Making Our Data Ready for the ML Model

Mohammed Osman
SENIOR SOFTWARE DEVELOPER

@cognitiveosman www.cognitiveosman.com

Overview

Pipeline once again

Continuing on data preparation

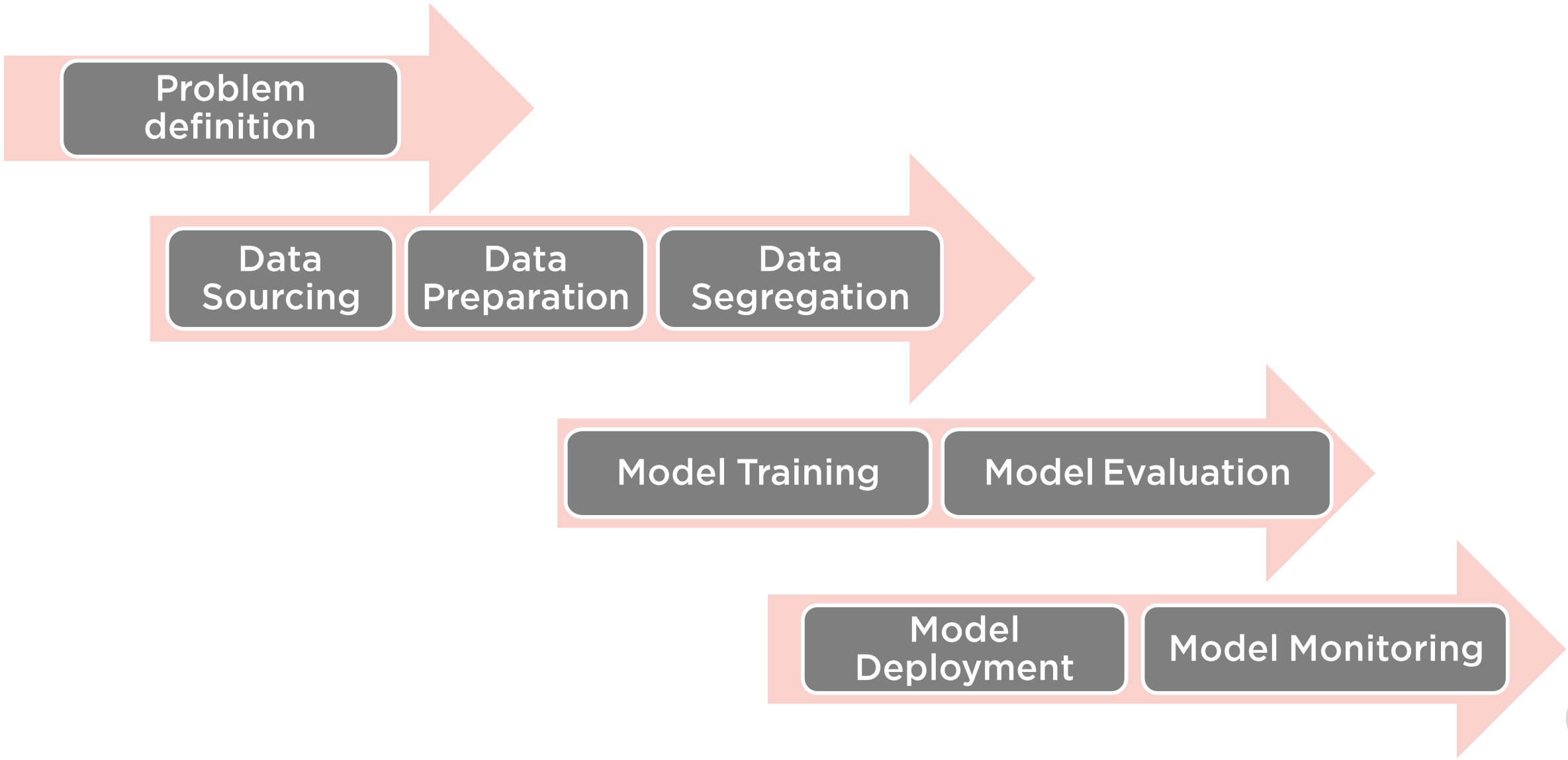
- Data scaling

Data segregation

Sci-kit learn

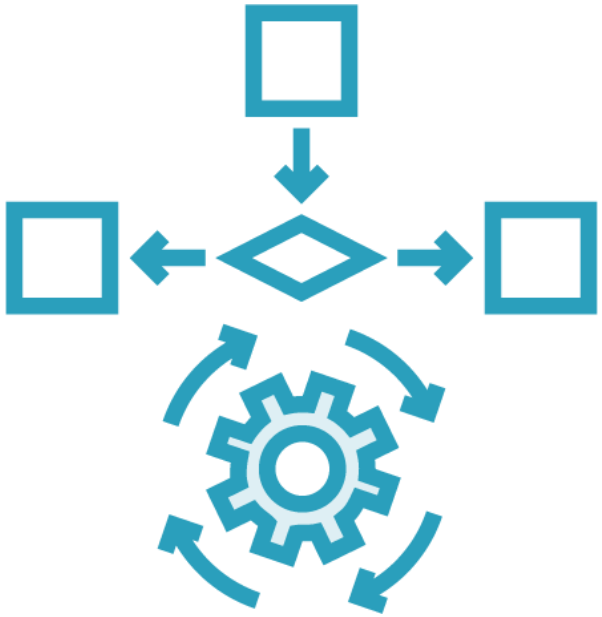
Demos

Data Segregation



Data Preparation: Data Scaling

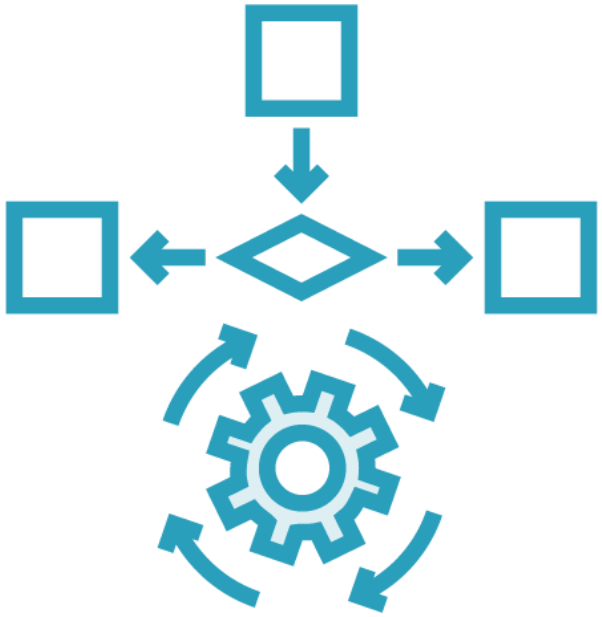
The Need for Data Scaling



Weight kg or lb
Length cm or inch
Duration second,minute or hour


LOCATION
SOUND
DATE
TAKE
SCENE
DIRECTOR

The Need for Data Scaling

