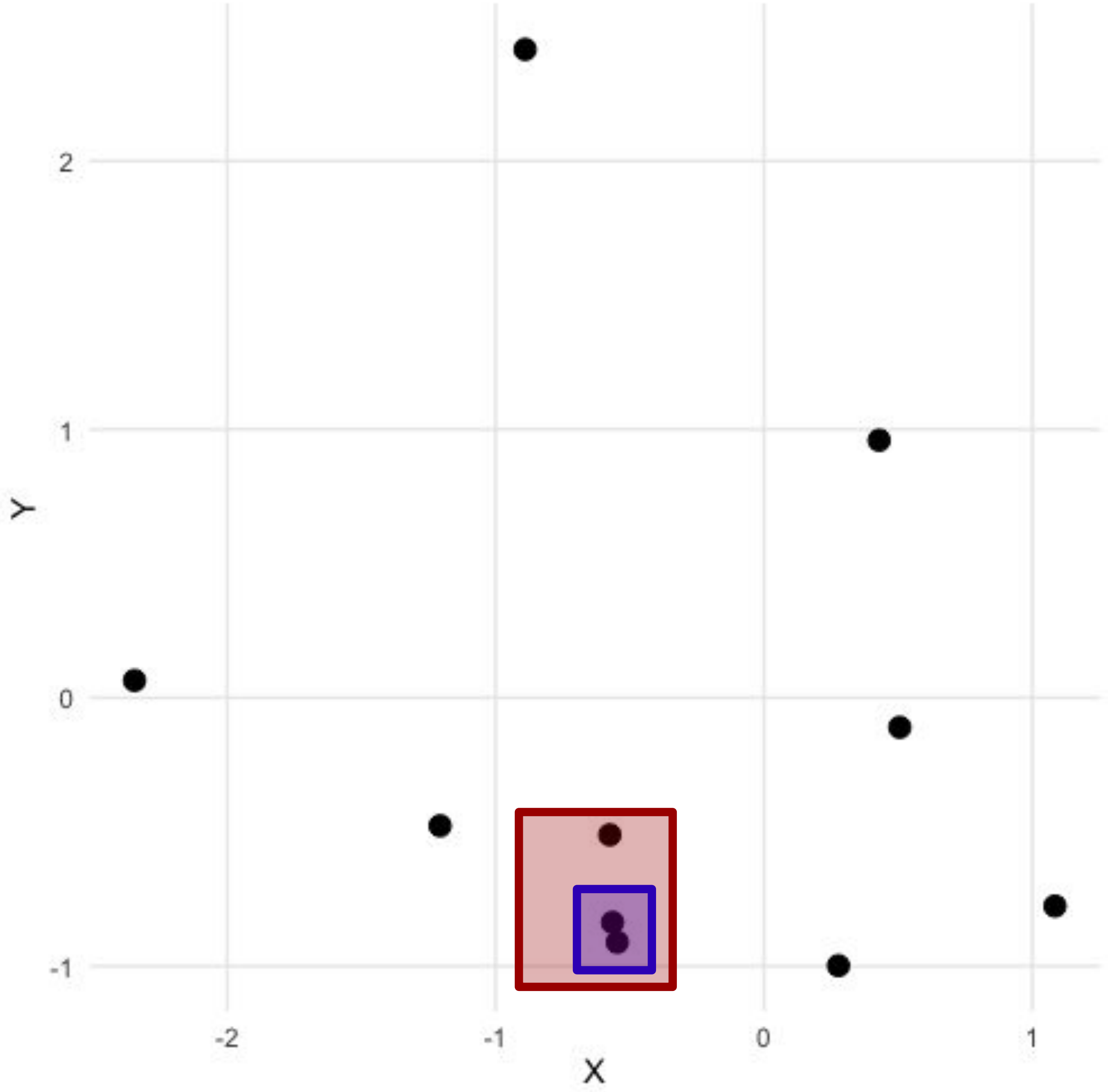
HAC



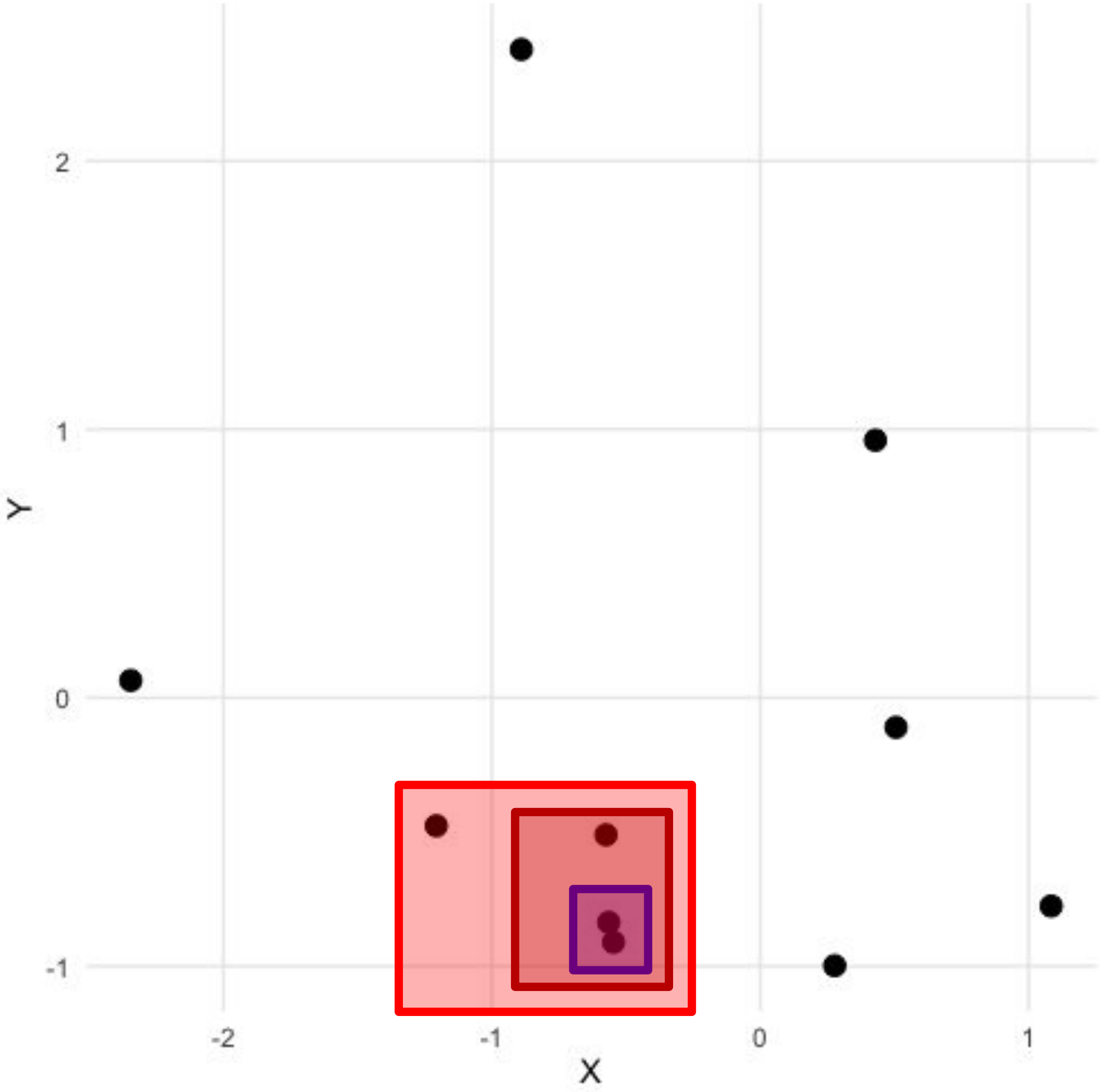
Algorithm



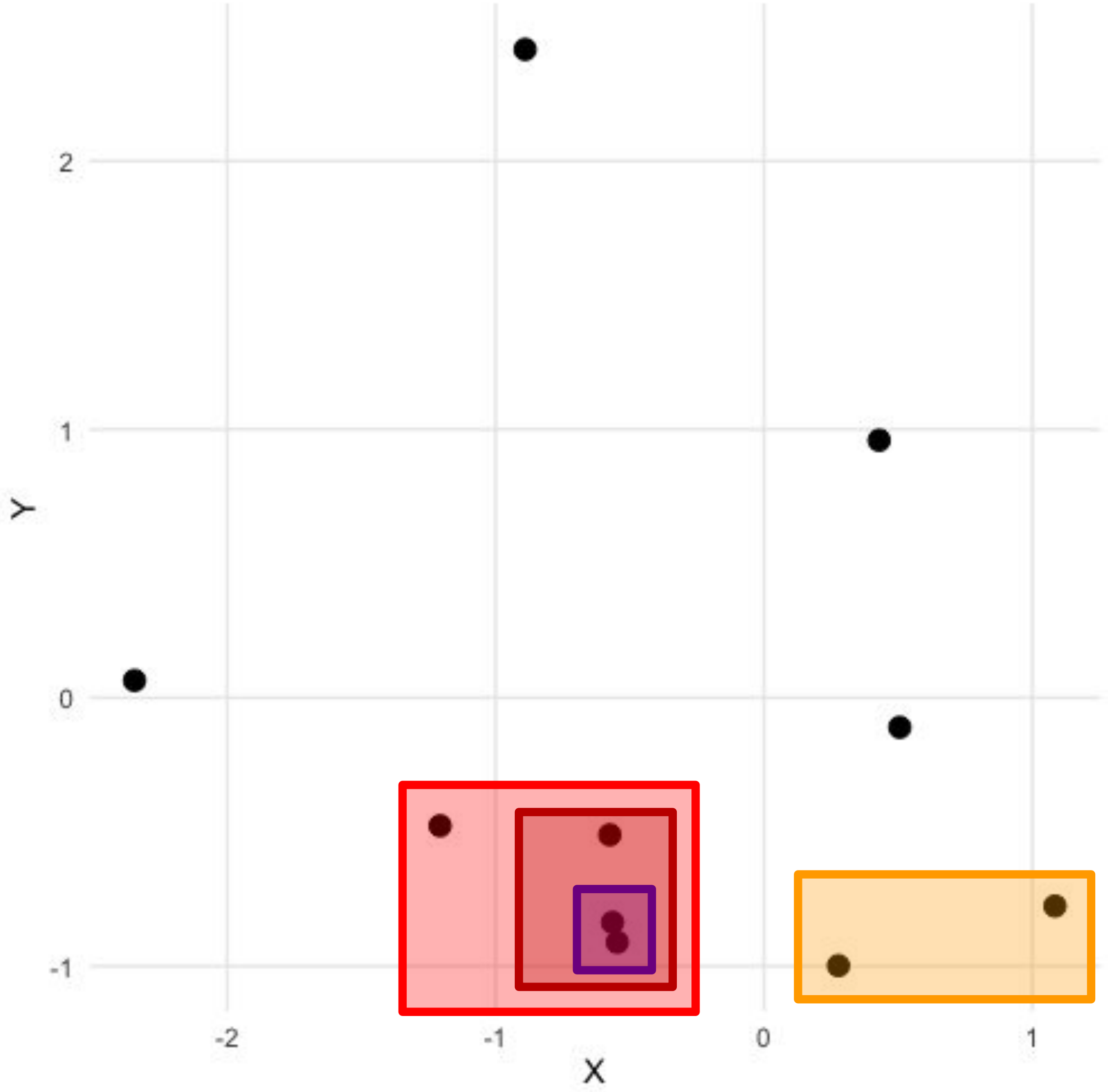
Algorithm



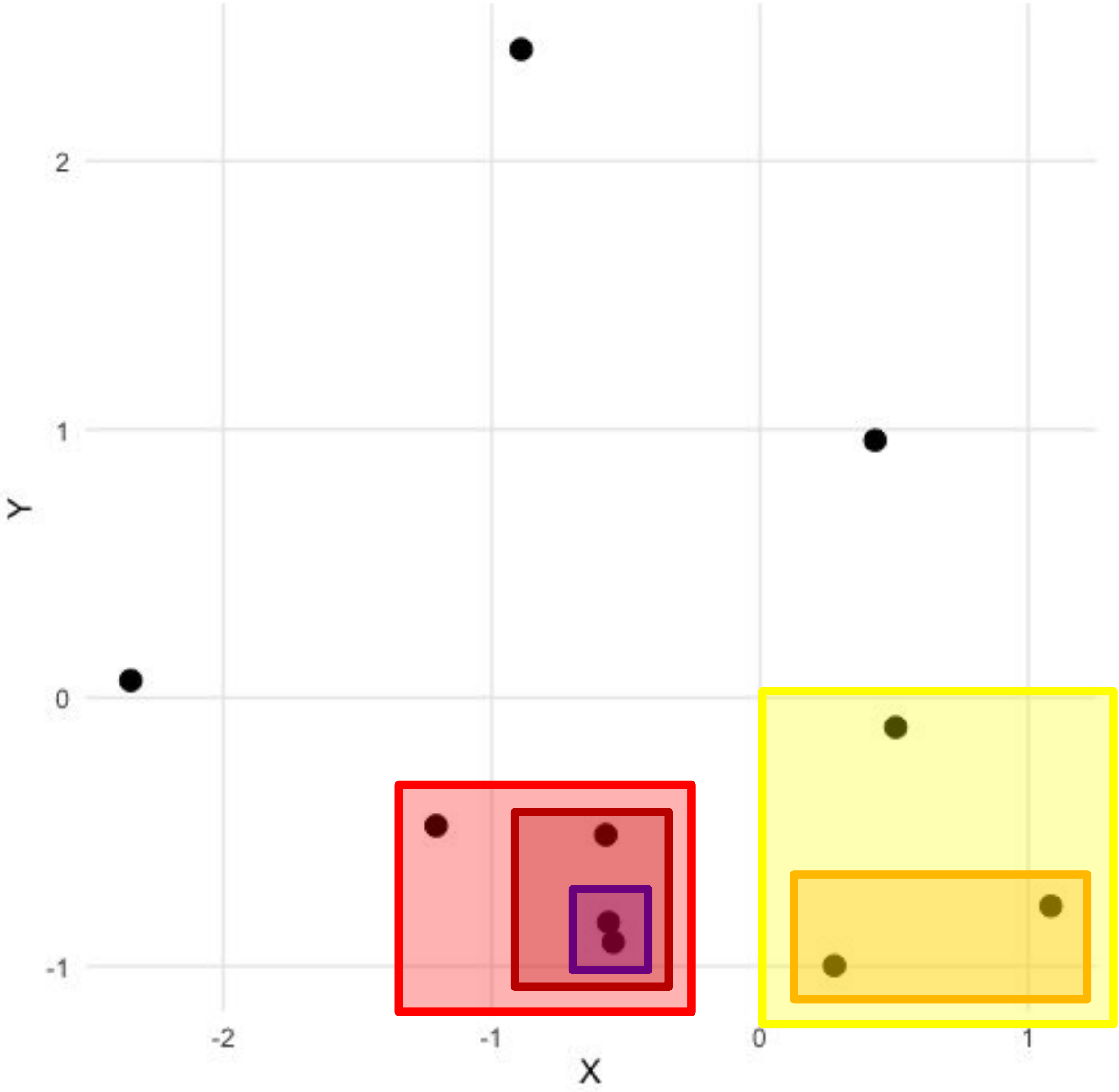
Algorithm



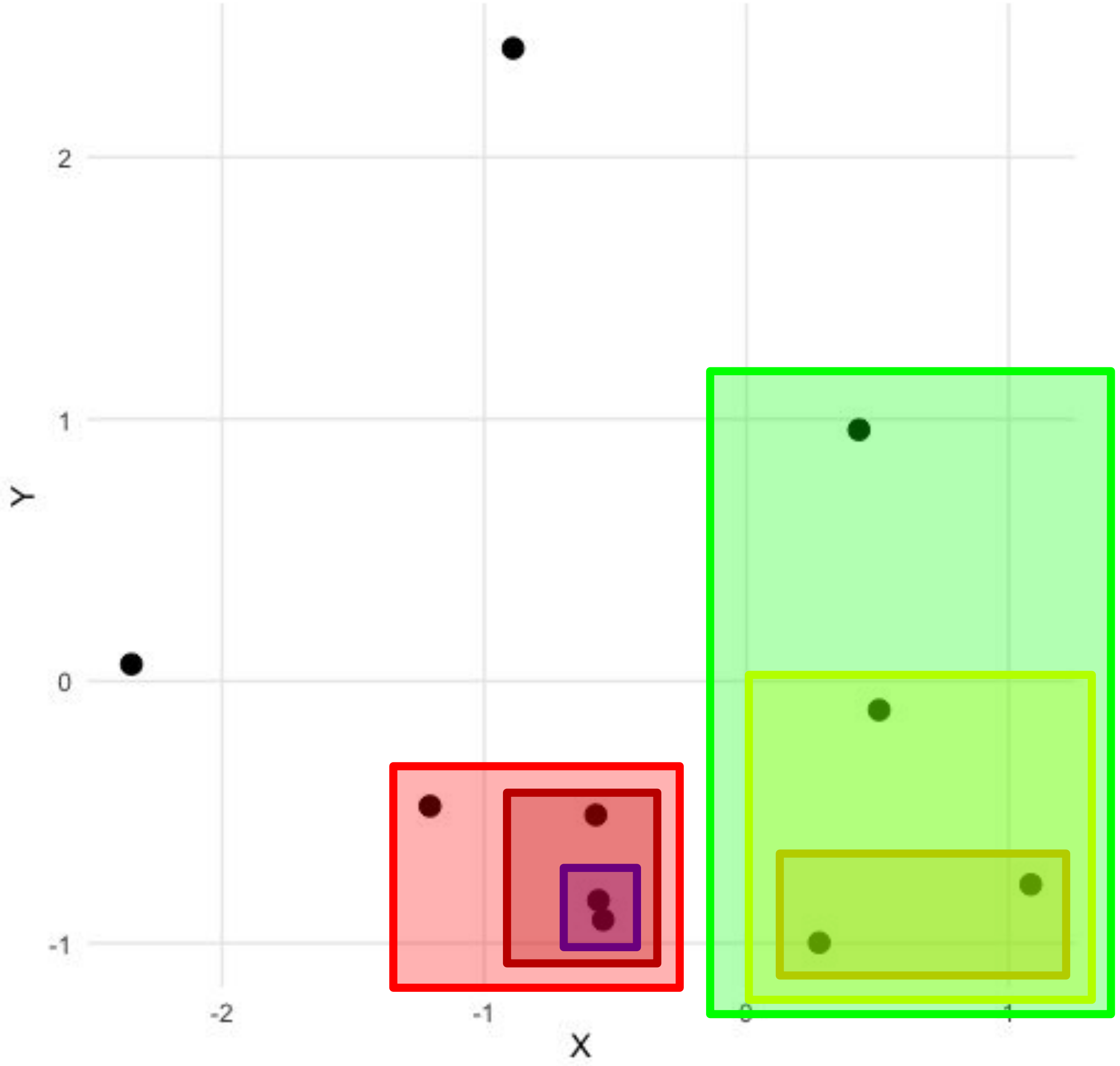
