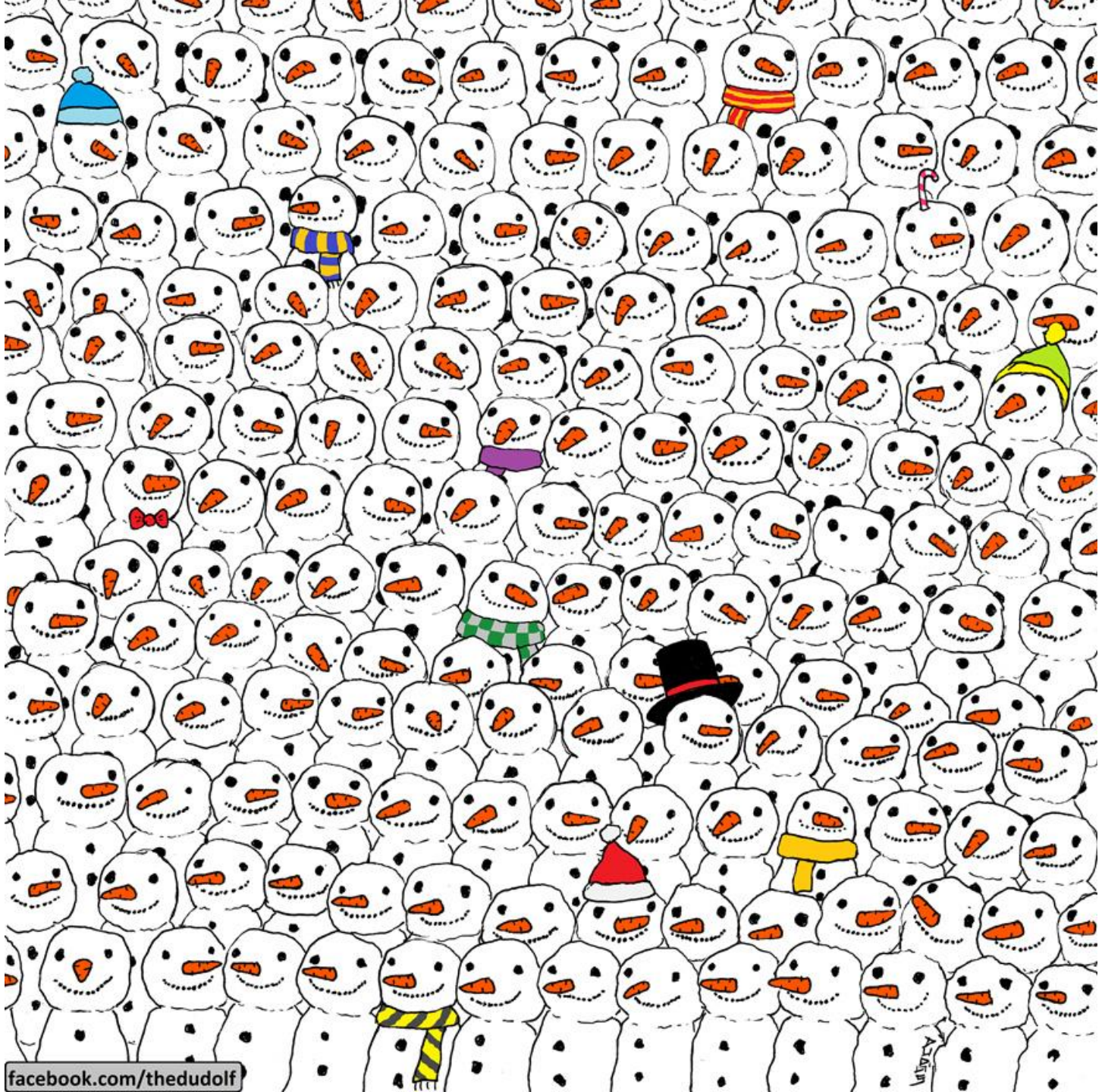
Modul 3

ML Fundamental

Data Science Program

Purwadhika
Startup and Coding School

Panda??



Cat??

Why?

- Human is really good at **detecting pattern**. However, we know the concept of **exhausting**.
- **Computer** is not born with our privileges. However, they can **work endlessly** 24/7, 7 days a week, 30 days per month, and 365 days to catch up with us.
- We can **leverage** this computer **advantages** to help us solve complex problems, which we might tired or unable to do.

Which one is which??

Purwadhika
Startup and Coding School

Which one is which??

Purwadhika
Startup and Coding School

Which one is which??

- We don’t only want to **describe** data.
- We want to **PREDICT** new data.

Purwadhika
Startup and Coding School

Machine Learning

- Able to incorporate **cumulative information,** in order to **learn** pattern form the data to predict new unseen data.
- It can **adapt** if exposed to **new data**, while human might have **subjective** bias on the decision.

Idea

- Training Set: The available dataset, usually labeled. This dataset is used to train our model.
- Test Set: Dataset independent from training set and can not be used to train model. This dataset can be unlabeled. When the labels are available, we can use it to evaluate our model.

Features

- To build our model, we need to **measure** the objects. Each of this measurement is what we call **feature**.
- Each object then is **described** by its features (collection of feature). This is what we called **features vector** approach.

