Day 14: SVMs Part 2 and Clustering

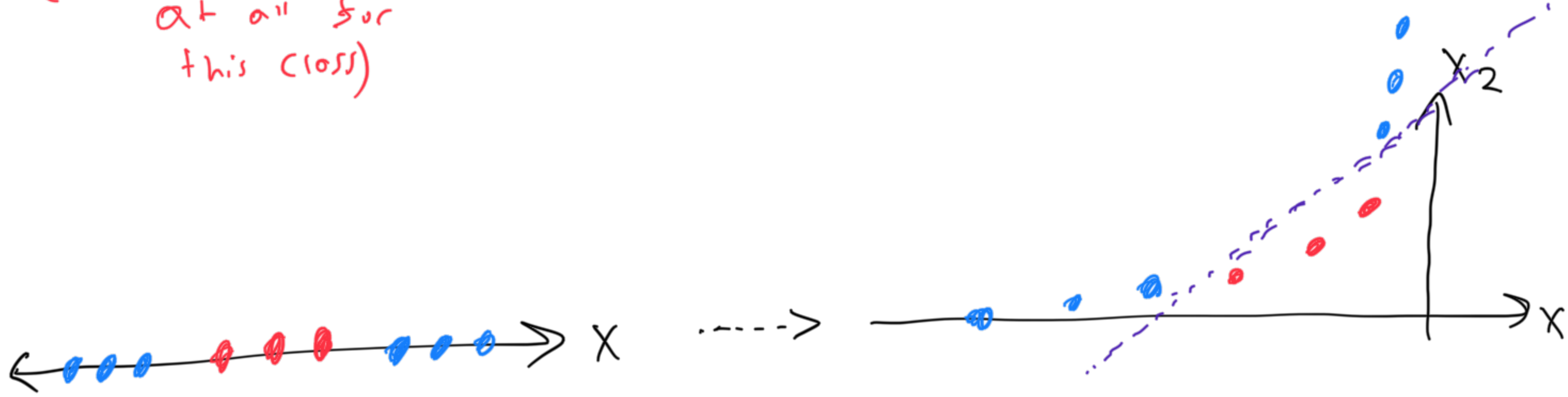
$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \underbrace{\left(X_i^T X_j \right)}$$

Subject to; $\ast\ \alpha_i \geq 0$

$\ast\ \sum_{i=1}^{n} \alpha_i y_i = 0$

(equation is not important at all for this class)

Key Observation: $X_i^T X_j$ is the bulk of the computation, especially in high dimensions



The Kernel Trick

Find a "Kernel function" which tells us what the dot product would be in a higher dimension

... to convert the data into

without requiring us to convert the data into the higher dimension and calculate the dot product.

$$\left(\text{Recall: } [a_1, a_2, a_3] \cdot [b_1, b_2, b_3] = a_1 b_1 + a_2 b_2 + a_3 b_3\right)$$

