EN3150 Pattern Recognition
Learning from data and related challenges

M. T. U. Sampath K. Perera,
Department of Electronic and Telecommunication Engineering,
University of Moratuwa.
(sampathk@uom.lk).
Semester 5 – Batch 20.

What is learning ?

"A computer program is said to learn from experience **E** with respect to some class of tasks **T** and performance measure **P,** if its performance at tasks in **T**, as measured by **P**, improves with experience **E** "[1]
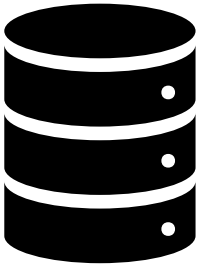
experience **E**

performance measure **P**

tasks **T**

performance at tasks in **T**, as measured by **P**, improves with experience **E**

[1] Mitchell, Tom M. "Machine learning." (1997).

Learning from data

data preparation
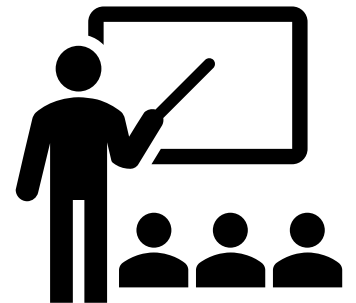
Model training

Model Evaluation

➢ There are different types of learning from data:

➢ Supervised

➢ Unsupervised

➢ Semi-supervised

➢ Reinforcement learning

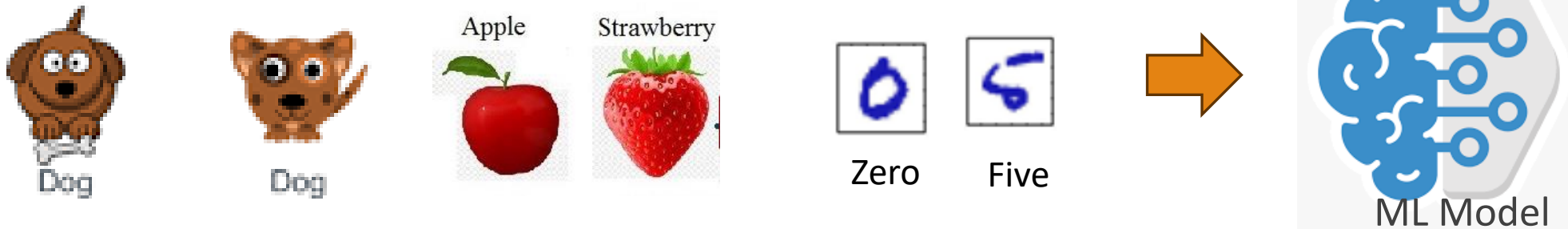# Learning from data: Supervised learning

➢Supervised learning:
  o The algorithm learns <mark>from labeled training data to make predictions or decisions</mark>.

➢ Labeled training data?

  o The training data consists of input examples (also called features) along with their corresponding output labels (also called targets or ground truth).

➢The goal of supervised learning is to learn a mapping function that can predict the correct output label for new, unseen input examples.
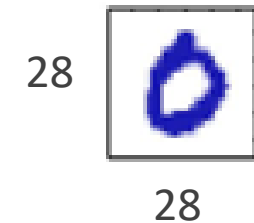
Dog   Dog   Apple   Strawberry

                                        Zero   Five

ML Model

Learning from data: Supervised learning

➢ Labeled training data

    ➢ Handwritten digit - MNIST dataset

Zero   Five

ML Model
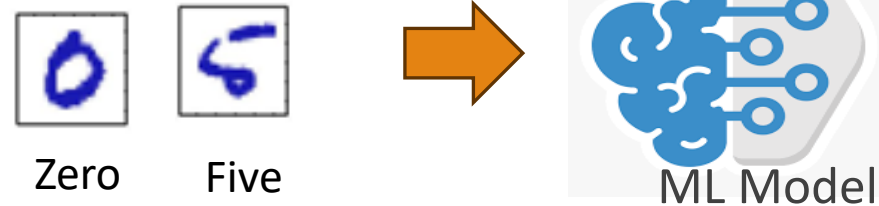
MNIST dataset
28x28 pixel images of handwritten digits (0 to 9) along with their corresponding labels.