Algorithm



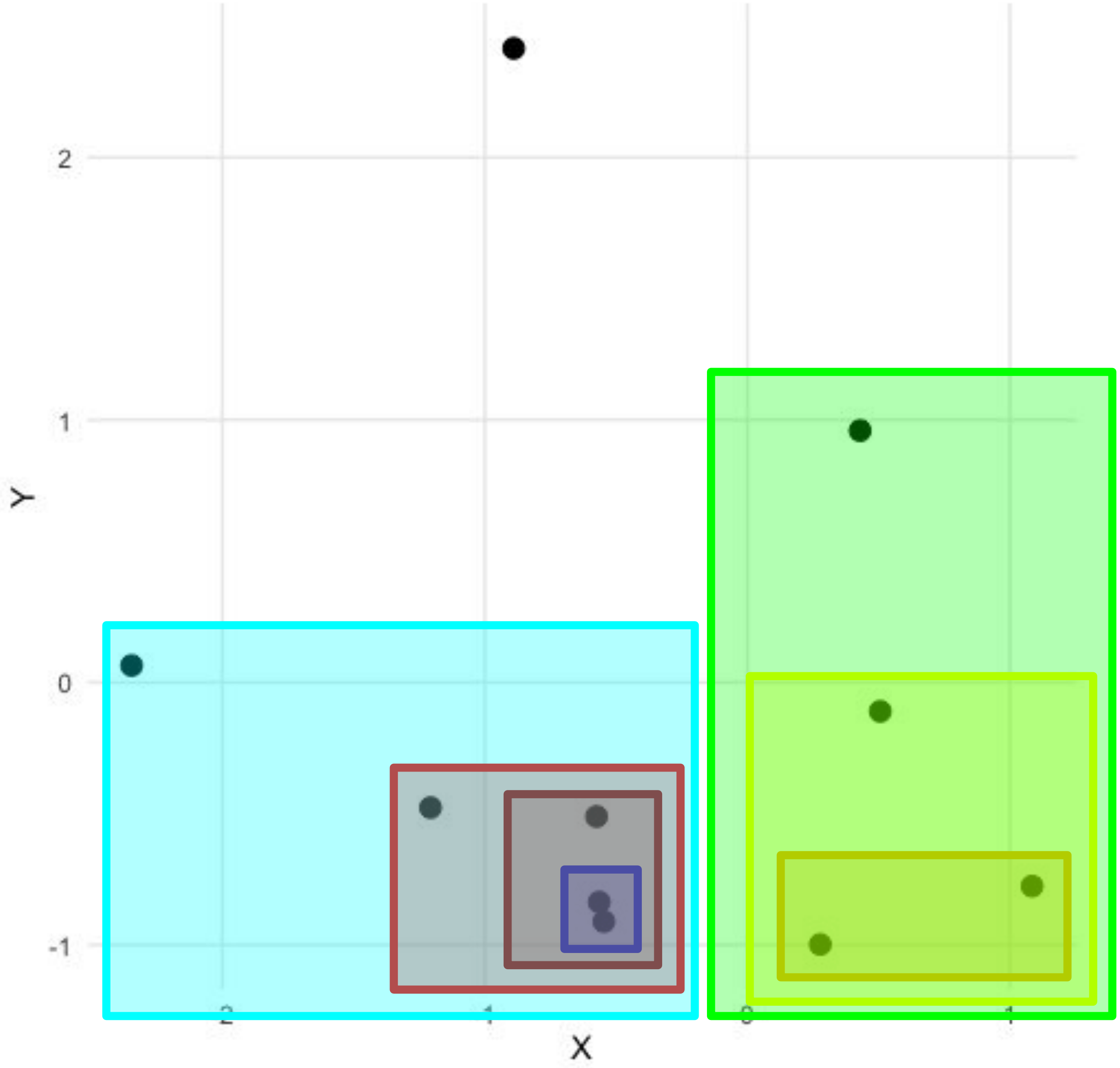
Algorithm



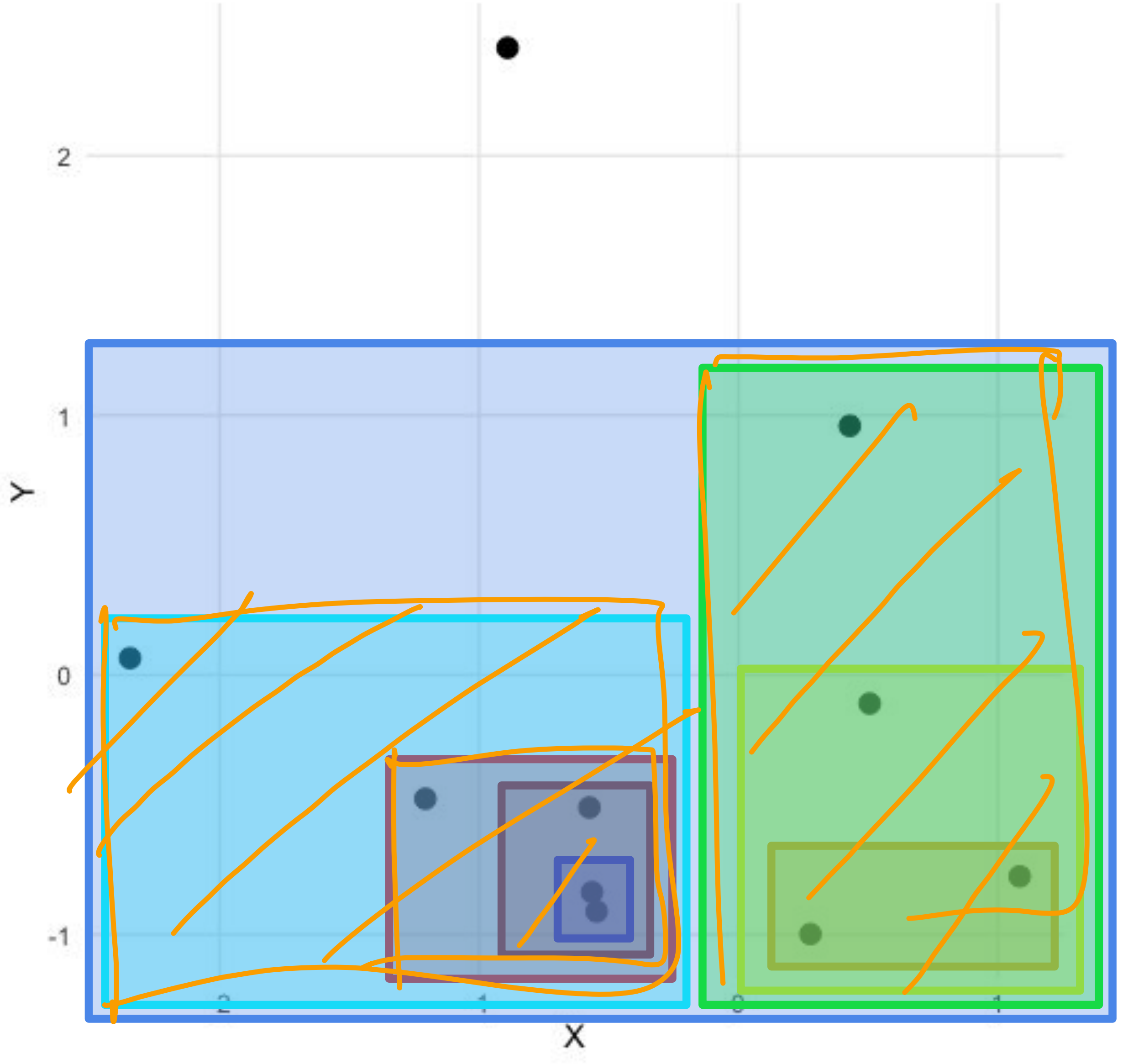
Algorithm



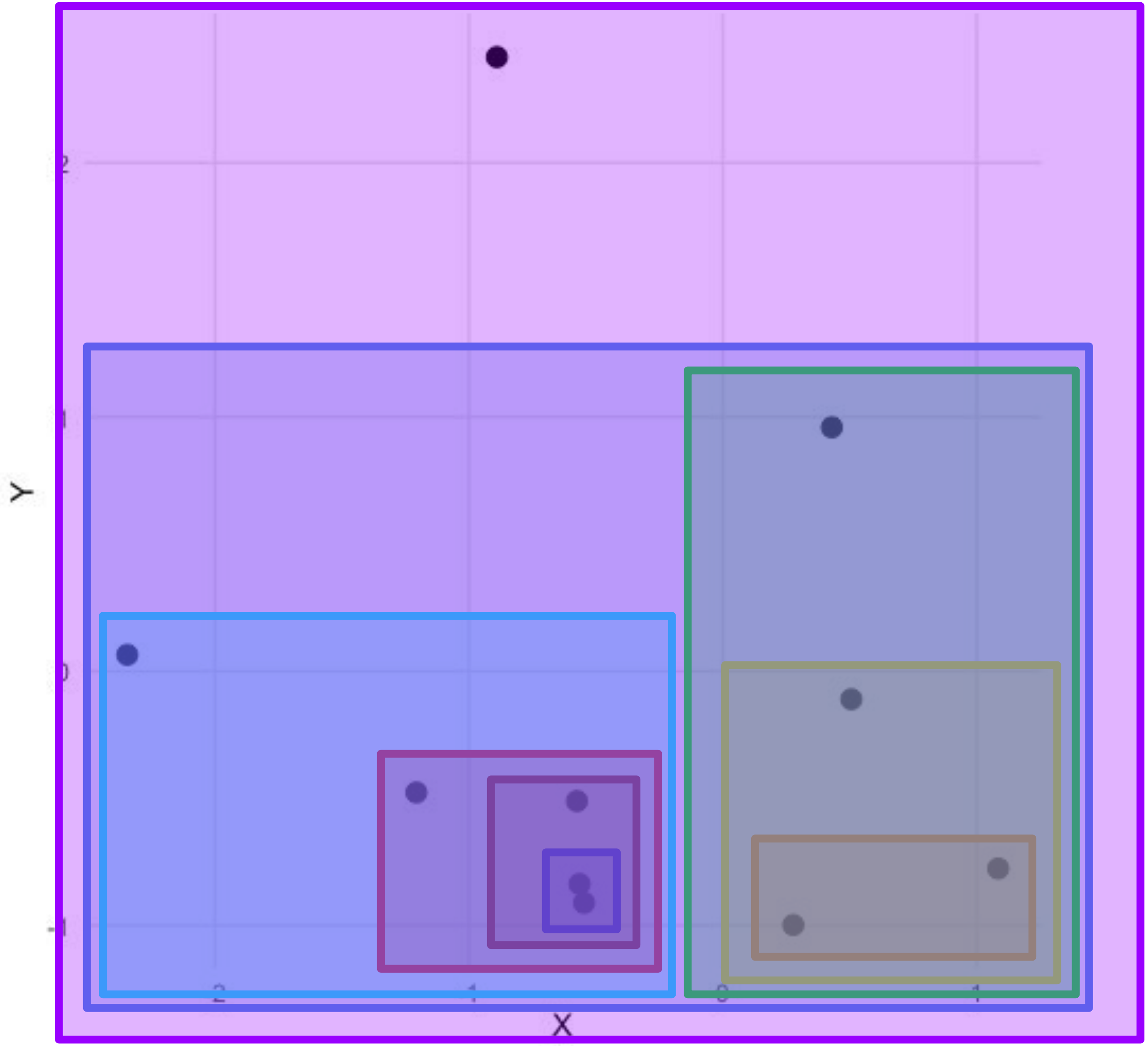
Algorithm



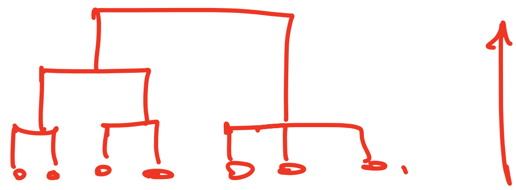
Algorithm