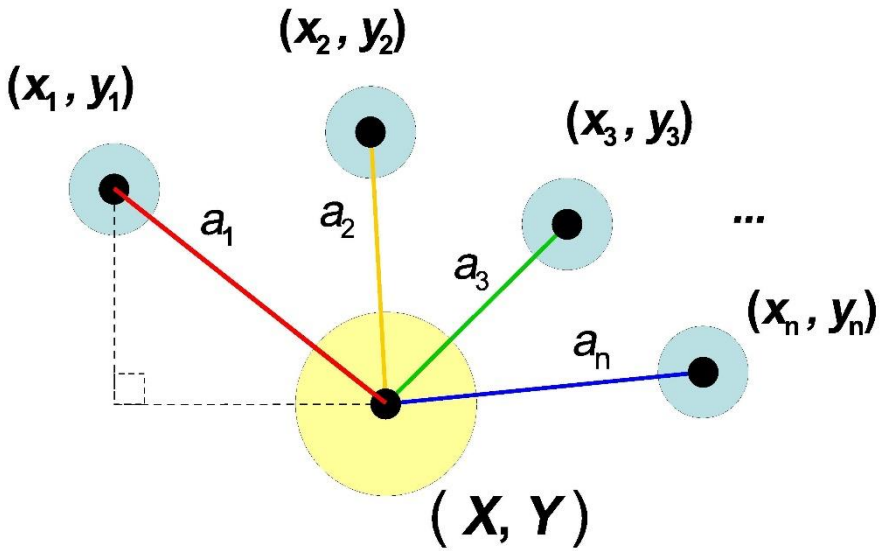


Weight kg or lb
Length cm or inch
Duration second,minute or hour

Normal distribution
Euclidean distance

Euclidean Distance

Is the distance between two points in an **Euclidean** space

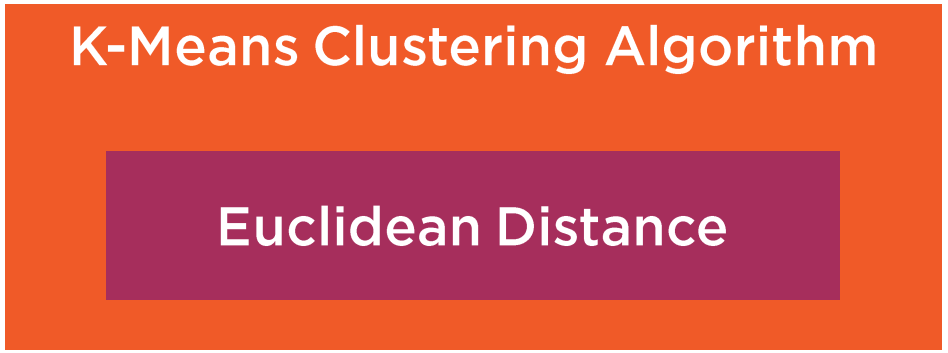


$$a_i = \sqrt{(x_i - X)^2 + (y_i - Y)^2}$$

Euclidean distance is defined as

$$a_i = \sqrt{(x_i - X)^2 + (y_i - Y)^2}$$

K-Means Clustering and Data Scale

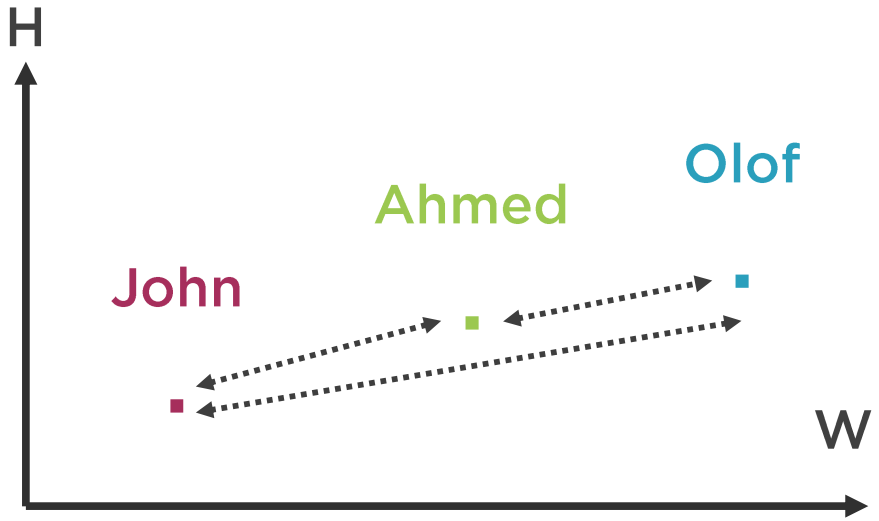


People Dataset

Name	Height"	Height (cm)	Weight
John	63	160	150lb
Ahmed	67	170.2	160lb
Olof	70	177.8	171lb