28

28

	0	1	...	781	782	783	label
0	0.0	0.0		0.0	0.0	0.0	5
1	0.0	0.0		0.0	0.0	0.0	0
2	0.0	0.0		0.0	0.0	0.0	4
3	0.0	0.0		0.0	0.0	0.0	1
4	0.0	0.0		0.0	0.0	0.0	9

Learning from data: Unsupervised learning

➢ Unsupervised learning involves training an algorithm on unlabeled data without explicit output labels

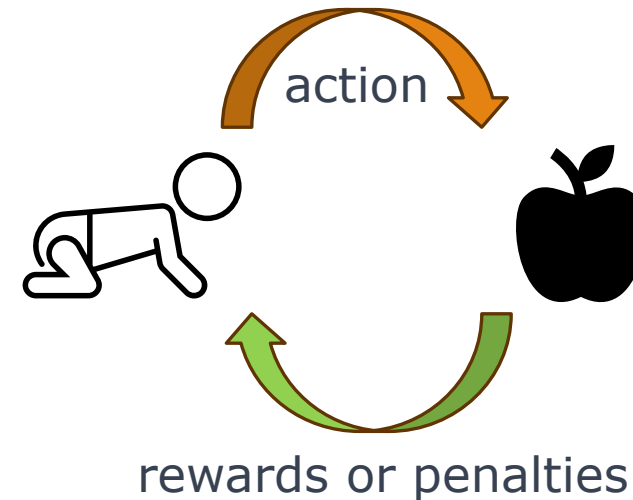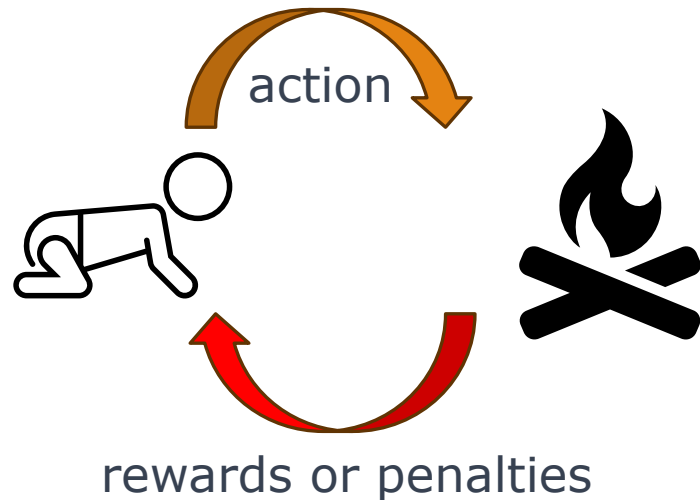➢The algorithm's objective is to find patterns, structures, or relationships within the data.
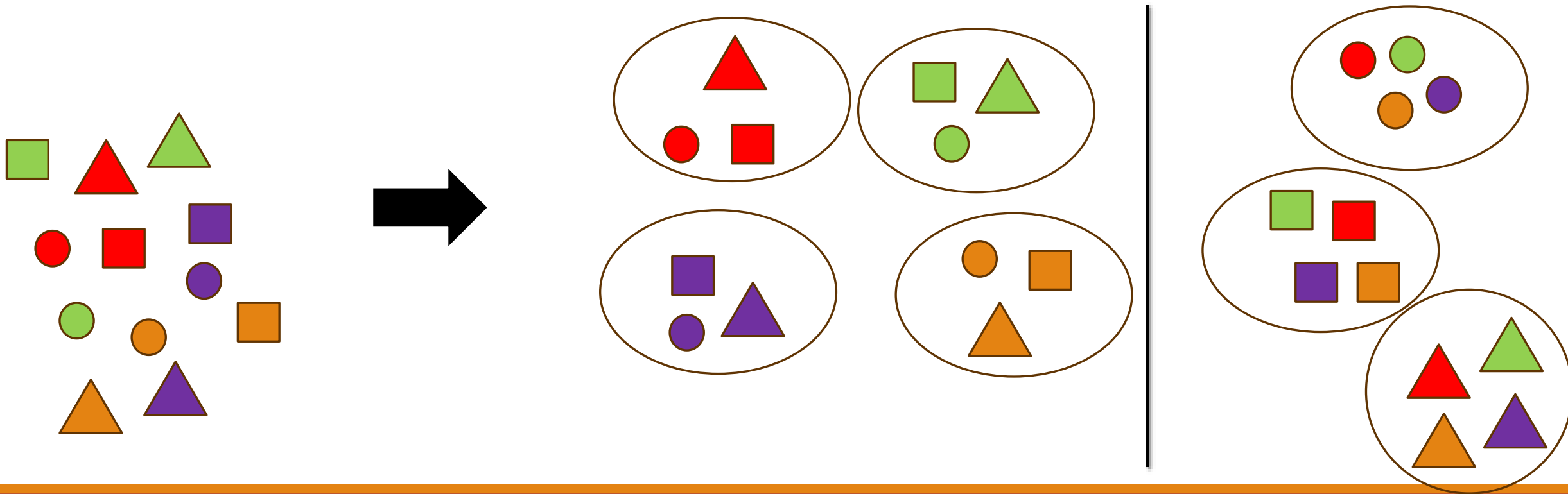
Learning from data: Semi-Supervised Learning