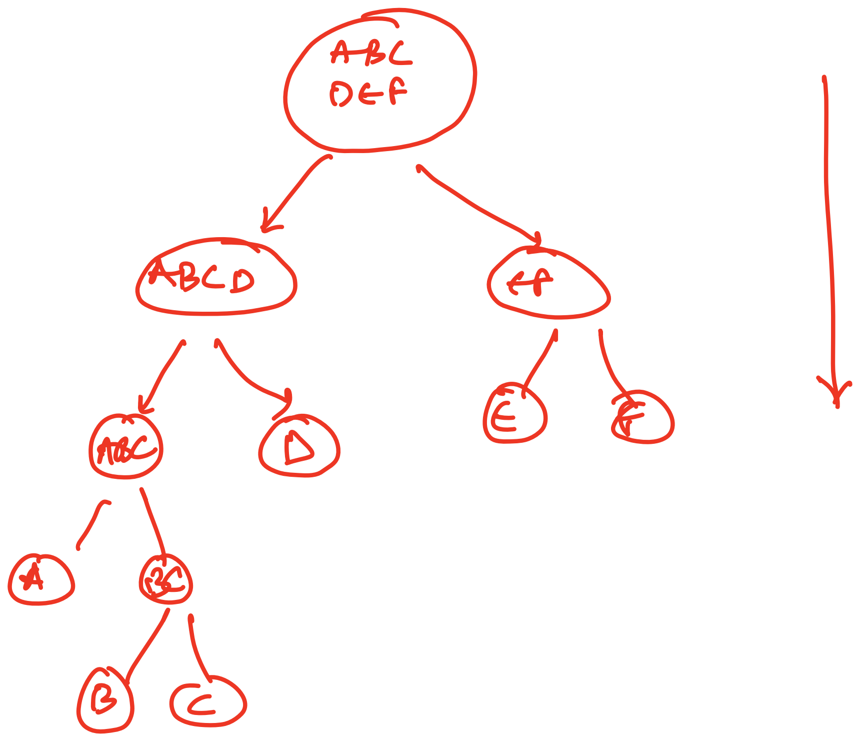
Algorithm



Agglomerative → BU



Divisive → (Top Down)



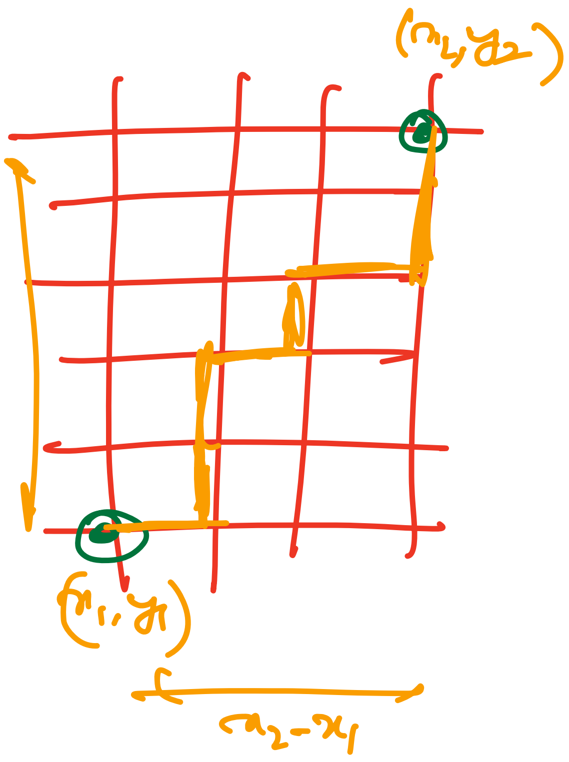
Distance Metrics

(Distance b/w Data Points)

- **Euclidean**: Continuous Data
- **Manhattan**: High Dimensions