Source: Data Science from Scratch (P132)

K-Means Euclidean Distance Calculation



Distance Calculation

Name	Height″	Height (cm)	Weight	(H″,W)	(H (cm),W)
John	63	160	150lb	(63,150)	(160,150)
Ahmed	67	170.2	160lb	(67,160)	(170.2,160)
Olof	70	177.8	171lb	(70,171)	(177.8,171)

Distance when height is in inches

John and Ahmed: 10.77
John and Olof: 22.14
Olof and Ahmed: 11.4

Distance when height is in centimeters

John and Ahmed: 14.28
John and Olof: 27.53
Olof and Ahmed: 13.37

Euclidean Distance is **affected** by the magnitudes of the input dataset, and since conversion units (e.g. inch to cm) **changes** the magnitude, Euclidean Distance results *will* change

Eliminating Scale Effect

Data Scaling

Standardization

Removing the mean and scaling to unit variance

MinMax Scaling

Rescaling all attributes to range between zero and one

Normalization Scaling

Rescaling each **observation (row)** to unit value

MinMax Scaling Distance Calculation

$$X_{Scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

MinMax Scaling Distance Calculation

$$X_{Scaled} = \frac{X_{min} - X_{min}}{X_{max} - X_{min}} = \text{zero}$$

MinMax Scaling Distance Calculation

$$X_{Scaled} = \frac{X_{max} - X_{min}}{X_{max} - X_{min}} = 1$$

MinMax Scaling Distance Calculation

$$X_{Scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

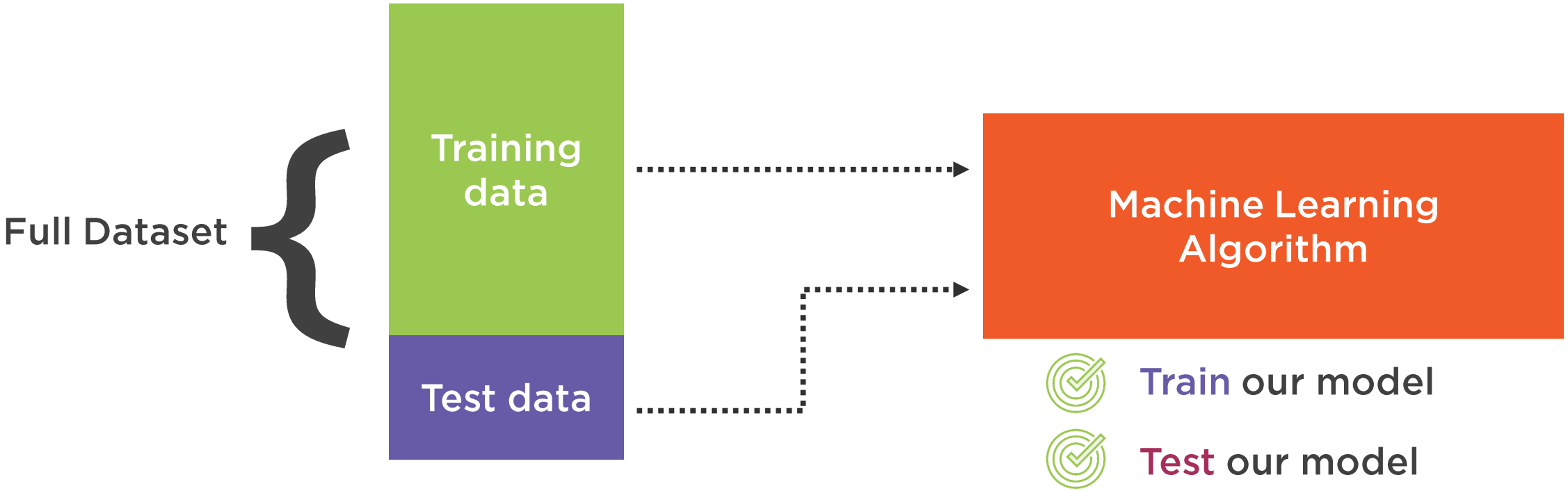
Name	Height″	Height (cm)	Weight	Scaled Height″	Scaled Height (cm)
John	63	160	150lb	0	0
Ahmed	67	170.2	160lb	0.57	0.57
Olof	70	177.8	171lb	1	1

$$X_{Scaled} = \frac{(X - X_{min}) * \cancel{unit}}{(X_{max} - X_{min}) * \cancel{unit}}$$

As a rule of thumb, always scale your data when the underlying algorithm calculates distance

Data Segregation

Why Data Segregation?

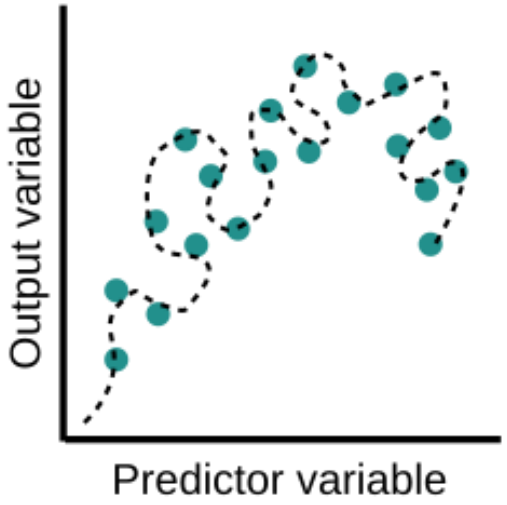


How should we choose training and test data?