➢Semi-supervised learning: supervised + unsupervised learning.

➢Training data contains a mixture of labeled and unlabeled examples.

Learning from data: Reinforcement Learning

➢An agent learns to make decisions by interacting with an environment.

➢The agent receives feedback in the form of rewards or penalties based on its actions.

➢The goal of the agent is to learn an optimal policy that maximizes the cumulative reward over time.

action

rewards or penalties

action

rewards or penalties

Supervised vs Unsupervised learning

| Supervised	Unsupervised
Uses labeled input and output data	Labels are not available
Well-defined objective (As labels are given, so you know possible type of results to expect)	May discover hidden relationships
	Can be used to learn meaningful representations or features from raw data
human intervention is required to label data	
If you have labeled data and a clear target variable to predict, use supervised learning for accurate predictions	If you have large amounts of unlabeled data and want to find patterns or groupings in the data, opt for unsupervised learning
If you have a mix of labeled and unlabeled data, or the cost of labeling data is high, consider using semi-supervised learning to leverage both types of data	
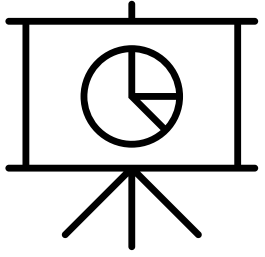
Learning from data and related challenges
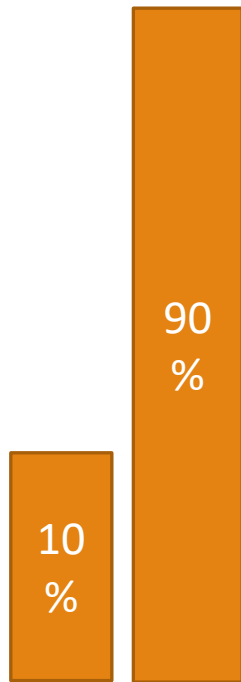
➤ Data Quality and Quantity
  o Noisy, incomplete data can lead to inaccurate and unreliable predictions.
  o Often requires large amount of data

➤ Data Imbalance:
  o E.g, in classification imbalance classes
    o May lead to poor performance