Color

Weight

Age

Smell

$$\boldsymbol{X} = [\, x_1, x_2, x_3 \ldots x_p \,]$$

Purwadhika
Startup and Coding School

Datasets

Feature

Object	Color	Weight	Smell	Label
Apple 1	35	15	3	1
Apple 2	37	13	2	1
Pear 1	45	9	6	2
Pear 2	46	10	5	2

measurement

Feature Vector

Unlabeled object

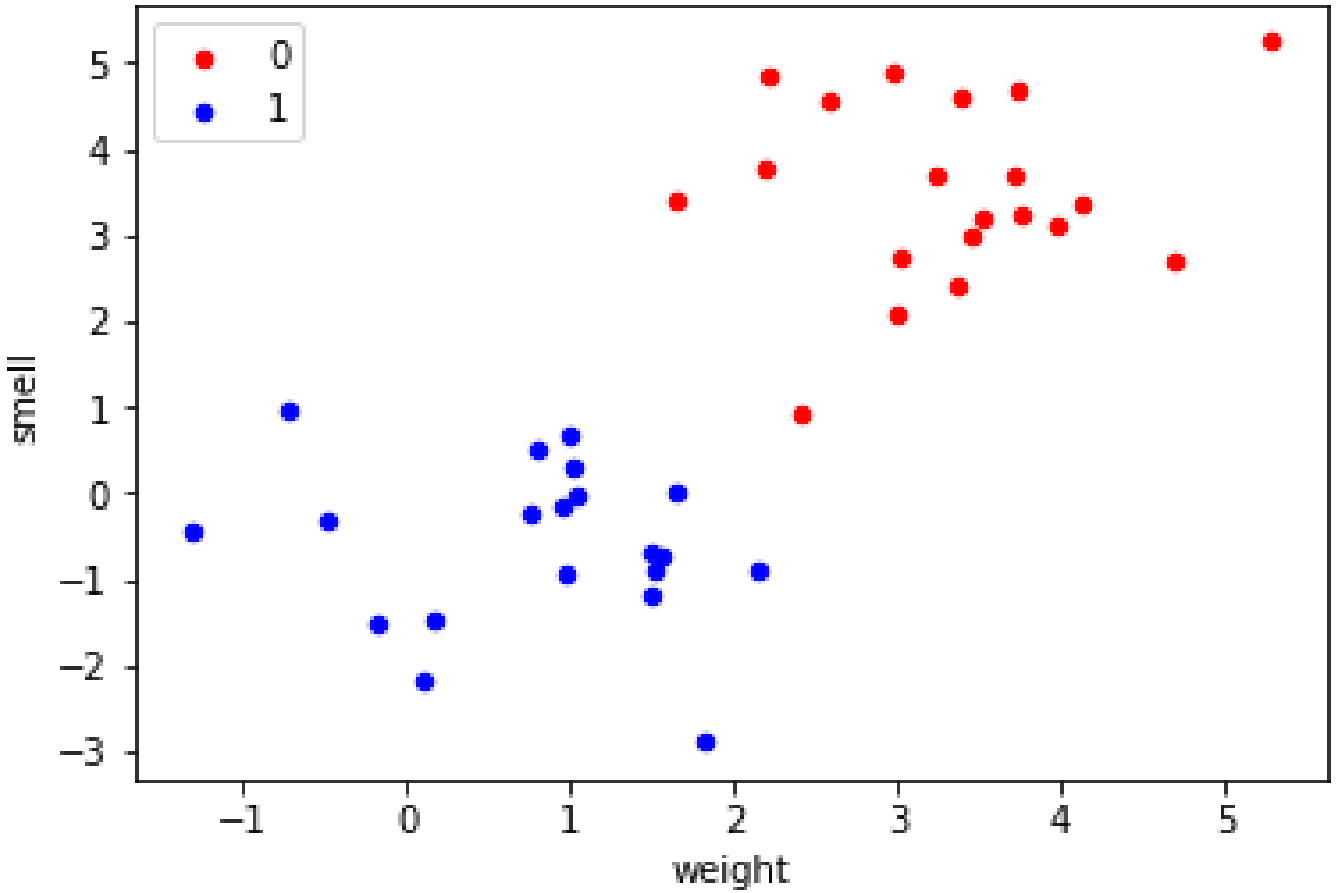
Labeled object

Purwadhika
Startup and Coding School

Other Approaches

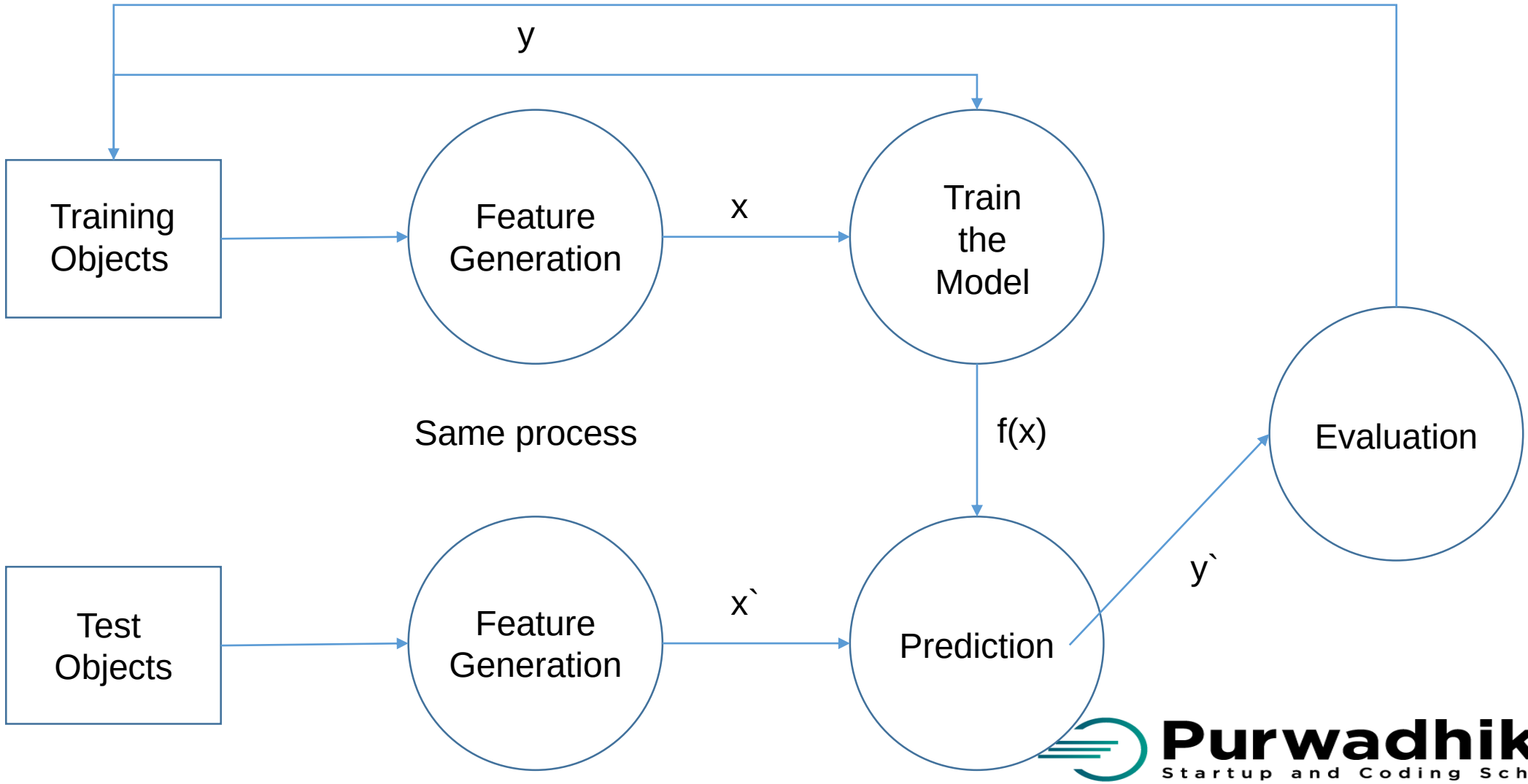
- Feature vectors approach is **not the only** approach. There are others such as Dissimilarity and Graph. However, we only discuss this approach because it is well developed.