How to randomize?

How big to split?

Why Not to Train/Test on the Whole Dataset?



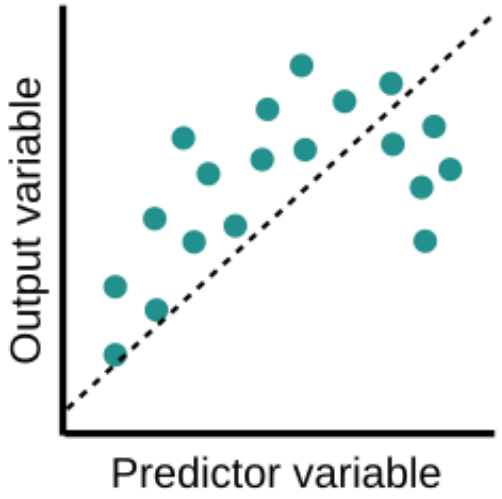
Training and testing on the same set can result in overfitting

Think of it as testing a student with the same tutorial questions

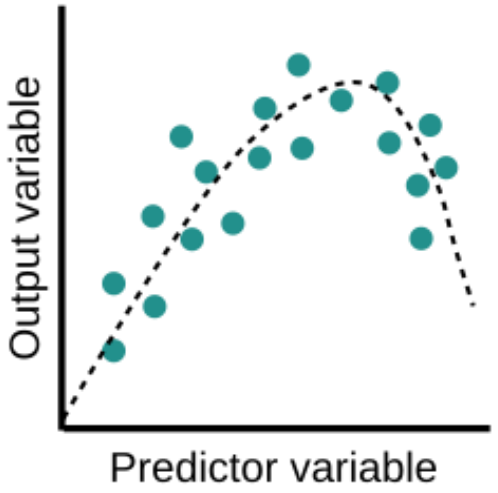