90%

10%

Underfitting

Overfitting

Learning from data and related challenges
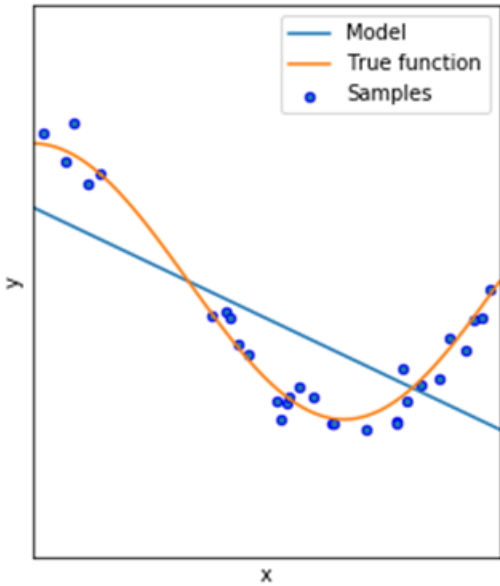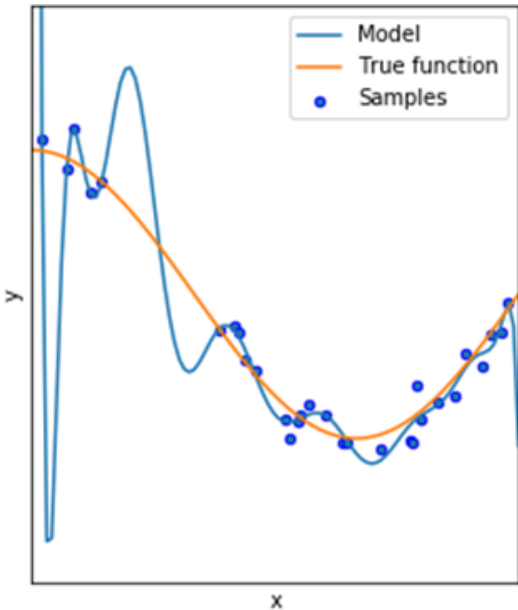
➢Overfitting: Overfitting occurs when a model performs exceptionally well on the training data but fails to generalize to new, unseen data.

➢ Underfitting: When a model is too simplistic to capture the underlying patterns in the data.

➢ Generalization: Ensuring that machine learning models generalize well to new, unseen data.

Data preparation

➢ Data cleaning: Handle missing or inconsistent data
  ○ Approaches:
    ○ Removing them
    ○ Filling with zeros/mean/median,
    ○ Interpolation
➢ Data cleaning: outlier* detection and removing

➢ Data Preprocessing: Feature scaling (E.g. normalization)
➢ Data Preprocessing: Dimensionality reduction
  ➢ Principal Component Analysis (PCA)
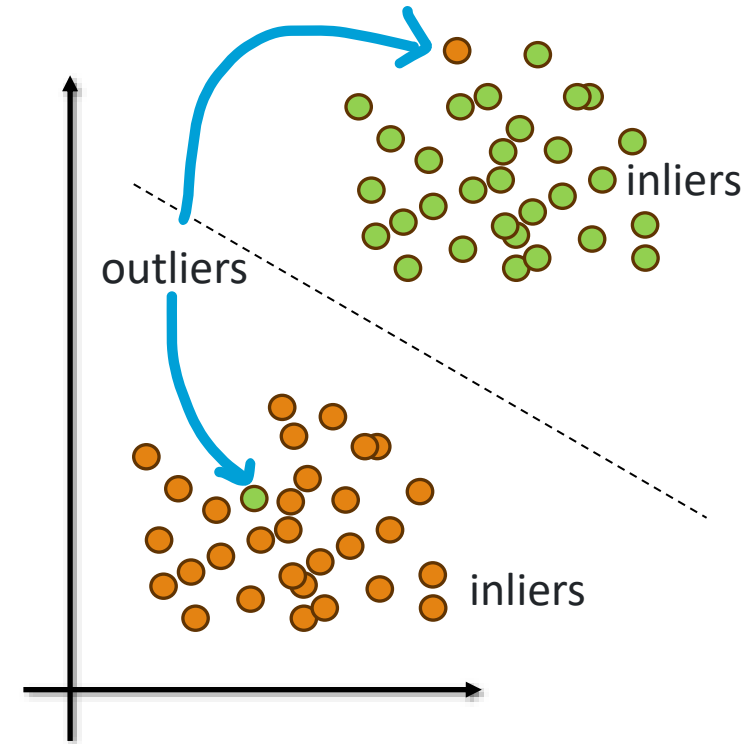
Data preparation

➢ Data Augmentation: Artificially expand the size and diversity of a given dataset.
  o E.g Image rotation, flipping, scaling, cropping ➡ New image

➢ Imbalanced Data:
  o Undersampling of majority class
  o Generating synthetic samples of the minority class