- **Hamming**: Categories
- **Cosine**: Word Counts

$$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$$

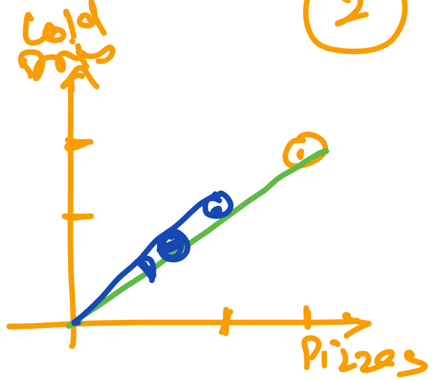
$$(x_2-x_1)+(y_2-y_1)$$



10111
10001

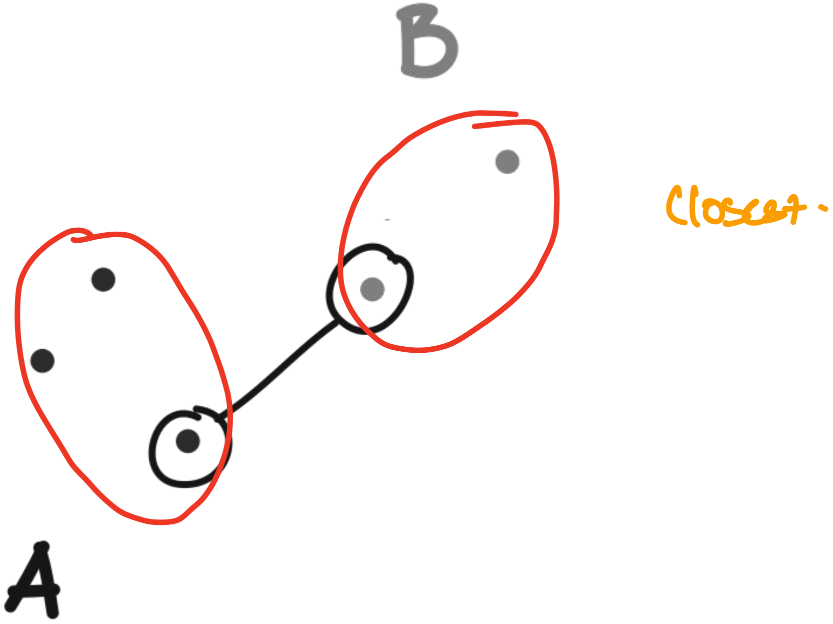
2

Cosine:



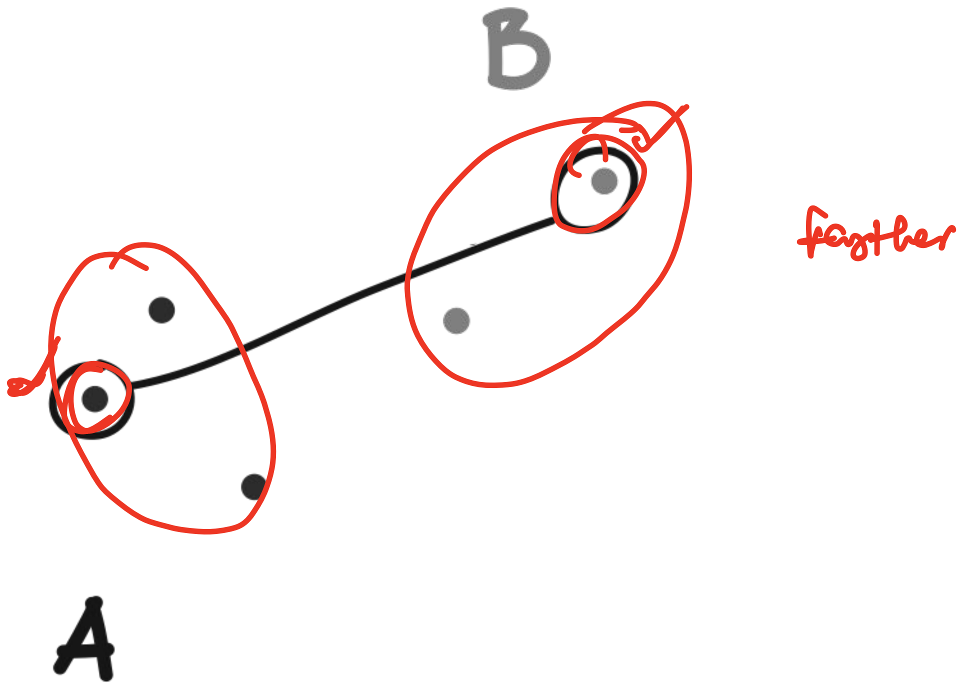
Linkage Criteria

(Distance b/w 2 clusters)