Polynomial Kernel

$$K(a, b) = (a^T b + r)^d$$

r, d are hyperparameters

d stands for: degree/dimension

a = data point #1

b = data point #2

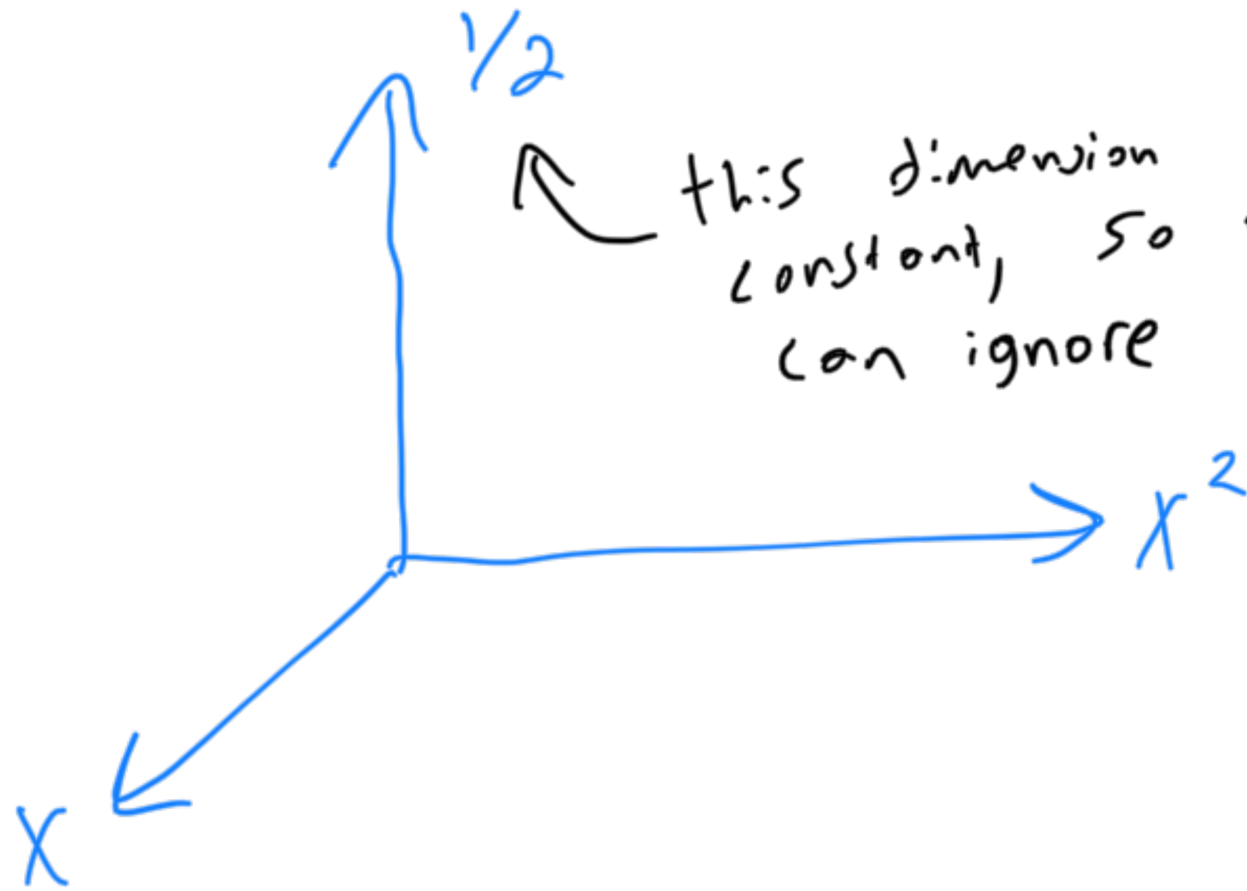
Example: $r = 1/2$ $d = 2$

$$(ab + 1/2)^2 = (ab + 1/2)(ab + 1/2)$$

$$= a^2 b^2 + ab + 1/4$$

$$= [a,\ a^2,\ 1/2]^T \cdot [b,\ b^2,\ 1/2]$$

Thus, $[x_i,\ x_i^2,\ 1/2]$ is the coordinate in the higher dimension



this dimension is a constant, so we can ignore

Another example:

$$(ab+1)^3 = a^3b^3 + 3a^2b^2 + 3ab + 1$$

$$= [a^3,\ \sqrt{3}a^2,\ \sqrt{3}a,\ 1] \cdot [b^3,\ \sqrt{3}b^2,\ \sqrt{3}b,\ 1]$$

There are other kernels as well, including some which project data into ∞-dimensional space (via Taylor Series Expansion) (see Radial Basis Kernel).