Data preprocessing example

1. Standardization: scale the features of a dataset to have zero mean and unit variance.

2. Scaling features to a range e.g., between 0 and 1
   - Min max scalar ➔ [0, 1]
   - Max Abs Scaler ➔ [-1, 1]

If outliers are there, will it work?

no, if there is an outliers they tend to give wrong interpretation about the dataset
after processing the dataset

Suggestions?

$$Standardization(x) = \frac{x - mean(x)}{std(x)}$$

$$MinMaxScaler(x) = \frac{x - min(x)}{max(x) - min(x)}$$

$$MaxAbsScaler(x) = \frac{x}{max(|x|)}$$

Data preprocessing example

California Housing Dataset

```
import pandas as pd
from sklearn.datasets import fetch_california_housing
```

Use pandas df.describe() to get followings

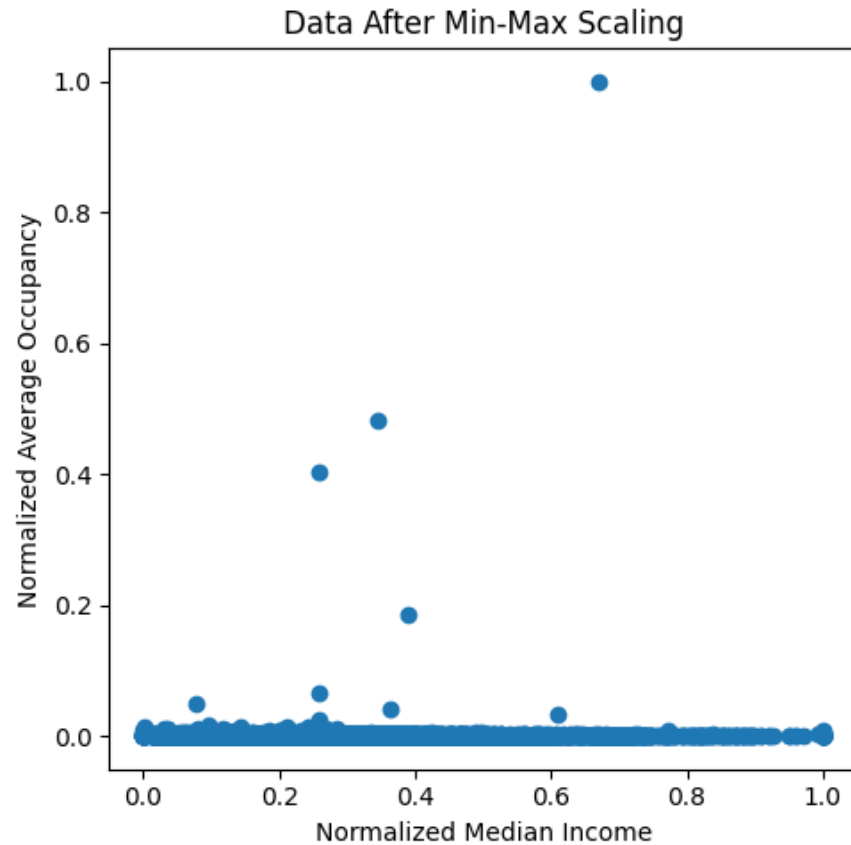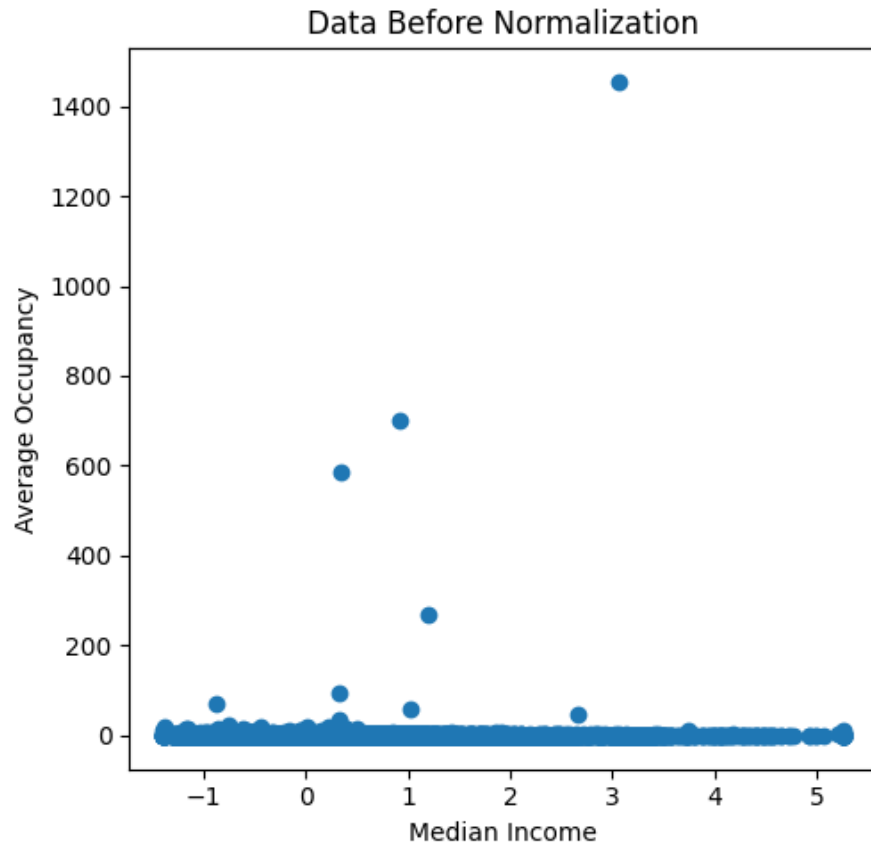|	MedInc	AveOccup
count	20640	20640
mean	3.870671	3.070655
std	1.899822	10.386050
min	0.499900	0.692308
25%	2.563400	2.429741
50%	3.534800	2.818116
75%	4.743250	3.282261
max	15.000100	1243.333333