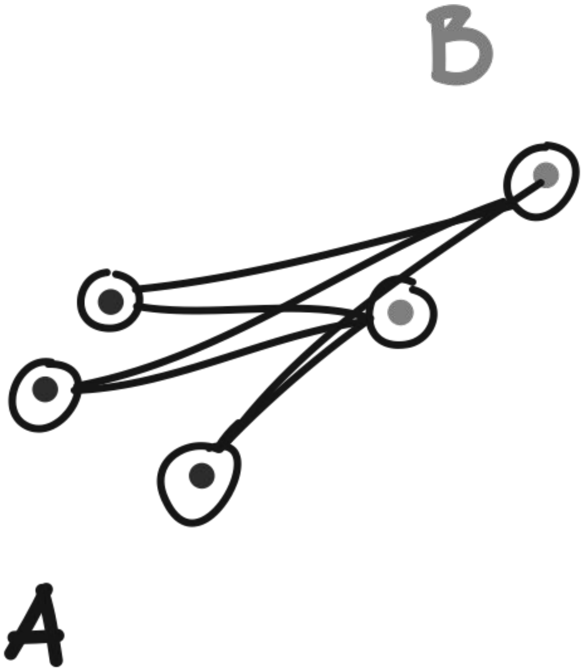
Single

Linkage Criteria



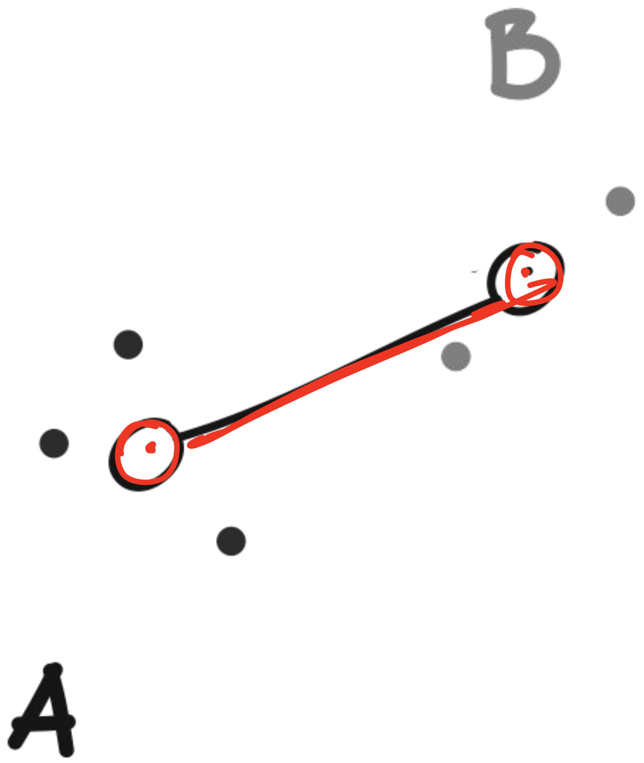
Complete

Linkage Criteria



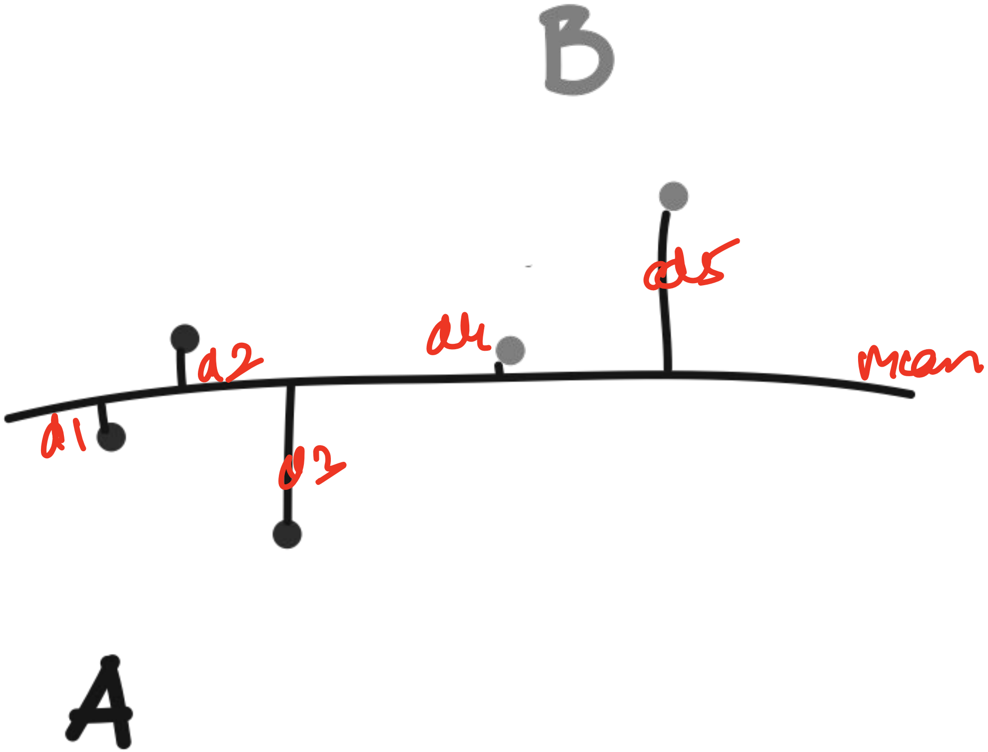
Average

Linkage Criteria



Centroid

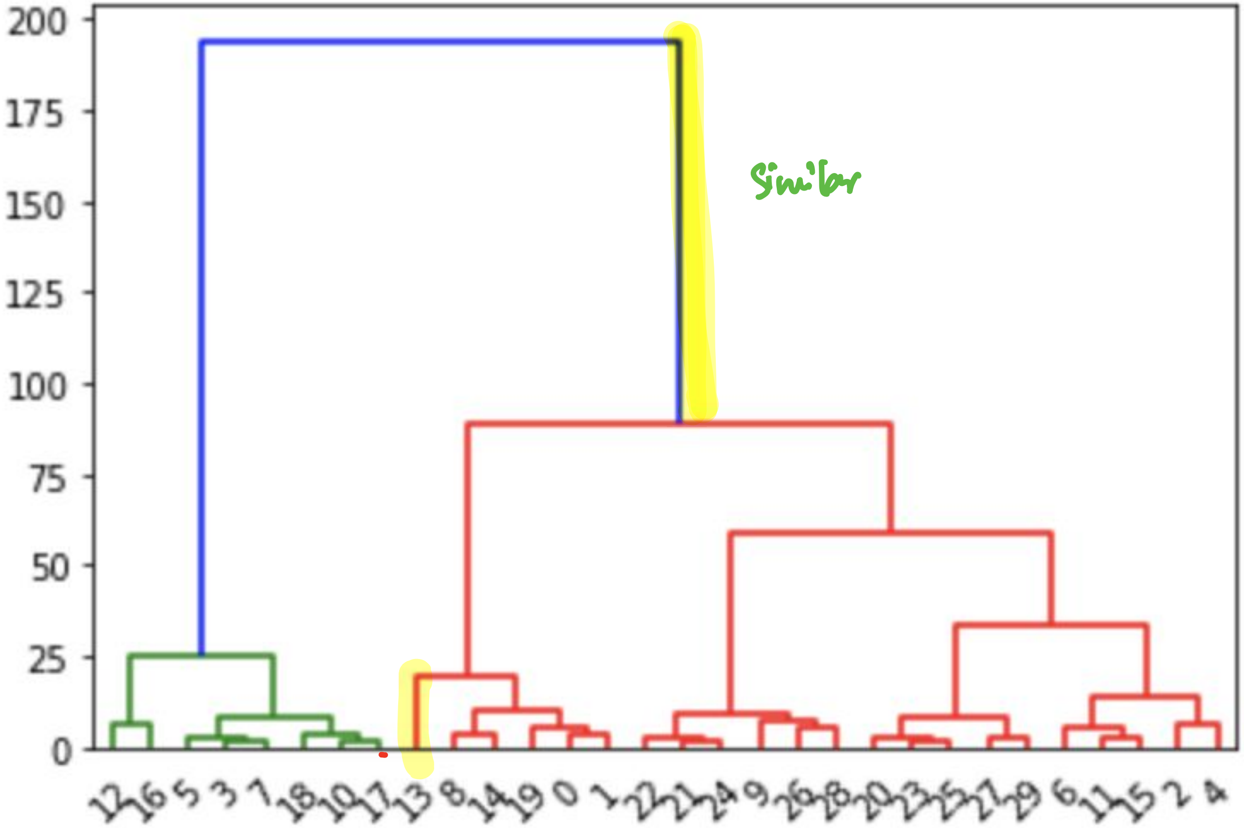
Linkage Criteria



(Variance)

Ward's

Reading a Dendrogram



DBSCAN:

Feature Selection:

10 features
↳ all of them might not be relevant.

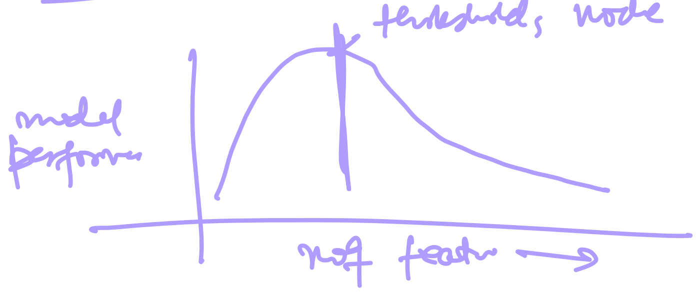
Features select

Curse of Dimensionality → features
↳ lot of features, model performance ↓

Predict Salary

Humidity | Temperature | College | CGPA | Branch | Experience

irrelevant X



Features

- Irrelevant
- Redundant

ht (cm)	ht (inch)	ht (m)
	X	X

age	DOB
	X

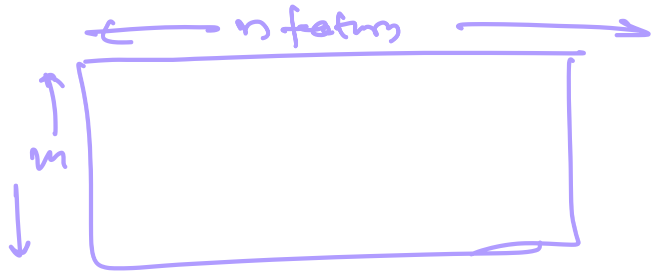
Curse of ~~Dimensionality~~ — no of features

Dimensionality Reduction

Feature Selection	Feature Extraction
a, b, c —— z }26	a, b, c —— z }26
top 5 features: b, f, i, z, e	5
	a+d, b*e, g+2*h + ——
	Algo: PCA

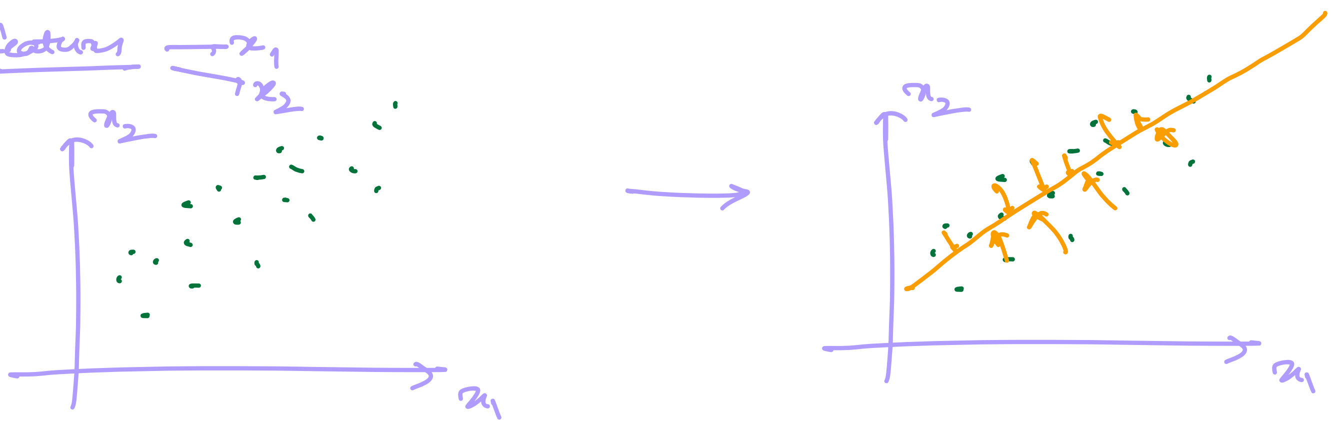
Principal Component Analysis (PCA)

$x \in R^n$



$x \in R^k$ $\quad k < n$

2 features → x_1, x_2

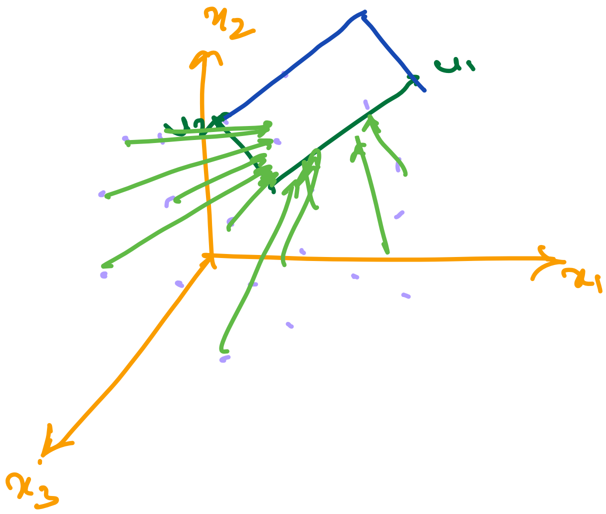


Application of PCA:

① Data Compression

50 Dimensional → 5D Data

↳ 50 values ↓ 5 values

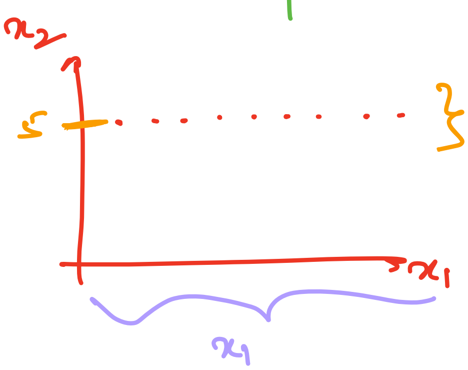


② Data Visualisation

③ Speed up Computation

Variance → Spread

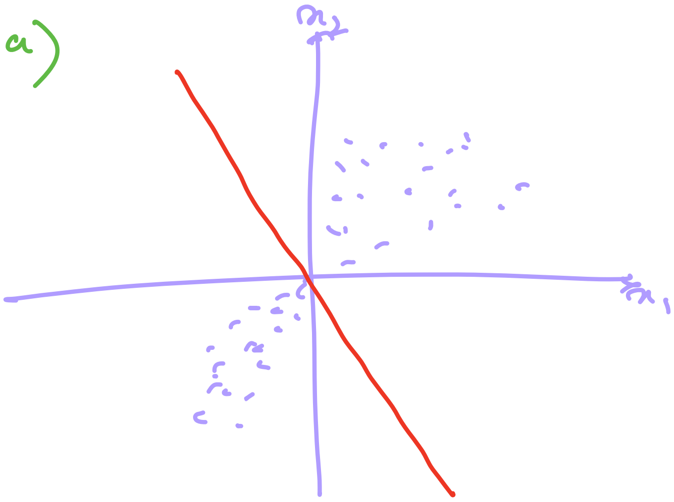
$$var = \frac{\sum (x-\bar{x})}{n}$$

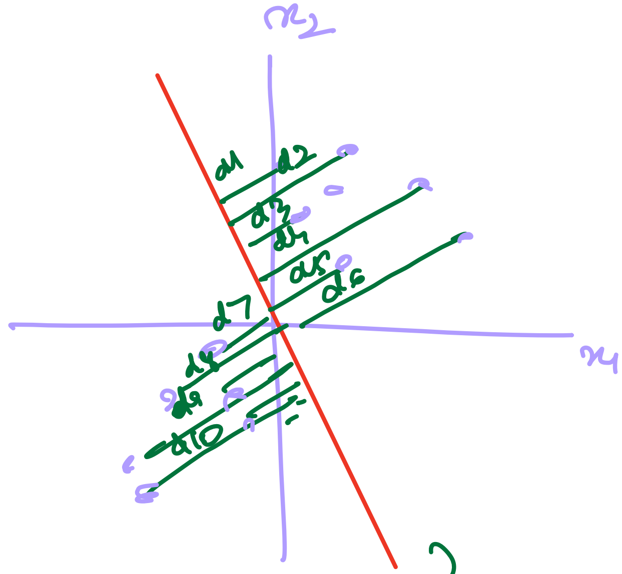


} variance $x_2 = 0$

→ ignore x_2
consider x_1

Case a)