Data Visualization



Purwadhika
Startup and Coding School

Simple Pipeline



Purwadhika
Startup and Coding School

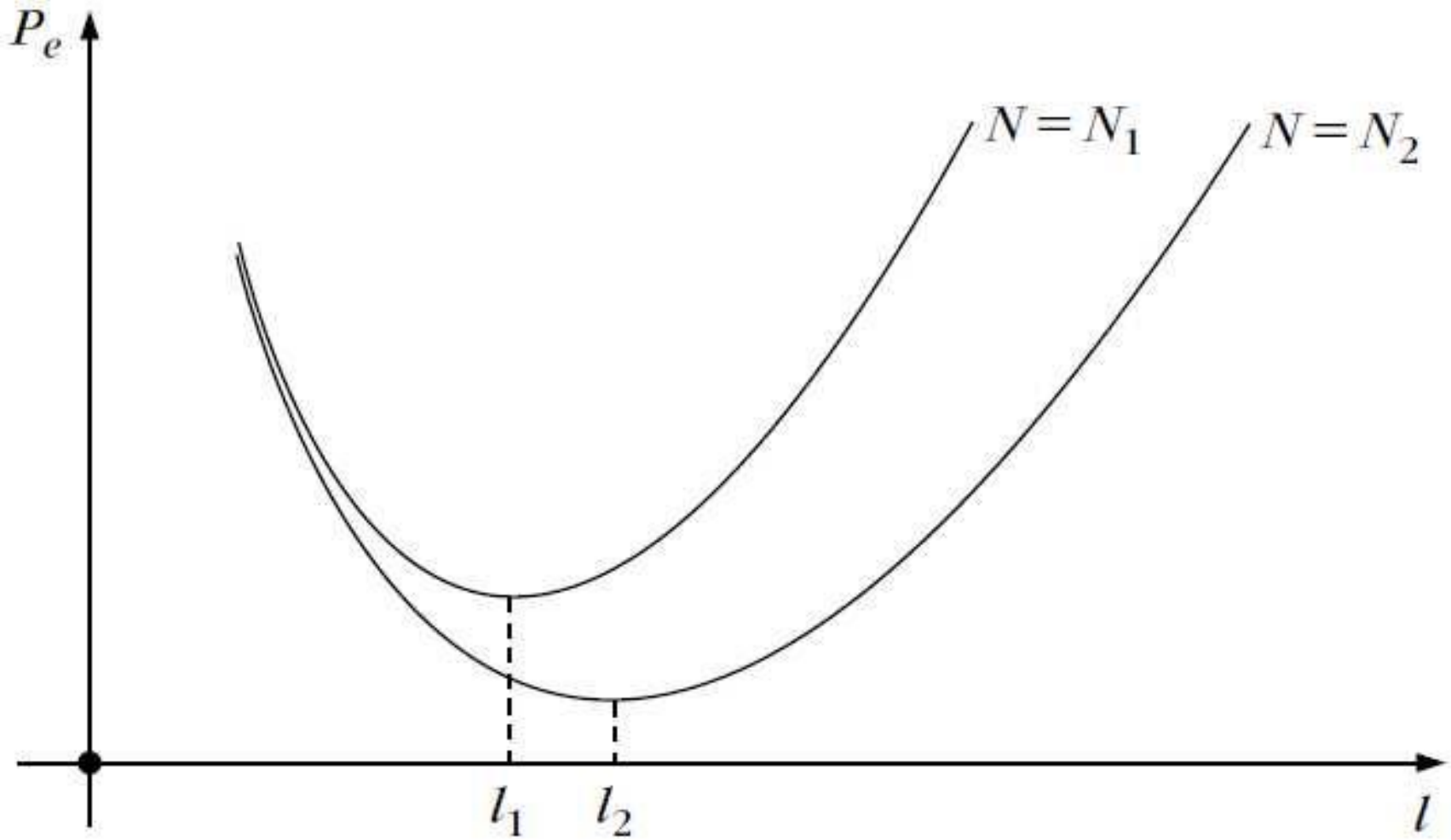
Feature Generation

- We already know that in order to make a model, we need measurements (features). Now our **goal** to make a model is to have a really good one, having a **good accuracy.**

- In the world that data is hard to obtained. Should we just do other measurements to **generate more features** to increase the accuracy?