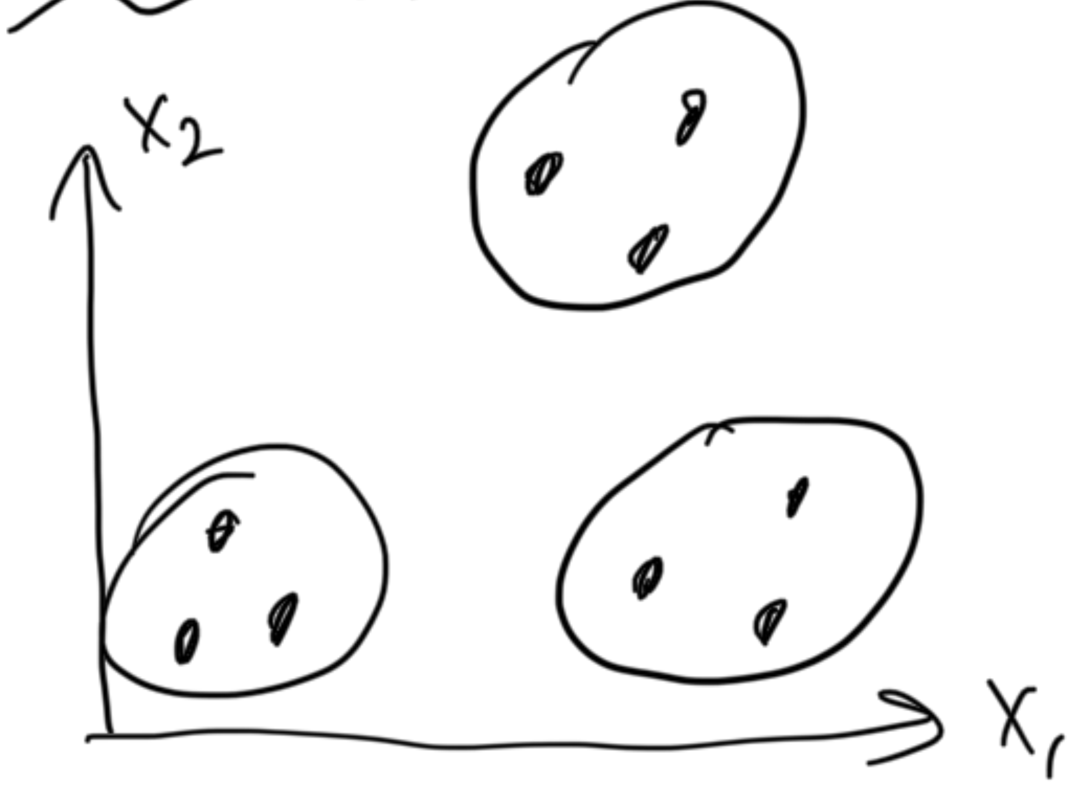
SVM: SVC + Kernel

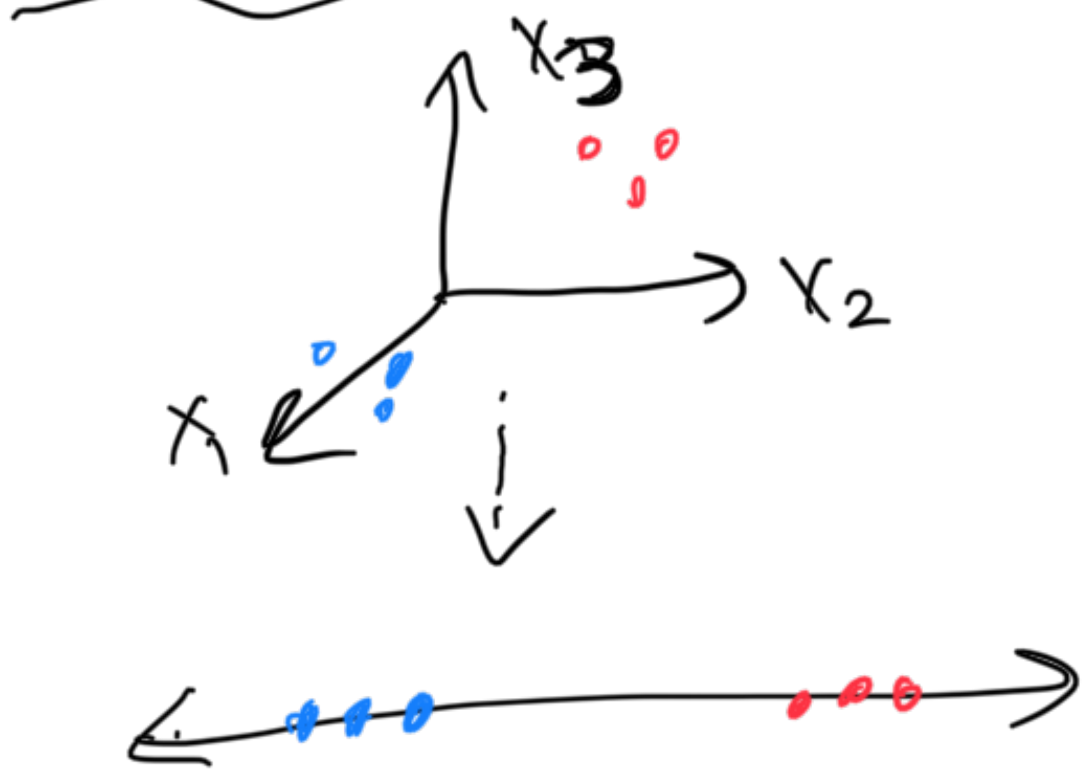
Unsupervised Learning

Learning without labels

Dimensionality Reduction

Clustering





Clustering

Use cases:

* Group website users into distinct categories for advertising
* Recommendation systems for Spotify, YouTube/Netflix/Amazon
* Discover categories/subcategories

(of diseases, species, etc.)

...

Most popular clustering algorithm is...

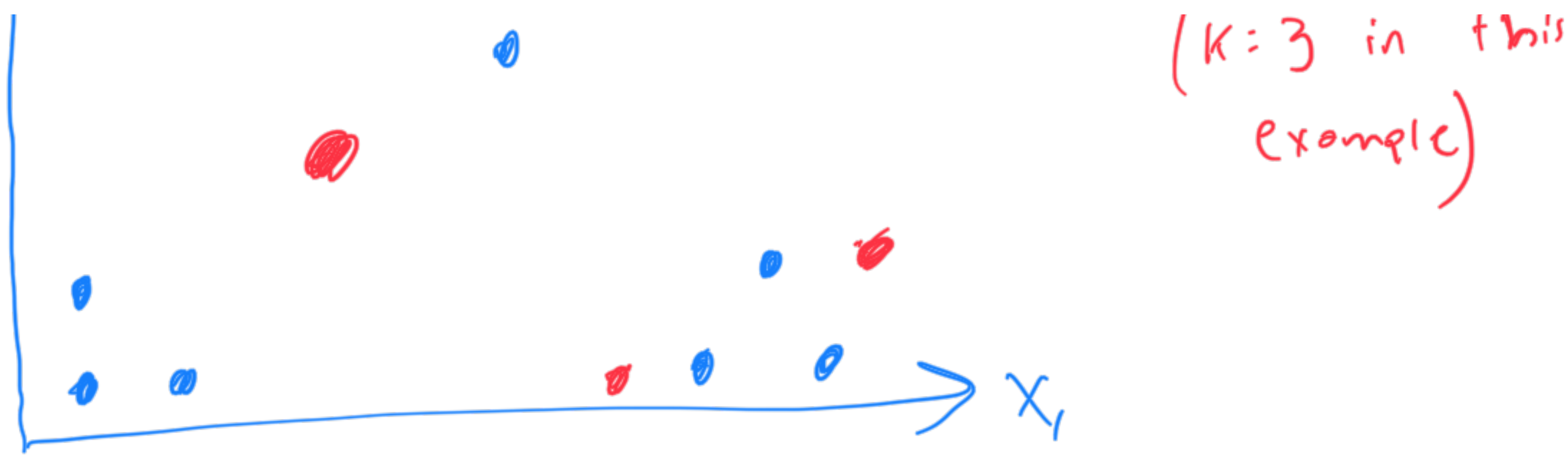
K-Means Clustering

Inputs: data, K (hyperparameter)