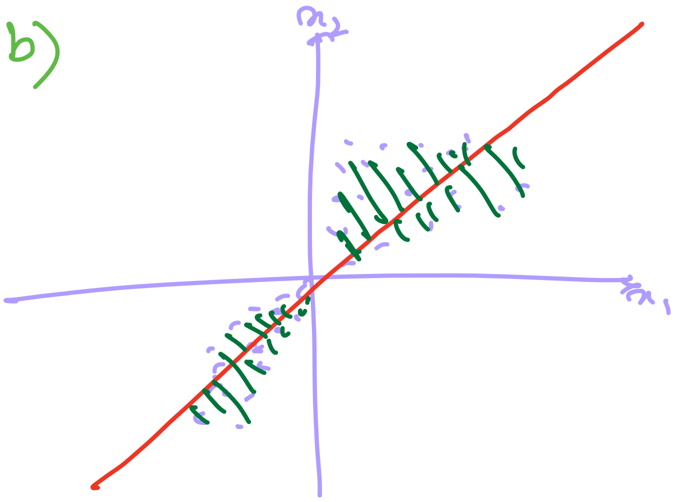
Perpendicular Distance

Projection Error

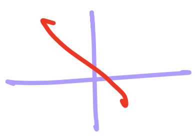
$$d_1^2 + d_2^2 + \text{——} d_{10}^2$$

$$\sum_{i=1}^{m} d_i^2$$

Case b)



Case a

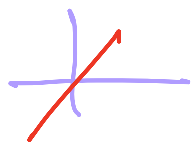


Projection Error high

spread less

(variance less)

Case b



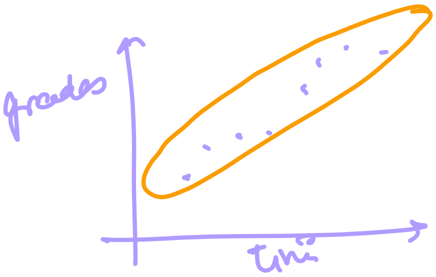
PE less

spread high

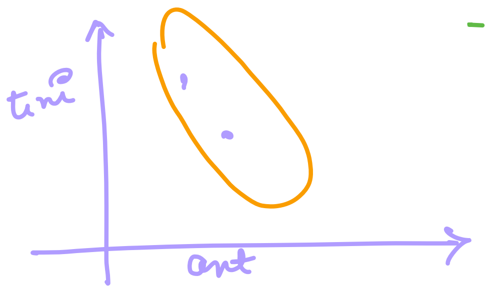
(variance high)

Covariance

↳ 2 variables are related to each other.



+ve

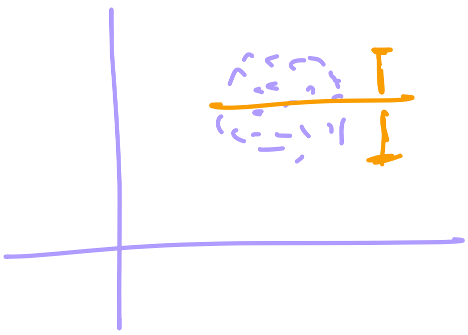


-ve

D → B

flight: amt ↑ time ↓

Train: amt ↓ time ↑



0

$$Cov(x, y) = \frac{\sum_{i=1}^{n} (x - \bar{x})(y - \bar{y})}{n-1}$$

$$Cov(x, x) = \text{Variance}(x) = \frac{\sum_{i=1}^{n} (x - \bar{x})^2}{n-1}$$

Covariance matrix:

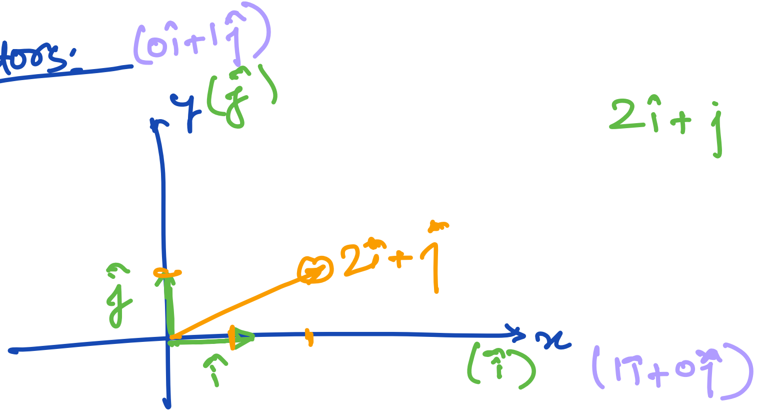
2 features

$$\begin{bmatrix} Cov(x_1, x_1) & Cov(x_1, x_2) \\ Cov(x_2, x_1) & Cov(x_2, x_2) \end{bmatrix}$$

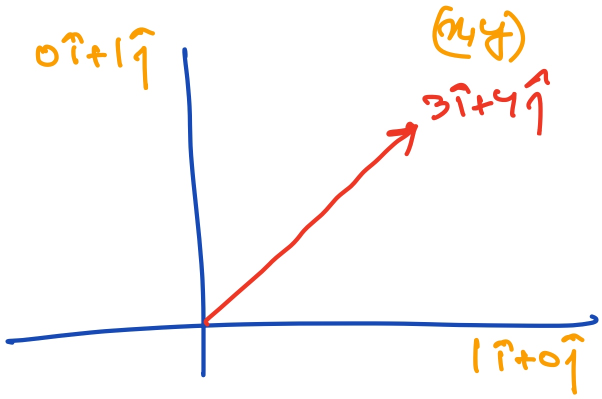
2×2

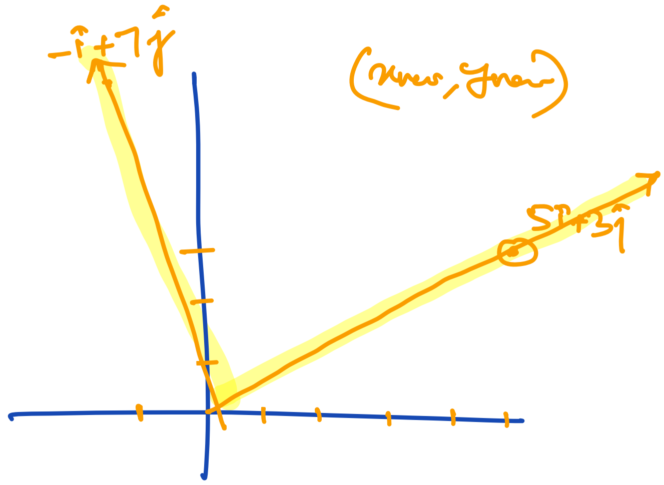
Symmetric matrix

Vectors:



$2\hat{i}+\hat{j}$





$$\underset{v}{\text{Transformed}} = 3\,(\underbrace{\text{Transformed } \hat{i}}_{\begin{bmatrix}5\\3\end{bmatrix}}) + 4\,(\underbrace{\text{Transformed } \hat{j}}_{\begin{bmatrix}-1\\7\end{bmatrix}})$$

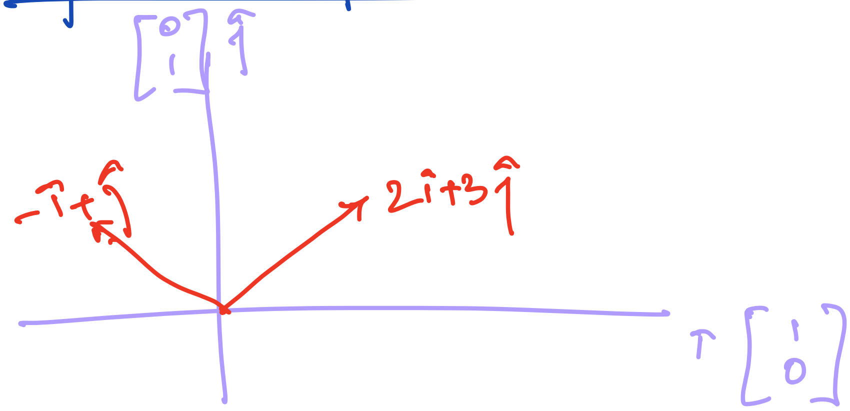
$$= 3\begin{bmatrix}5\\3\end{bmatrix} + 4\begin{bmatrix}-1\\7\end{bmatrix}$$