Purwadhika
Startup and Coding School

Curse of Dimensionality



Dimensionality Reduction

- Benefit:
  - Fewer features means **faster** to obtain result.
  - Get the **important** features
  - Avoid curse of dimensionality

- How:
  - Feature Selection
  - Feature Extraction

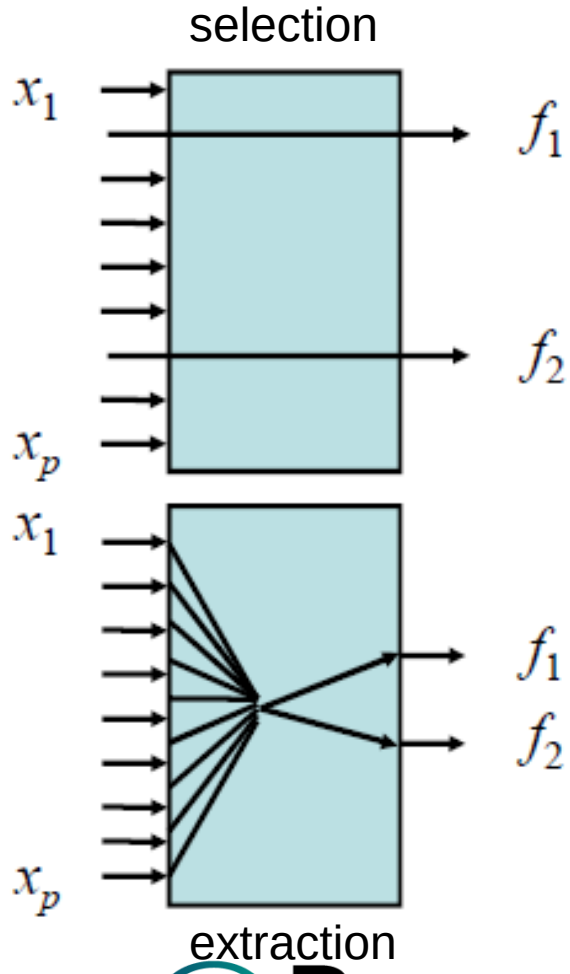
Selection vs Extraction

Feature Selection:

- Select *d* best features out of *p* features. (From the available features, only select few best features)

Feature Extraction:

- Map *d* features from *p* features.(From the available features, combine into fewer features)



Purwadhika
Startup and Coding School

Feature Selection

- What we need:
  - Criterion function: to measure between distributions.
  - Search algorithm: to pick features.

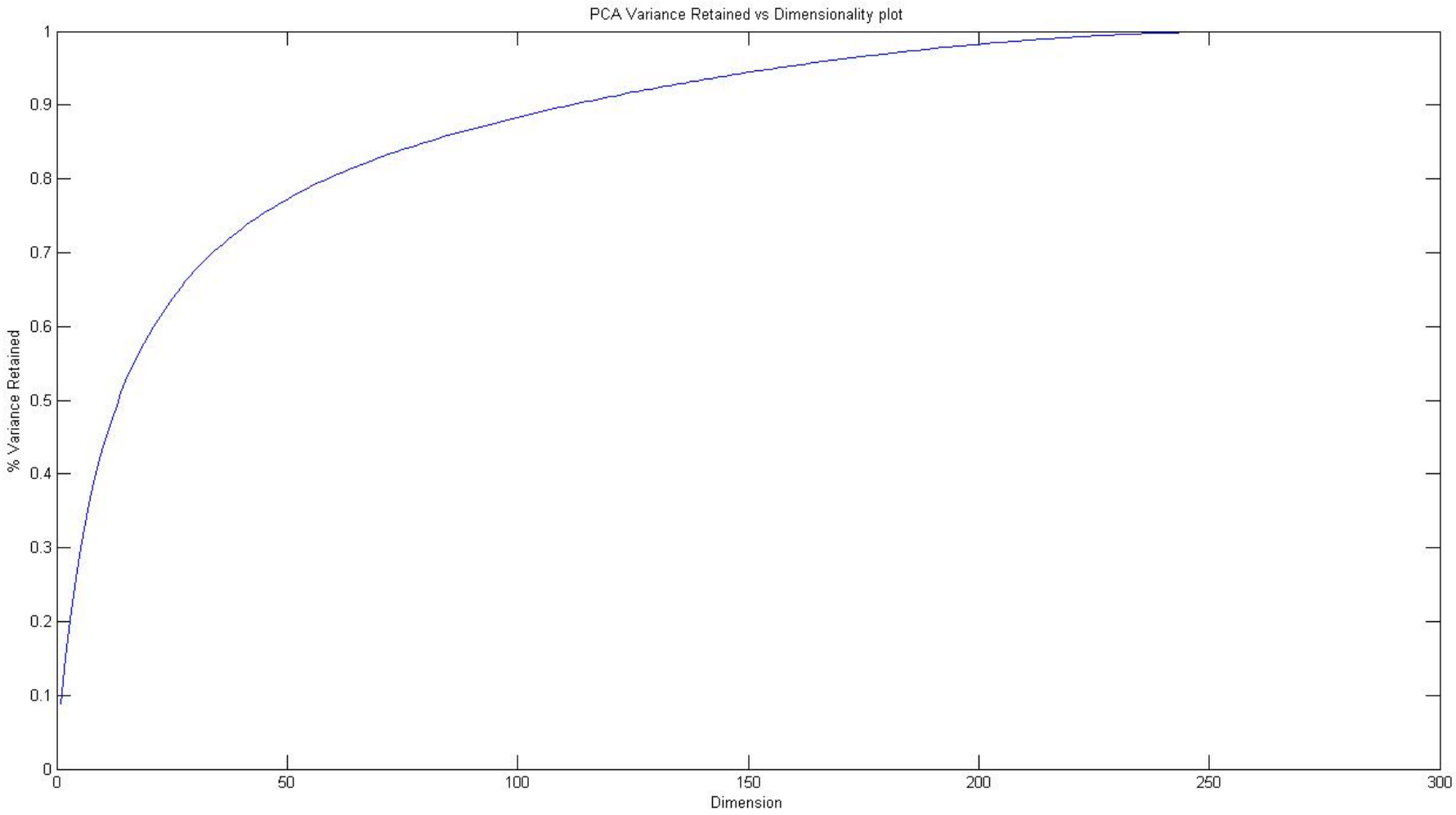
- Criterion example: Probabilistic, Scatter matrices, Mahalanobis distance.