Output: cluster assignment

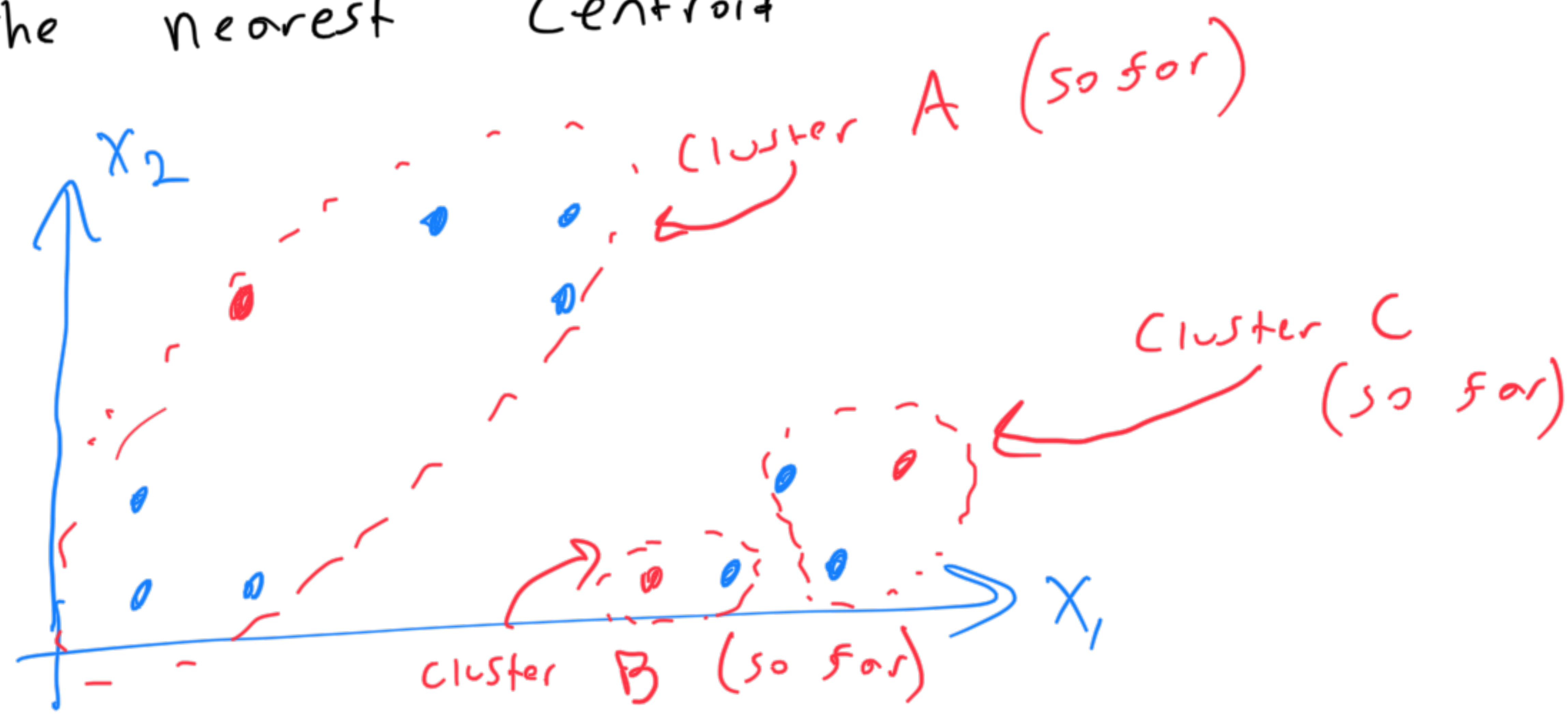
these are not necessarily data points in the data set

1. randomly assign K cluster "Centroids" (the center of the cluster)

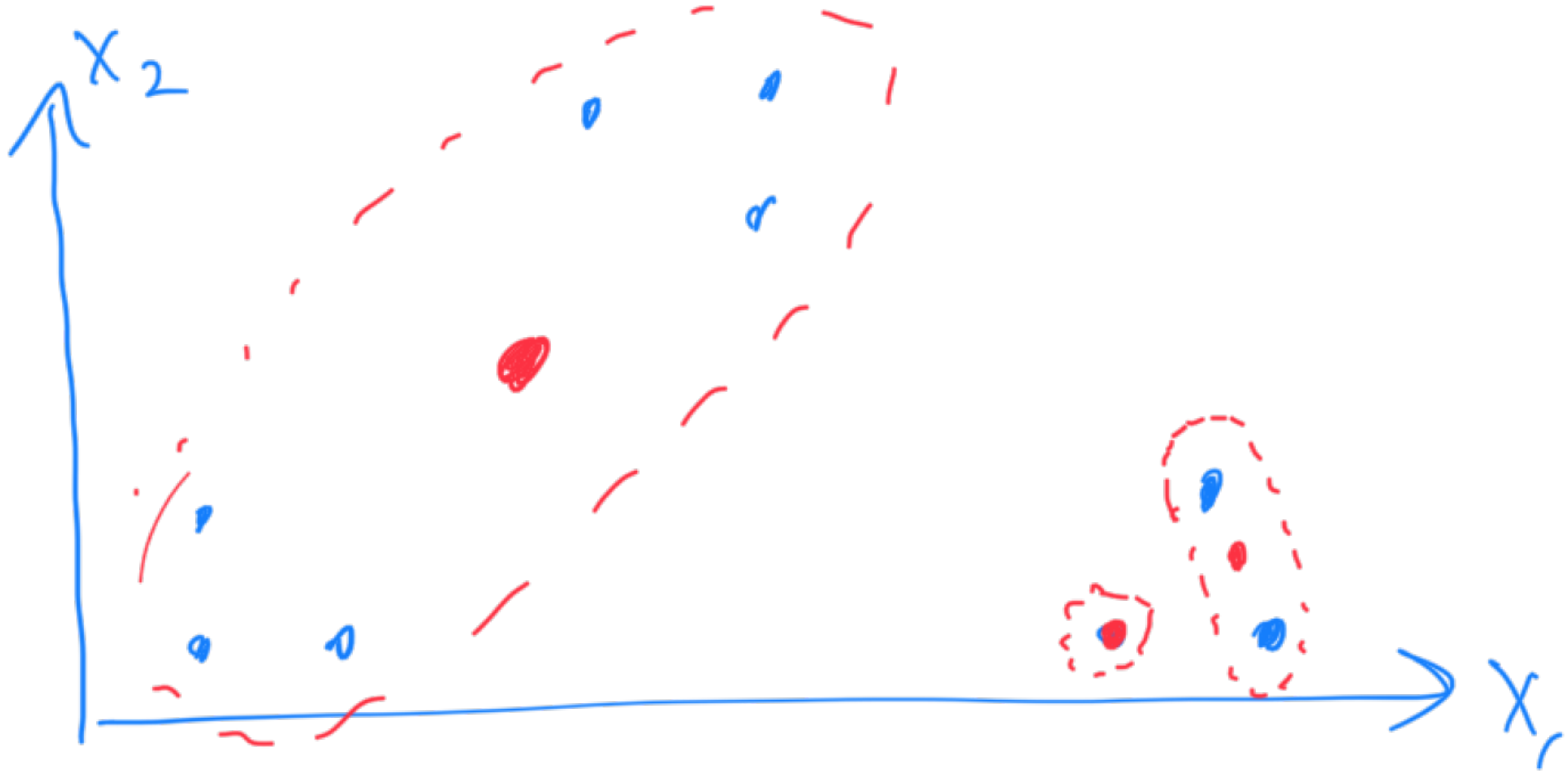


② while the assignment of data points to centroids is still changing:

ⓐ assign each data point to the nearest centroid

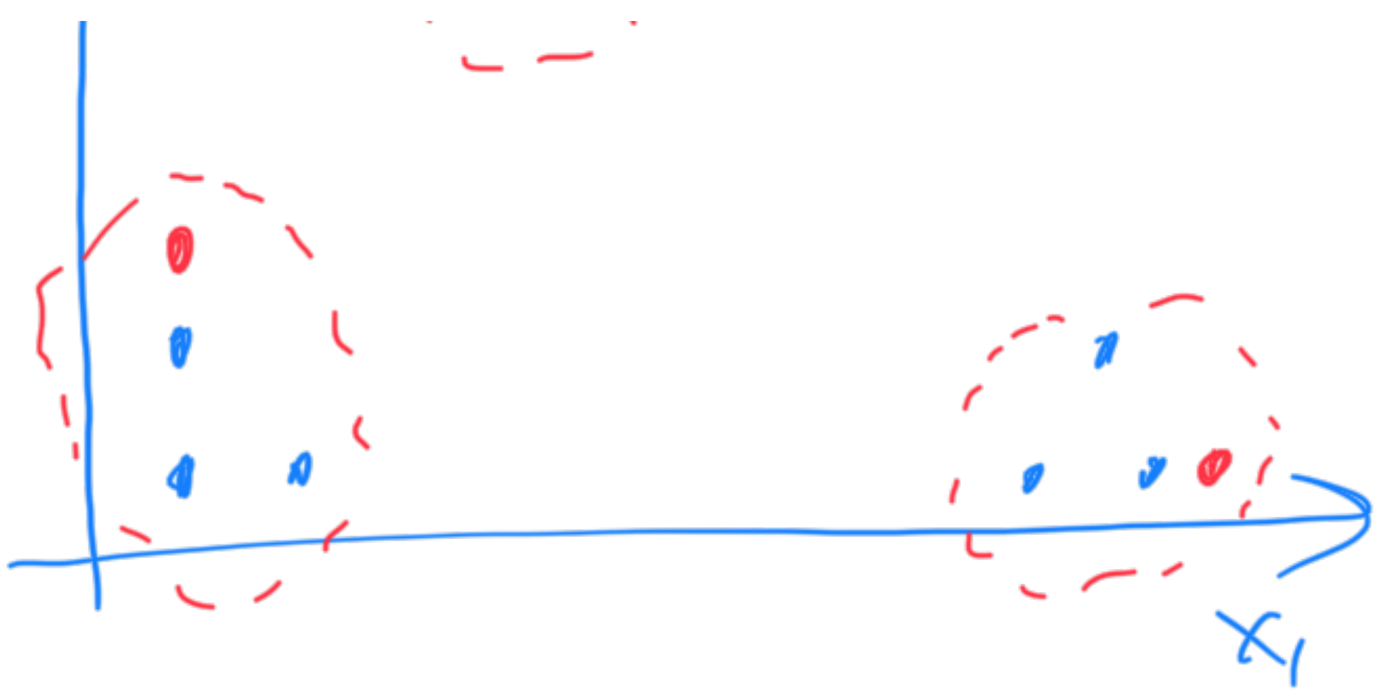


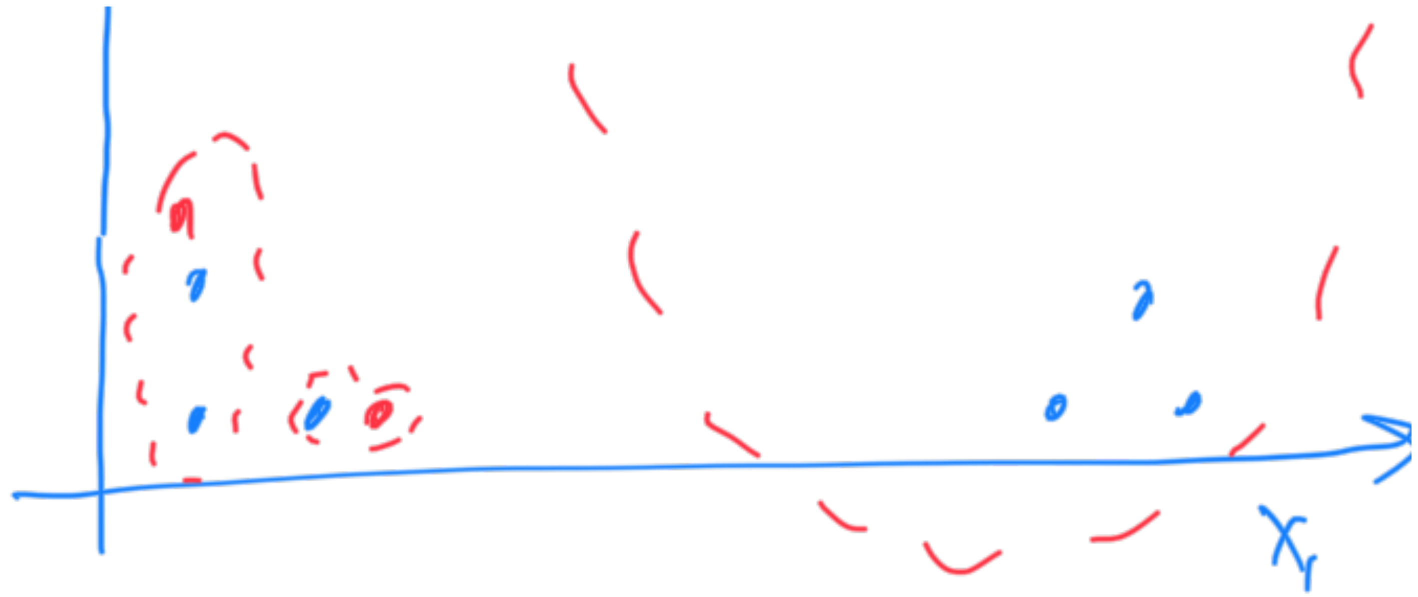
(b) recalculate the centroid as the mean of the data points in each of the current clusters