Source:
http://bit.ly/338ozjK

Underfitting and Fitting

Underfitting



Fitting



Source:
http://bit.ly/338ozjK

Data Segregation Techniques

Train/Test Split

K-Fold Cross Validation

Train/Test Split

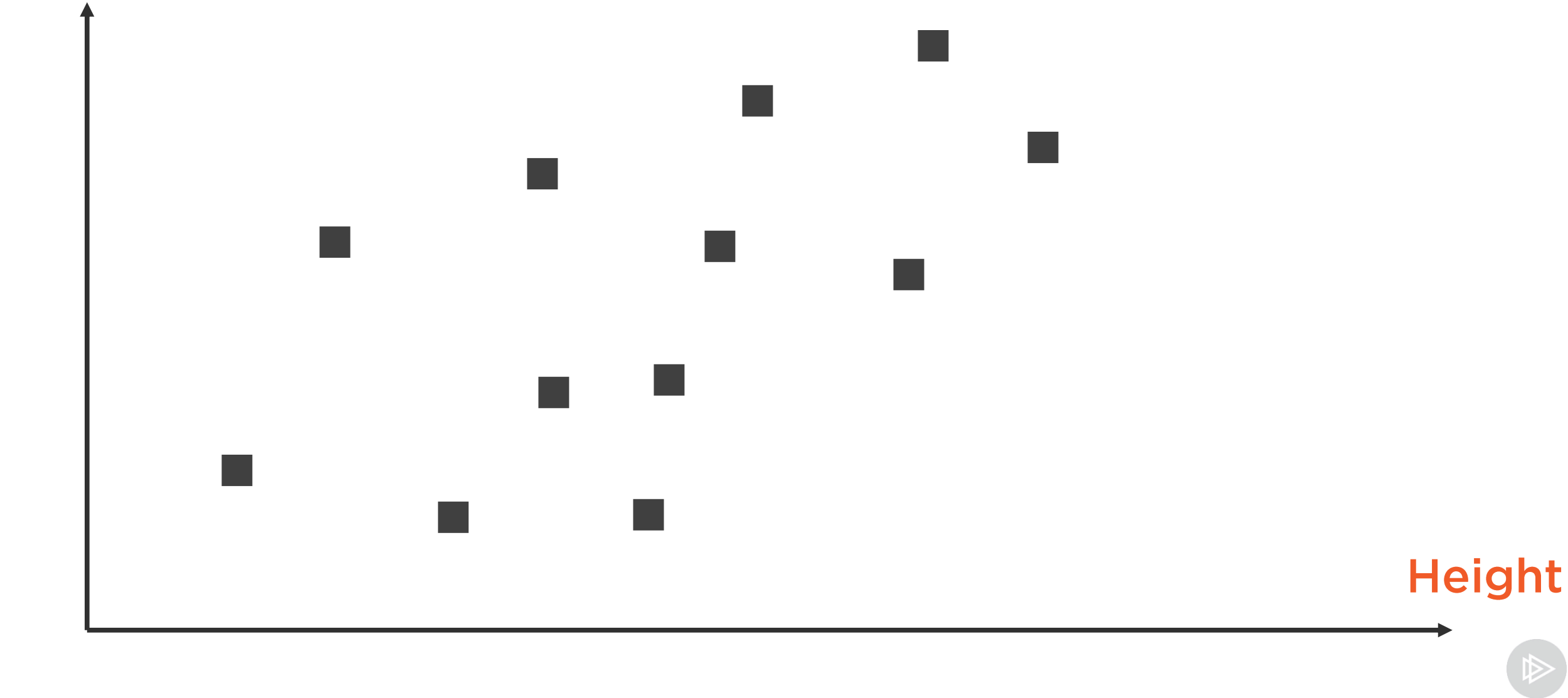
70% to 80%

Training Set