Independent variables
1. MedInc   Median income in block group (measured in tens of thousands of US Dollars)
2. HouseAge   Median house age in block group (a lower number is a newer building)
3. AveRooms   Average number of rooms per household
4. AveBedrms   Average number of bedrooms per household
5. Population   Block group population
6. AveOccup   Average number of household members
7. Latitude   Block group latitude (a higher value is farther north)
8. Longitude   Block group longitude (a higher value is farther west)
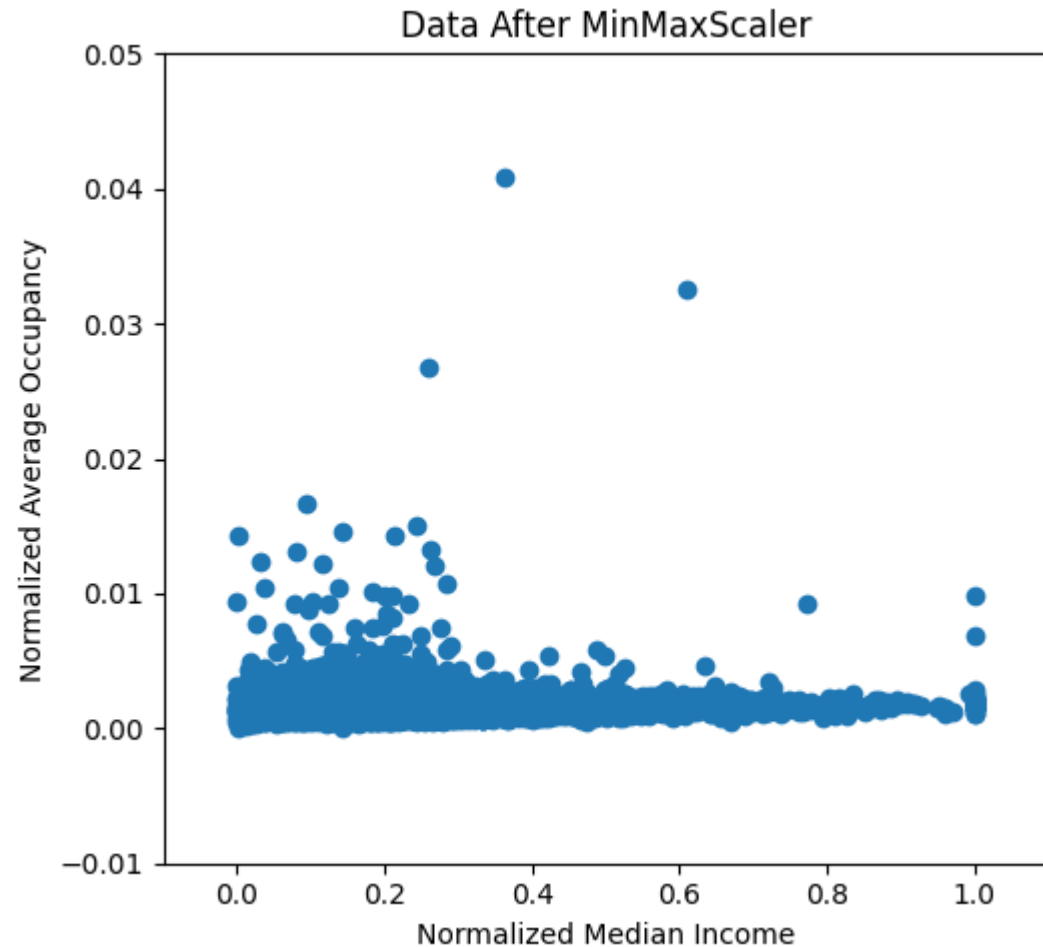
Dependent variable
1. medianHouseValue   Median house value for households within a block (measured in US Dollars)