(3) return the final clusters

<u>Important Note:</u> the end results heavily depend on the initialization

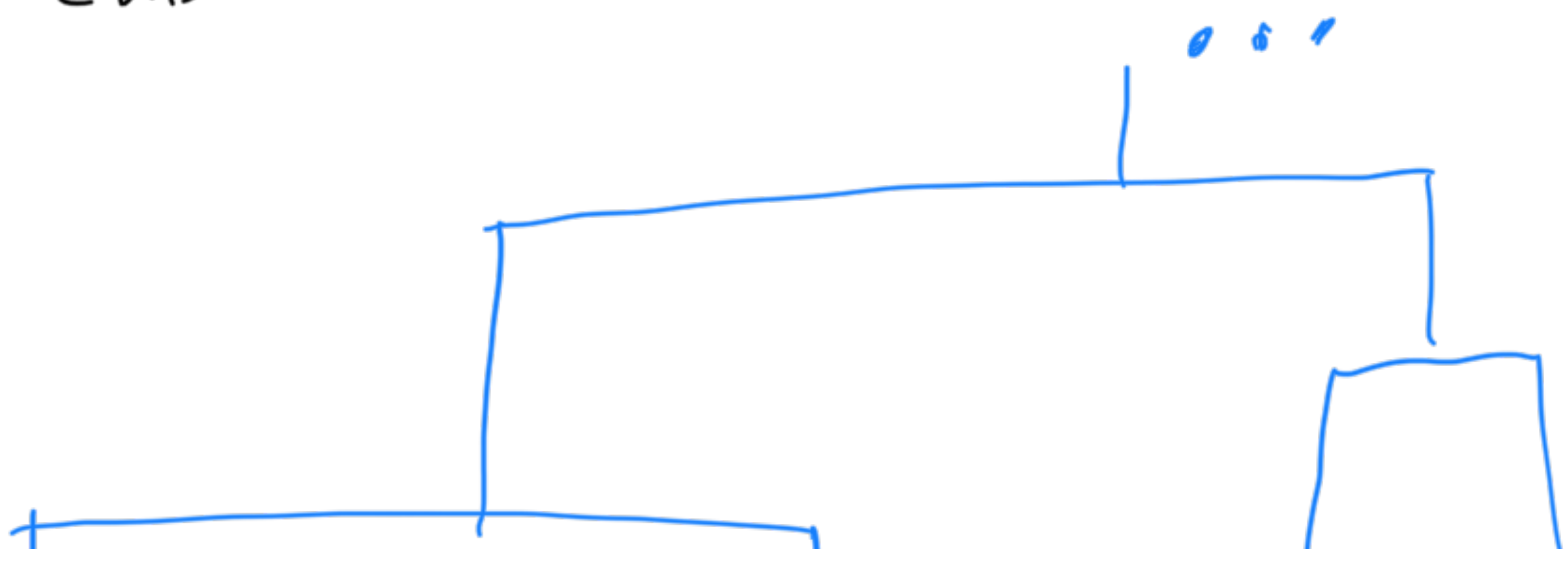


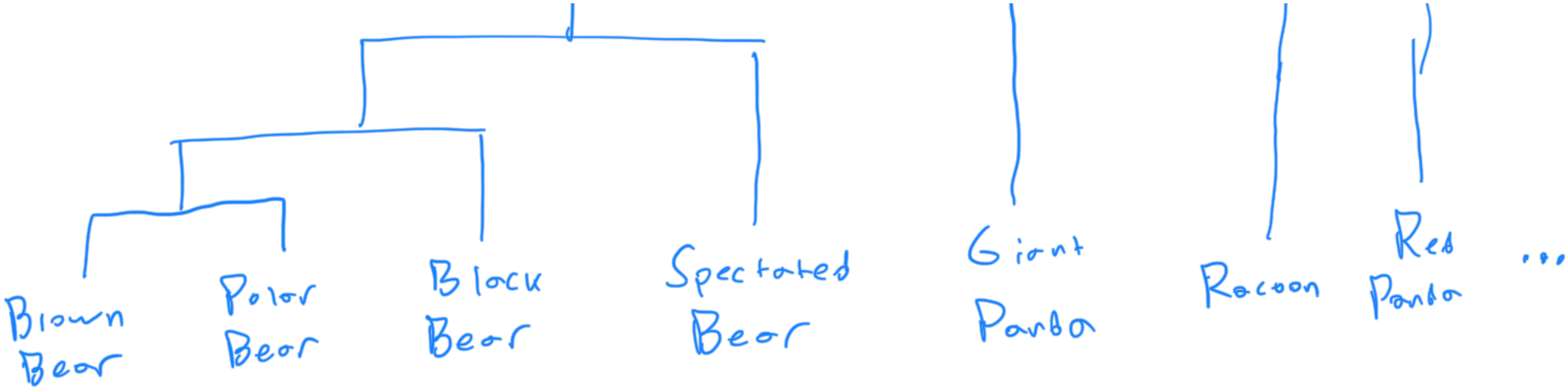


Hierarchical Clustering

Goal is to build a hierarchy of clusters

Big use case: construction of evolutionary trees



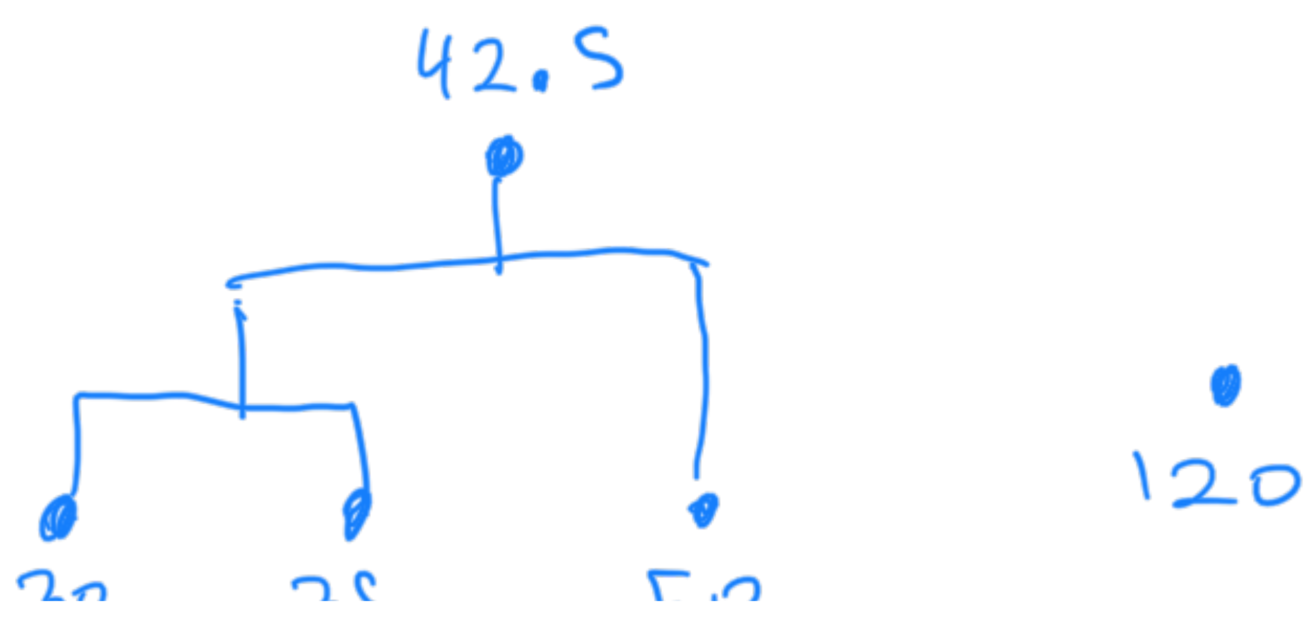


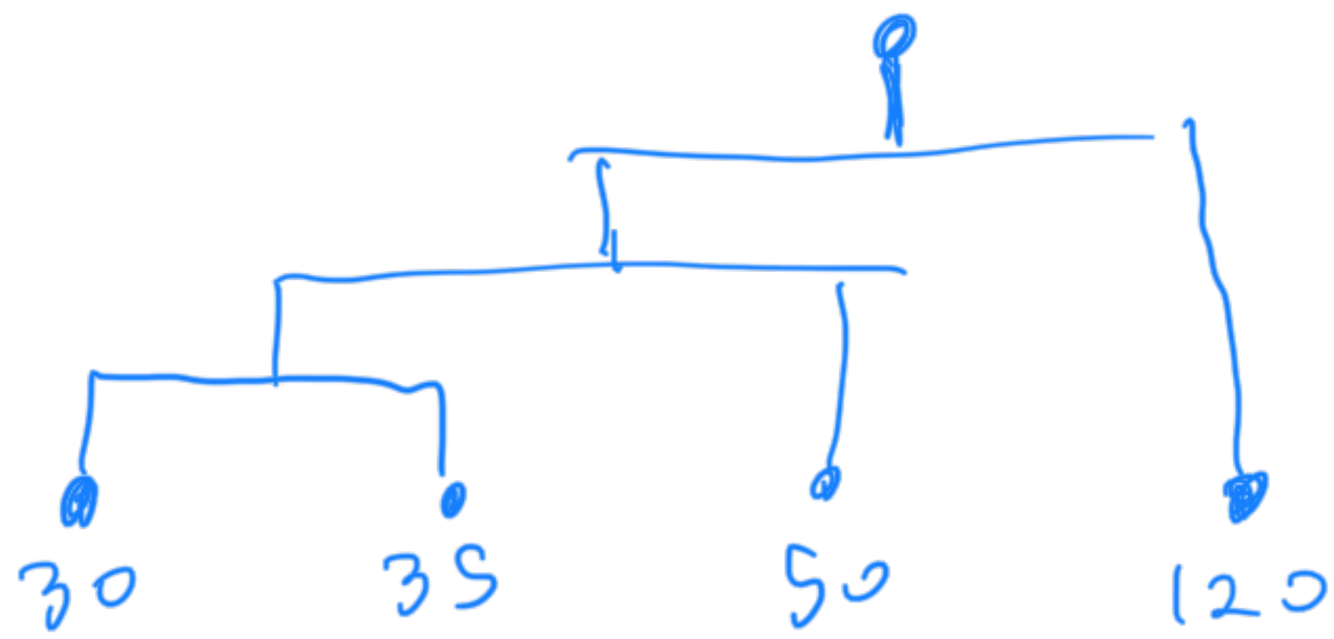
Agglomerative ("bottom up") hierarchical clustering:

1. each data point is its own cluster

(2) While there is more than 1 cluster left:

merge the two closest clusters, using the mean of the data points in the new cluster as its "centroid"





Crucial Note: can define similarity between 2 points in many different ways

Similarity$([3, -20, 10], [4, 8, 0])$

↳ Euclidean Distance (L2):

$$\sqrt{(3-4)^2 + (-20-8)^2 + (10-0)^2}$$

↳ Manhattan Distance (L_1).

$$|3-4| + |-20-8| + |10-0|$$

...