- Search Algorithm example:
  - Forward Selection
  - Backward Selection
  - Floating Selection

Feature Extraction - PCA

- We will discuss main feature extraction techniques commonly used: **PCA**.
- PCA (Principal Component Analysis) is a technique that retains **as much variation as** possible for the data. It doesn't necessarily retain class separation.
- Goals: Obtain **principal components** that explain variation of data as much as possible. Principal components are uncorrelated variables that is the result of mapping of the original (correlate) variables.

PCA


PCA Variance Retained vs Dimensionality plot
% Variance Retained
Dimension
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
50
100
150
200
250
300

Training Model

- Supervised Learning: When you have the data and also the labels to indicate classes (target).

- Unsupervised Learning: When you only have the data, but lack of labels and you can’t determine which one belong to which class.

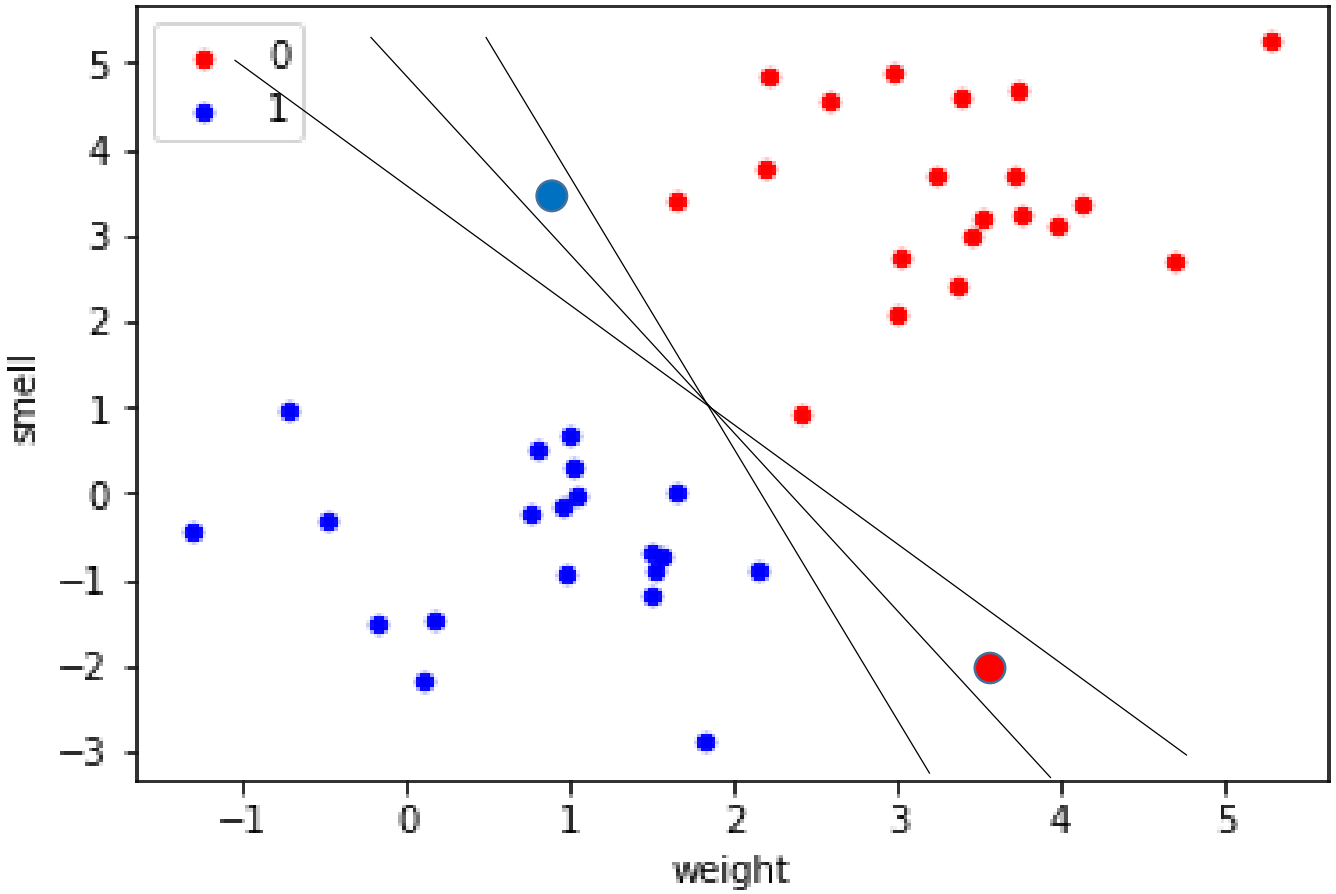
Supervised

Object	Color	Weight	Smell	Label
Apple 1	35	15	3	1
Apple 2	37	13	2	1
Pear 1	45	9	6	2
Pear 2	46	10	5	2