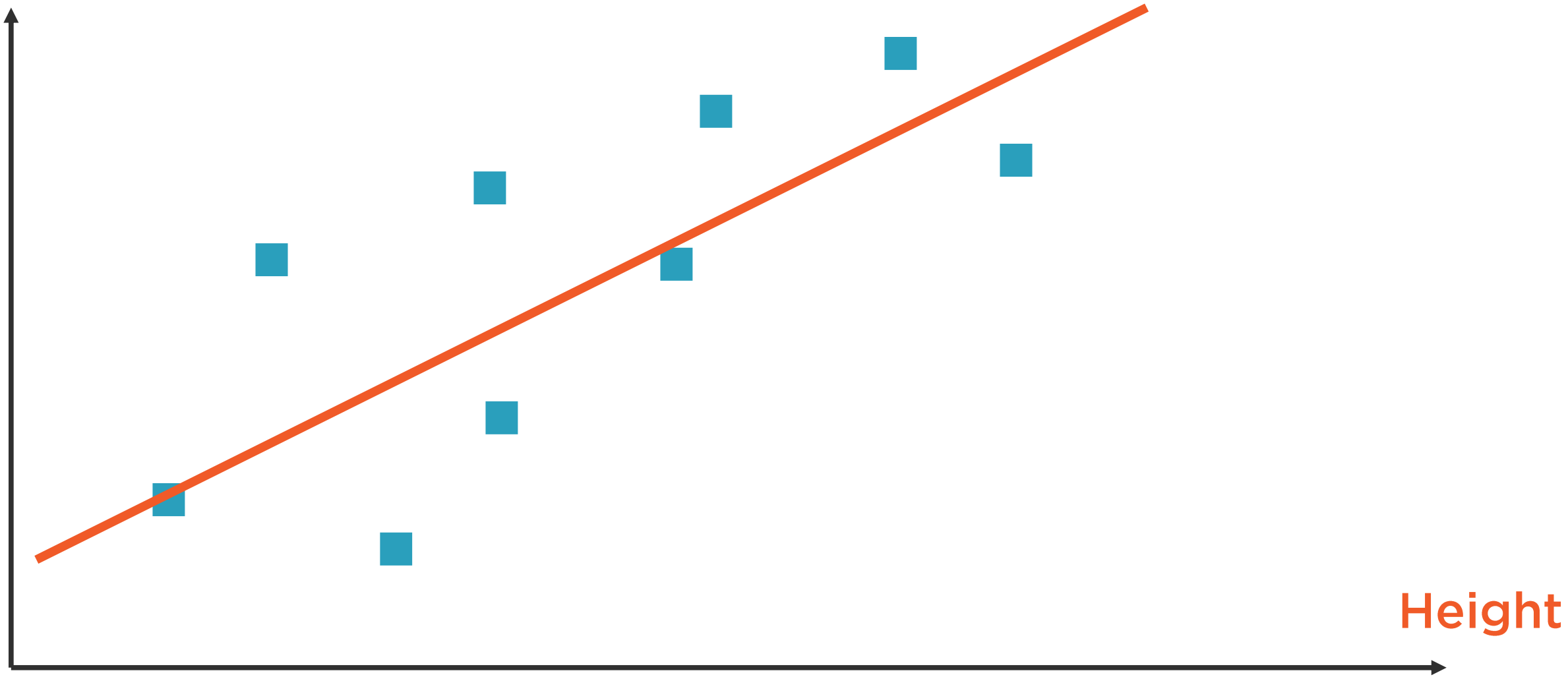
20% to 30%

Test Set

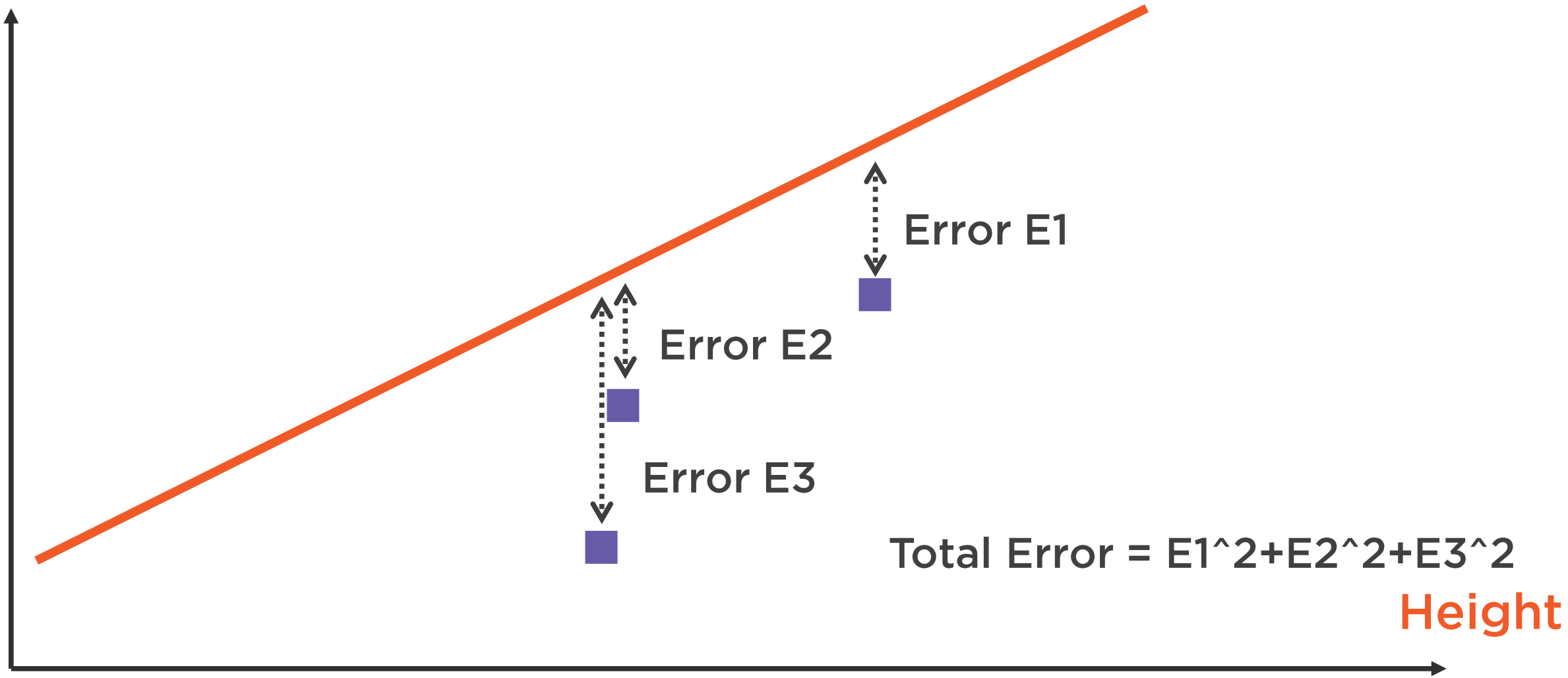
Train/Test Split Explained



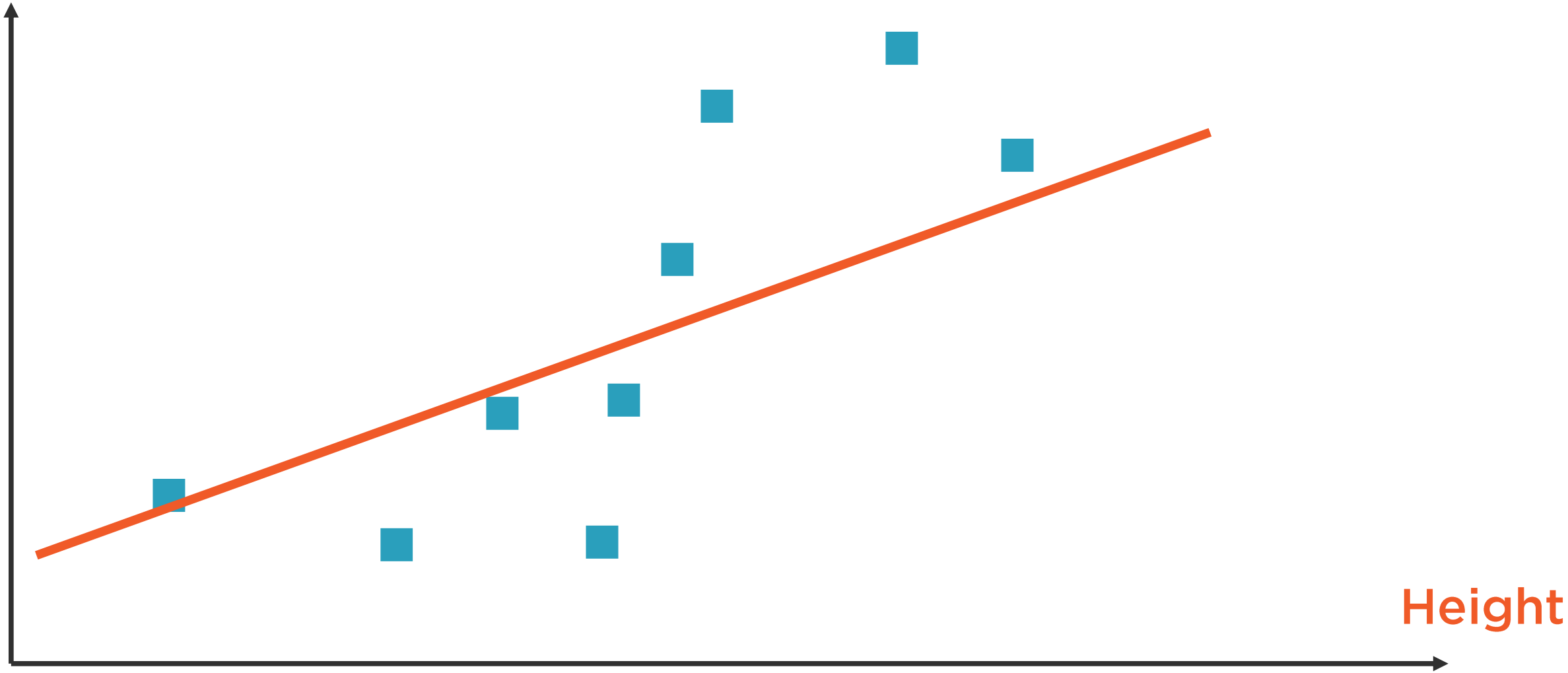
First Combination: Training



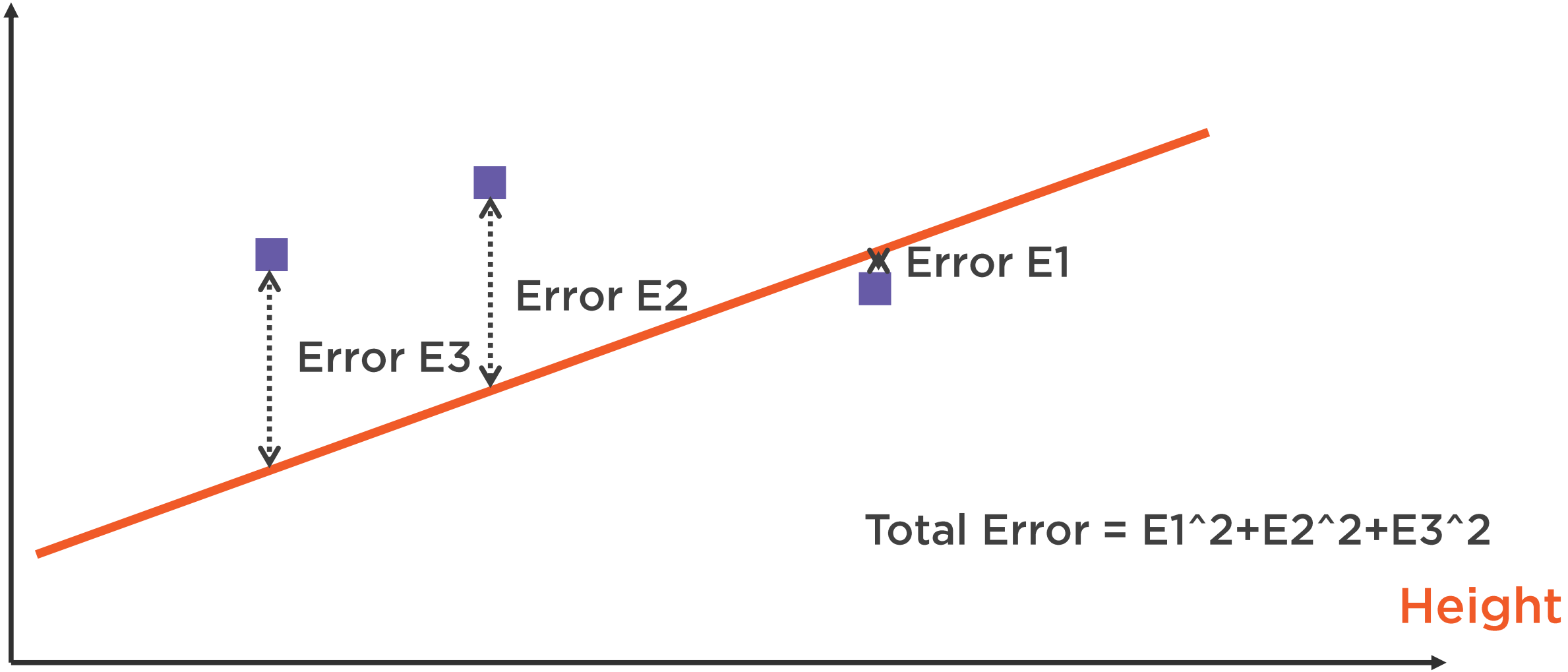
First Combination: Testing



Second Combination: Training



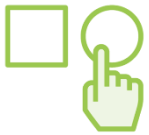
Second Combination: Testing



As you noted: The error from the first chosen train/test combination is *different* from what we have got from the second train/test combination – NOT GOOD!

K-Fold Cross Validation

Split the dataset to K groups (folds)