Similarity("AATCCCG", "A--CCGA")

↳ Levenshtein Distance (Edit Distance)

(string similarity metric used in text processing and bioinformatics)

...

<u>Another Crucial Note</u>: there are many ways to represent data!!!!!

A key area of AI research and practice.

* How do we represent our data?
* How do we define similarity between 2 data points?

Feature Engineering

How do we represent our data?

Example: Action Recognition

$y = f\Big($  $\Big)$

Standing

model may overfit to irrelevant features

vs.

$y = f\Big($  $\Big)$

Kicking

Can represent the data with [illegible]

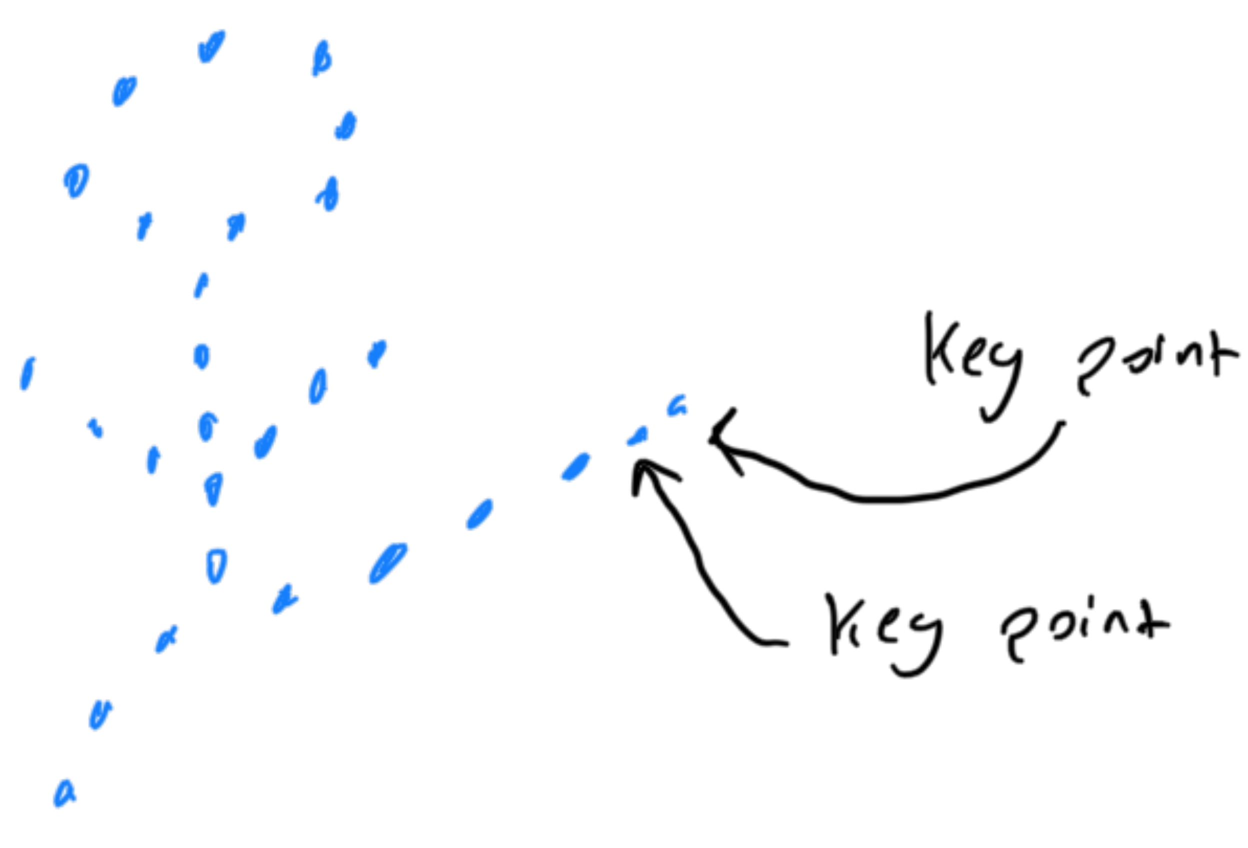
feature vector that only describes

the relevant parts of the data:

$$f\left(\begin{bmatrix} \text{Key point 1 coords} \\ \text{Key point 2 coords} \\ \cdots \\ \cdots \\ \cdots \end{bmatrix}\right)$$

much less likely to overfit

Aside: Key points



Four main approaches:

1. transform data using a pre-defined transformation

"how much wood would a wood chunk chunk" $\xrightarrow{\text{Bag of Words}}$ $\begin{array}{cccc} 2 & 2 & 0 & \cdots \\ \text{wood} & \text{chunk} & \text{apple} & \cdots \end{array}$

2. transform data using a learned representation

"word" → $\boxed{\text{ML model}}$ → [-22.3, ...]

3. feature selection: remove unimportant features

4. dimensionality reduction

4 g. Menstr[illegible]