Supervised

- We have the **privileges** of having data that is labeled. According to experiences, it is some effort to label data from real life case.
- **Assign** each object to a **class**, we splits feature space in to separate region for each class.
- Splitting feature space region means that we have to create **decision boundaries**.

Supervised



Purwadhika
Startup and Coding School

Supervised

- Parametric: known distribution and we can estimate parameter
- Example: Nearest Mean Classifier (NMC), etc.

- Non-parametric: unknown distribution
- Example: k-Nearest Neighbor, Parzen Density Estimation, etc.

Nearest Mean Classifier

- Assume one knows the distribution (let say Gaussian (Normal Distribution)) we can estimate the **mean** and **variance** of each classes.
- The decision boundary then can be inferred: (two class problem)

$$f(x) = w^T x + w_0$$

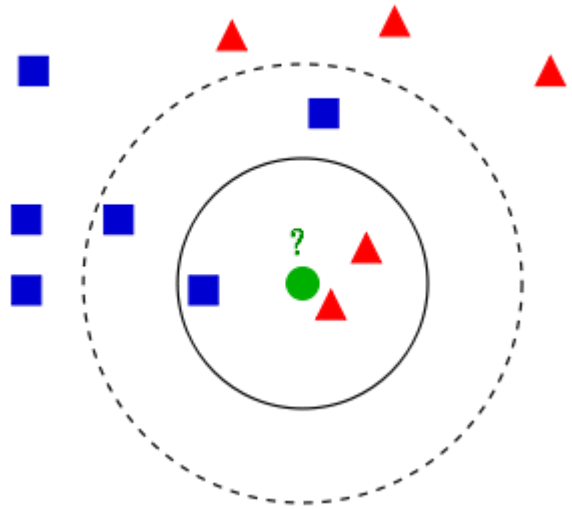
$$\boldsymbol{w} = \hat{\mu}_2 - \hat{\mu}_1$$

$$w_0 = \frac{1}{2}\hat{\mu}_2^T\hat{\mu}_2 - \frac{1}{2}\hat{\mu}_1^T\hat{\mu}_1 + \sigma^2 \log\frac{p(\omega_1)}{p(\omega_2)}$$

Purwadhika
Startup and Coding School

k-Nearest Neighbor

- Estimation density is **hard**. Instead we can use the approach of picking *k*, user-defined constant, and label the object based on the **most frequent** class **member** among k nearest training object.
- On two class classification problem, the k-value should be **odd** to avoid tie. It can be optimized using several techniques that we won't discuss here.



Purwadhika
Startup and Coding School

Supervised

- Linear: decision boundary is linear
- Example: NMC, Perceptron, SVM, etc.

- Non-linear: decision boundary is non-linear
- Example: SVM polynomial kernel, Quadratic classifier, etc.

Linear Classifier

Dot product of two vectors (w.x). T is just a transpose

$$\mathbf{w} \cdot \mathbf{x} = \mathbf{w}^{\mathbf{T}}\mathbf{x} = \sum_{i=0}^{n} w_i * x_i$$

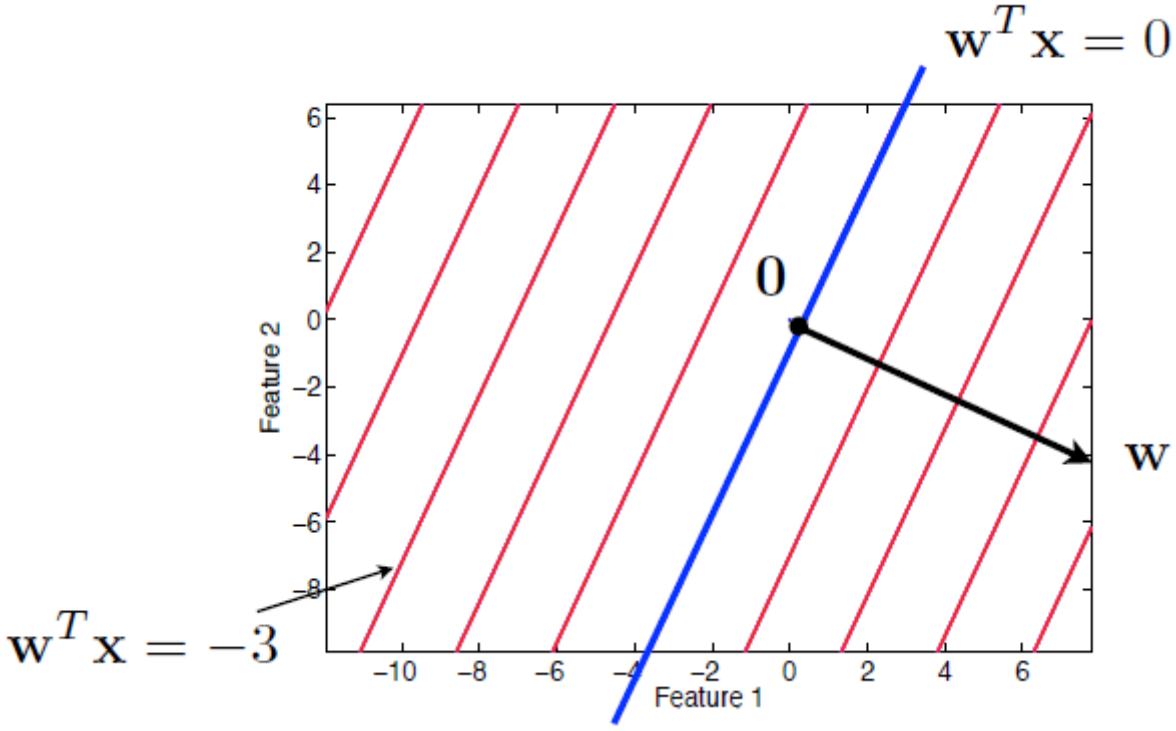
- Assume our decision boundary is $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 = 0$