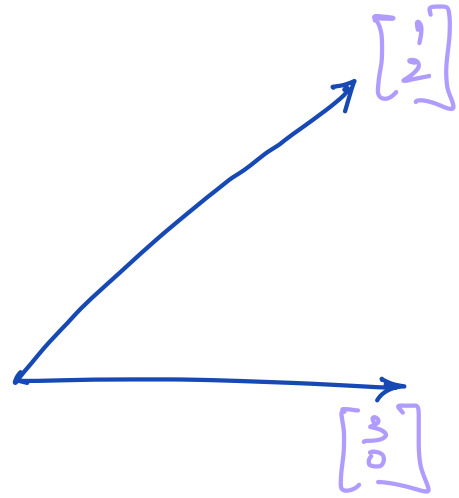
$$= \begin{bmatrix}15\\9\end{bmatrix} + \begin{bmatrix}-4\\28\end{bmatrix} = \begin{bmatrix}11\\37\end{bmatrix} \rightarrow 11\hat{i}+37\hat{j}$$

Transformation Matrix

$$\begin{bmatrix}\underbrace{\begin{matrix}5\\3\end{matrix}}_{\text{new } \hat{i}} & \underbrace{\begin{matrix}-1\\7\end{matrix}}_{\text{new } \hat{j}}\end{bmatrix} \underbrace{\begin{bmatrix}3\\4\end{bmatrix}}_{\text{coordinates of vector}} = \begin{bmatrix}15-4\\9+28\end{bmatrix} = \underbrace{\begin{bmatrix}11\\37\end{bmatrix}}_{\text{new position of vector}}$$

Eigen Values & Eigen Vectors





$2\hat{i}+3\hat{j}$

$$\begin{bmatrix}3 & 1\\0 & 2\end{bmatrix}\begin{bmatrix}2\\3\end{bmatrix} = \begin{bmatrix}6+3\\0+6\end{bmatrix} = \begin{bmatrix}9\\6\end{bmatrix} = 3\begin{bmatrix}3\\2\end{bmatrix}$$

$\hat{i}+\hat{j}$

$$\begin{bmatrix}3 & 1\\0 & 2\end{bmatrix}\begin{bmatrix}-1\\1\end{bmatrix} = \begin{bmatrix}-3+1\\2\end{bmatrix} = \begin{bmatrix}-2\\2\end{bmatrix} = 2\begin{bmatrix}-1\\1\end{bmatrix}$$

mag. dirn

Eigen vectors

$$A\cdot\bar{V} = \lambda\cdot\bar{V}$$

Transform matrix

Eigen vector

Eigen value (Scalar)

$$A\bar{V} = \lambda I.V$$

$$I = \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$$

$$A\bar{V} - \lambda I\bar{V} = 0$$

$$(A-\lambda I)\bar{V} = 0$$

$$\lambda I = \begin{bmatrix}\lambda & 0\\0 & \lambda\end{bmatrix}$$

$$|A - \lambda I| = 0$$

$$\begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = 0$$

$$\begin{bmatrix} 3-\lambda & 1 \\ 0 & 2-\lambda \end{bmatrix} = 0$$

$$(3-\lambda)(2-\lambda) = 0$$

$$\lambda = 2, 3$$

Eigen Values

---

15 Nov 2023

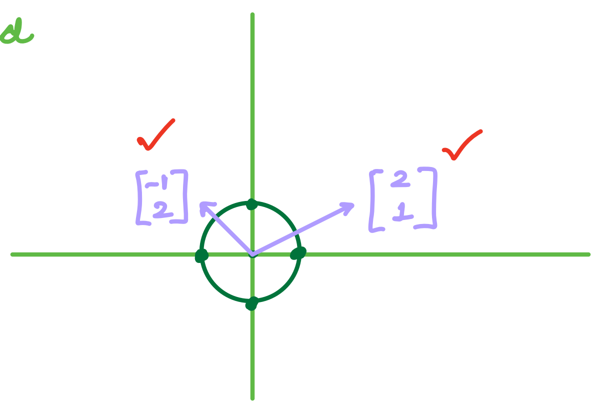
$\mathrm{Cov}(x_1, x_1)$ $\quad$ $\mathrm{Cov}(x_1, x_2)$

Covariance Matrix = $\begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$

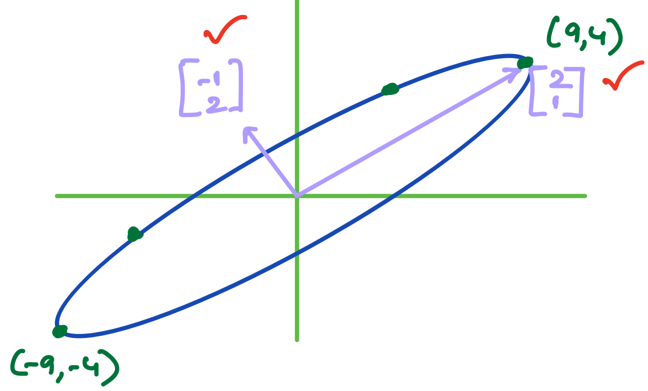
$\mathrm{Cov}(x_2, x_1)$ $\quad$ $\mathrm{Cov}(x_2, x_2)$

- Symmetric
- used as transformation matrix

old



new



Transformation matrix

$$\begin{pmatrix}9 & 4\\ 4 & 3\end{pmatrix}\begin{pmatrix}x\\ y\end{pmatrix} = x\begin{bmatrix}9\\ 4\end{bmatrix} + y\begin{bmatrix}4\\ 3\end{bmatrix} = \begin{bmatrix}9x+4y\\ 4x+3y\end{bmatrix} \checkmark$$

$$(x,y) \longrightarrow (9x+4y,\ 4x+3y)$$

$$(0,0) \longrightarrow (0,0)$$

circle

$$(1,0) \longrightarrow (9,4)$$
$$(0,1) \longrightarrow (4,3)$$
$$(-1,0) \longrightarrow (-9,-4)$$
$$(0,-1) \longrightarrow (-4,-3)$$

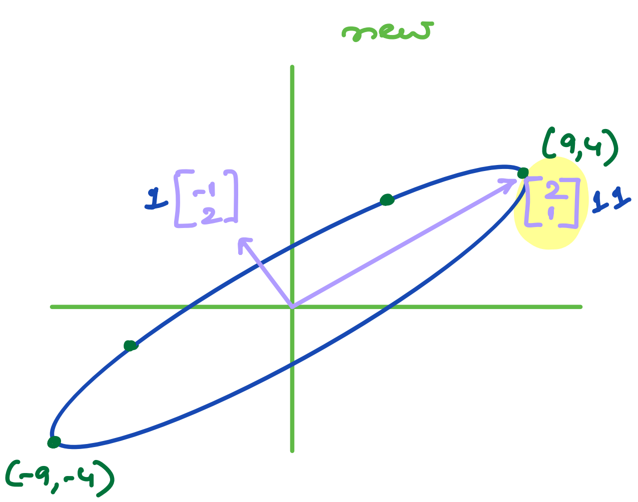
plot → ellipse

$$(2,1) \longrightarrow (22,11) = 11(2,1)$$
$$(-1,2) \longrightarrow 1(-1,2)$$

Eigen vector

eigen value

In PCA, goal was to find new axis.



$\begin{bmatrix}-1\\ 2\end{bmatrix}$ and $\begin{bmatrix}2\\ 1\end{bmatrix}$ are going to be the new axis.

These are the lines having highest spread.

Highest variance will be given by Eigen vector.

EV1 → 11
EV2 → 1

Eigen value → Magnitude

Spread

we will project all points on Eigen vector 1 bcz its magnitude is larger.

Steps:

1. Load Data
2. Standardization

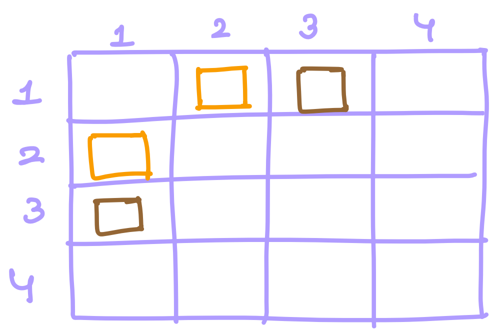
$\mu = 0$ (mean = 0)

$\sigma = 1$ (std = 1)

3. Covariance Matrix : how are 2 features related to each other.

n features

4 4



$n \times n$

4×4

4. Eigen Values & Eigen Vector

4

evec1 → evalue1

evec2 → evalue 2

evec3 → evalue 3

evec4 → evalue 4

5. larger eigen value means spread/variance is more along that vector.

evec1 → 50

evec2 → 30

evec3 → 60

evec4 → 10

eigen values

Choose the axis along which spread is larger.
spread is given by eigen value.

Sort them acc. to eigen value

$$\left.\begin{array}{l} evec\,3 \rightarrow 60 \\ evec\,1 \rightarrow 50 \\ evec\,2 \rightarrow 30 \\ evec\,4 \rightarrow 10 \end{array}\right\} \text{sorting} \Big\} \underline{\underline{\overset{\rightarrow 2}{k}\ \text{vectors.}}}$$

6. Pick top k eigen vectors.

$\downarrow$ new dimension.

7. Projection

new axis : max variance

$$\underbrace{\begin{bmatrix} \text{matrix of data points} \end{bmatrix}}_{n \text{ features}} \begin{bmatrix} | & | \\ evec\,3 & evec\,1 \\ | & | \end{bmatrix} = \begin{bmatrix} \quad \end{bmatrix}$$

$(m \times n)$ $\qquad$ $(n \times k)$

$\downarrow$ final points in $\underline{\underline{2D}}$.

$(m \times k)$