Choose one group as a test set and others as training set

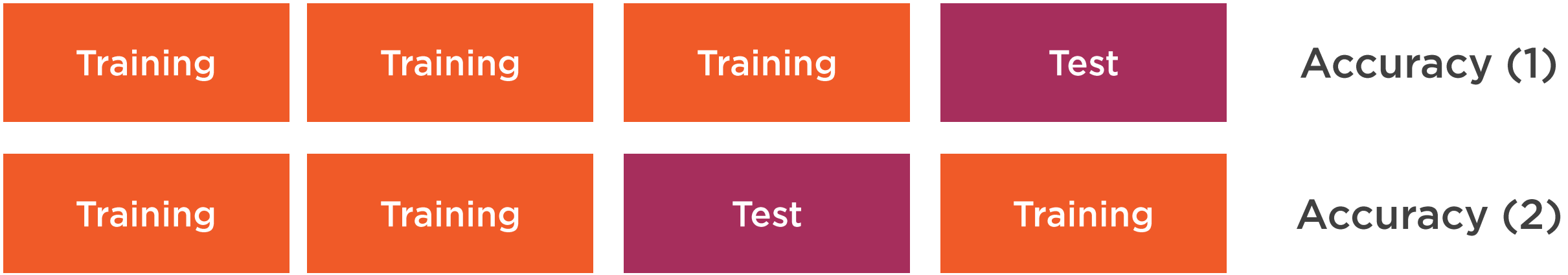
Train and **calculate** the accuracy

Choose next group as a test set and repeat

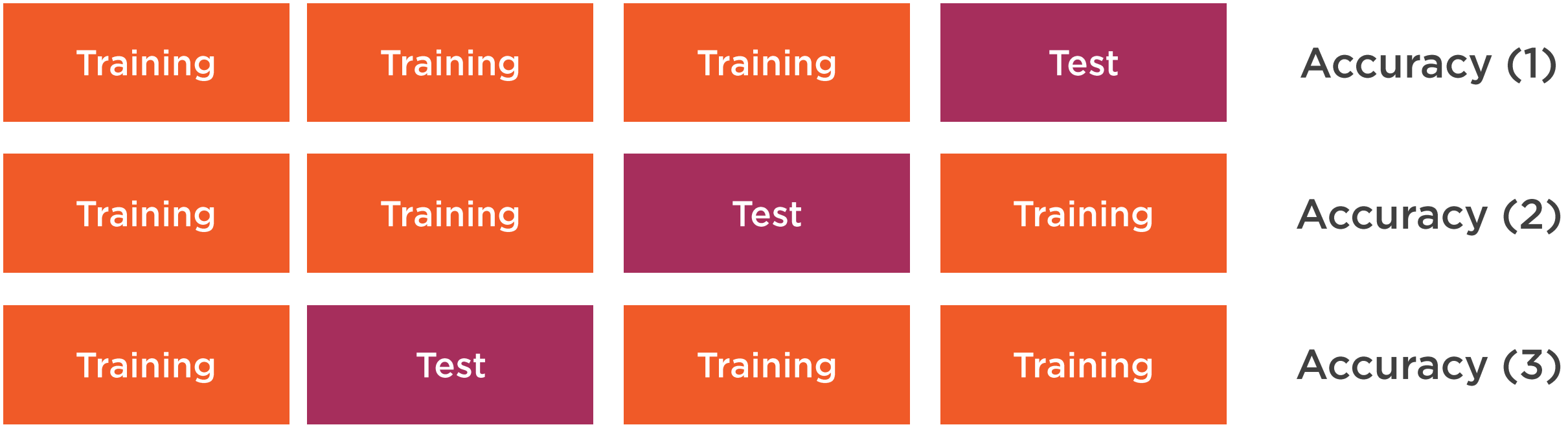
Calculate average accuracy from all training rounds

First Round: 4-Fold Cross Validation

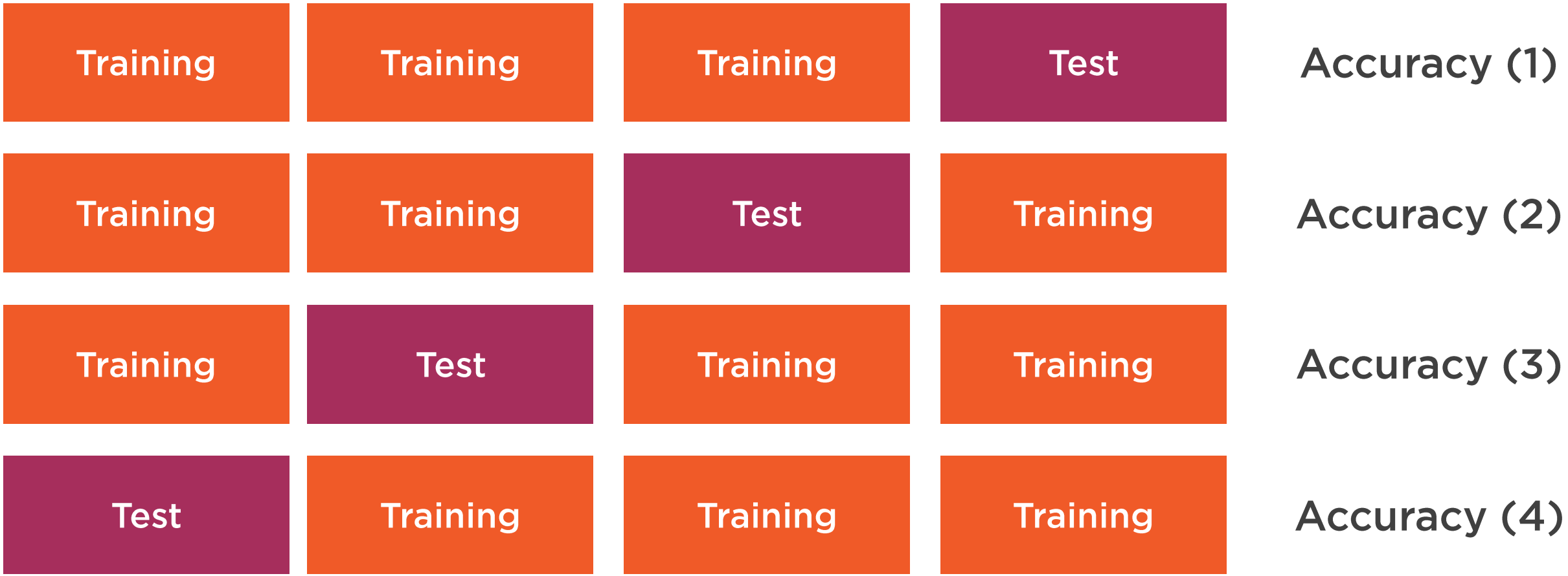
Second Round: 4-Fold Cross Validation



Third Round: 4-Fold Cross Validation



Fourth Round: 4-Fold Cross Validation



Model Accuracy = Avg (Accuracy (1), Accuracy (2), Accuracy (3), Accuracy (4)

Key Considerations with Data Segregation

Use rule-of-thumb numbers

Train/Test: Pareto Principle!

Cross Validation: K = 10

Randomize your dataset

Adjacent records tend to have selection bias!

Cross Validation vs Train/Test

Cross Validation is more accurate but slower

Train/Test is faster but less accurate

Understanding scikit-learn

scikit-learn

scikit-learn

Open source machine learning, data mining and data analysis library

Built on NumPy, SciPy and matplotlib

Home for ML algorithms

```
from sklearn.model_selection
import train_test_split

from sklearn.model_selection
import KFold


X_train, X_test, y_train, y_test =
train_test_split(X, Y,shuffle=True
random_state=4)


kf =
KFold(n_splits=4,shuffle=False)
.split(range(16))
```

- Importing train/test split function
- Importing K-Fold cross validation function
- Separating our dataset into training and testing sets with randomization
- Creating 4-Fold cross validator and applying it on array with values (0 to 15)

Demo

Data Segregation

- Train/Test split
- K-Fold Cross Validation

Summary

Another round with ML pipeline

Data Scaling

- Why
- How

Data Segregation

- Why
- How

scikit-learn

Demo