$$\text{classify } \mathbf{x} \text{ to } \begin{cases} \omega_1 & \text{if } \mathbf{w}^T\mathbf{x} + w_0 \geq 0 \\ \omega_2 & \text{if } \mathbf{w}^T\mathbf{x} + w_0 < 0 \end{cases}$$

What do we mean by w^Tx? What is w_0?

Purwadhika
Startup and Coding School

Linear Classifier



How do we find **w** & w_0?

Purwadhika
Startup and Coding School

Perceptron

- Perceptron is algorithm for binary classifier or known as Linear Binary Classifier that based on human biological neural node
- For simplicity, lets assume that there are two input values, x and y for a certain perceptron P. Let the weights for x and y be A and B for respectively, the weighted sum could be represented as: A x + B y.
- Perceptron outputs a non-zero value only when the weighted sum exceeds a certain threshold C, one can write down the output of this perceptron as follows:

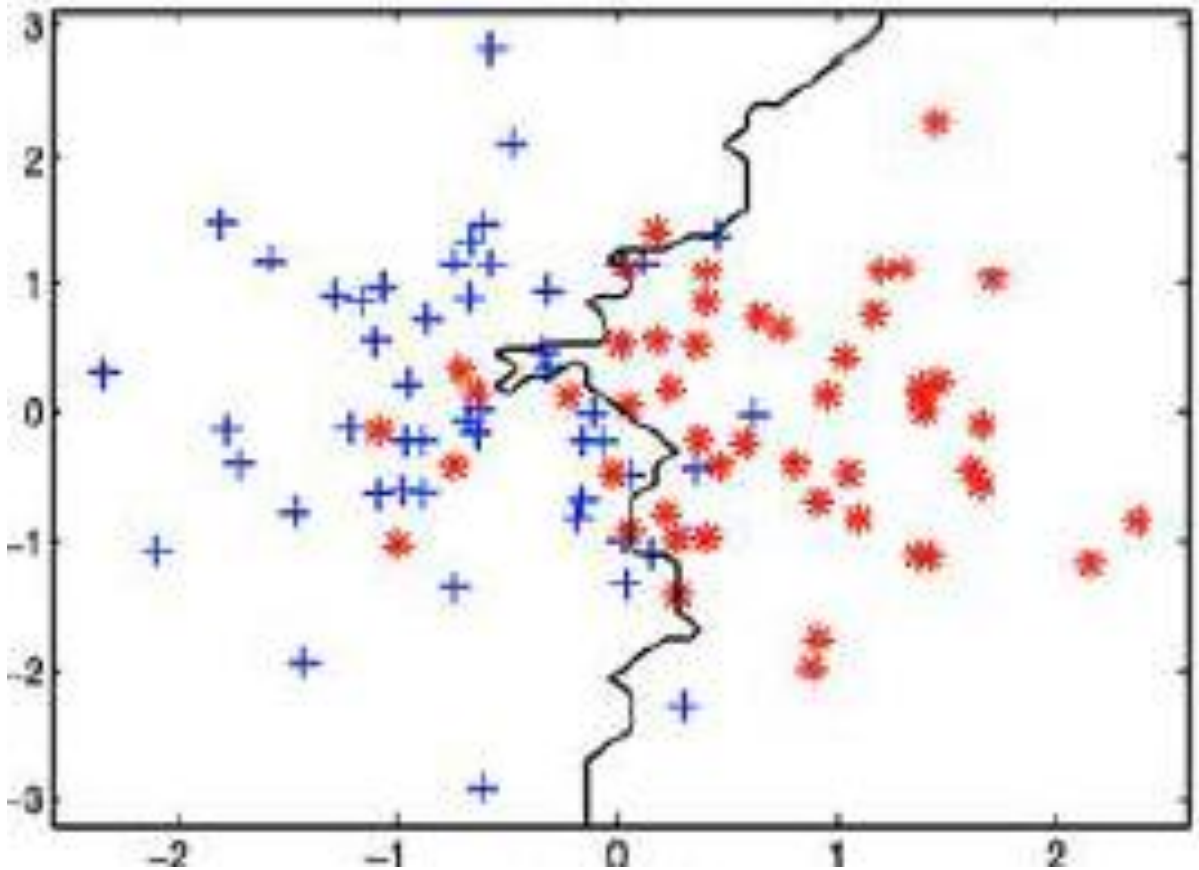
Output of P = {1 if A x + By > C

{0 if A x + By <= C

Non-Linear Classifier

- Why non-linear? Many problems can’t be solved by a mere linear classifier. Non-linear classifier introduce flexibility but also complexity.
- I won’t discuss here the normal non-linear classifier such as quadratic classifier, 2-layer Perceptron (NN) and such. However, we can do some trick to make existing linear classifiers become non-linear.
- You can choose to incorporate **kernel** to many classifiers, for example SVM (Support Vector Machine). For example polynomial kernel or gaussian kernel.

Non-Linear Classifier



Purwadhika
Startup and Coding School

Supervised Use Case

- For its new fintech company, Astra want to have a model that can predict whether a person will be able to return the lending money on time or not. Astra can have the data source from previous motor industry credit.

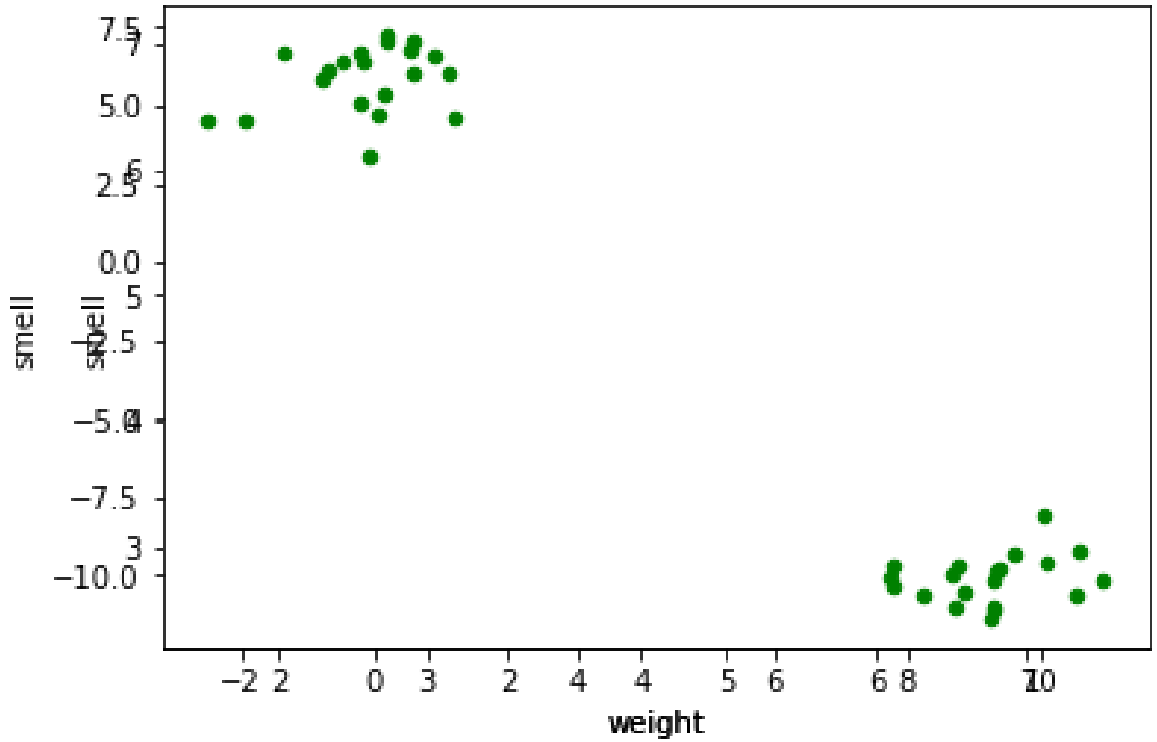
- Astra wants to boost its car division sales by using targeted sales / advertisement based on profile. You can create a model to predict which model suitable for which user profile by using past sales.

Unsupervised Classifier

Object	Object	Color	Color	Weight	Weight	Smell	Smell	Label
Apple 1	Object 1	35	35	15	15	3	3	1
Apple 2	Object 2	37	37	13	13	2	2	1
Pear 1	Object 3	45	45	9	9	6	6	2
Pear 2	Object 4	46	46	10	10	5	5	2

Purwadhika
Startup and Coding School

Unsupervised Classifier



Purwadhika
Startup and Coding School

Unsupervised

- We **don’t have** the **label** of data. Either we start labelling it now, or we should find other approaches.
- Sometimes, labeling the data itself **can be tricky**. It can introduce personal bias, takes a lot of time, etc.
- Or we can try **clustering**. Approach on grouping data that are closer together. But how we define **close together** or **far apart**?
- We can use (dis)similarity measure to define it.

k-Means Clustering

- Goals: cluster data into **k** cluster.
- Each cluster is represented by **prototypes**.
- Use **dissimilarity** (squared Euclidean distance) of prototypes to group data.
- **Start with random** prototypes and **iteratively** choose others until convergence. (There are techniques to choose better prototypes)
- Produce **local maxima** depend on initial prototypes.

k-Means Clustering

- How to do:
  1. Choose the desired number of cluster **k**
  2. Choose initial prototypes
  3. For each objects, compute distance to prototypes to label the data.
  4. Compute cluster means as the new prototypes.
  5. Repeat step 3 & 4 until there is no change in prototypes.

k-Means Clustering

- What is the best **k** number to be used? We could try the elbow method to determine the best **k**

Unsupervised Use Case

- For its new maintenance company, Astra want to have a model that can predict whether a car is on good condition or not based on images. Astra does not have this kind of data as they have only the images. So what they can do is cluster images that *similar* to determine group of condition.

- Astra wants to detect customer behavior but do not know what are differences between behavior so they can determine the labels. What you can do is group people that *similar* together and go from there.

Recap

- We now know about the basic of machine learning, on constructing dataset, the simple pipeline, and how to acquire the *best* features.
- We understand the differences between supervised and unsupervised learning.
- We have knowledge on the differences between parametric and non-parametric, as well as between linear and non-linear for supervised learning.
- We know how to cope with unlabeled data, especially on training clustering model using k-means clustering.

Purwadhika
Startup and Coding School