Data preprocessing example: Min-max Scaler

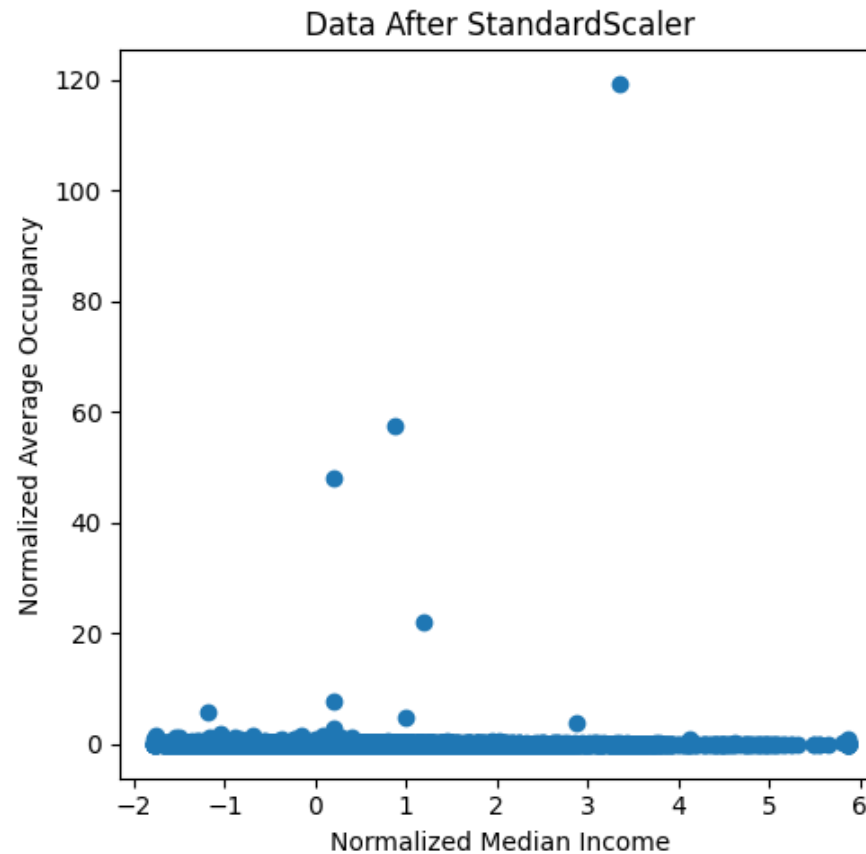# Data preprocessing example: Min-max Scaler

Data After MinMaxScaler

Min-max Scaler

Average Occupancy: Inliers in narrow range [0, 0.005]

Features scales differently

Balanced feature scales cannot be guaranteed with outliers
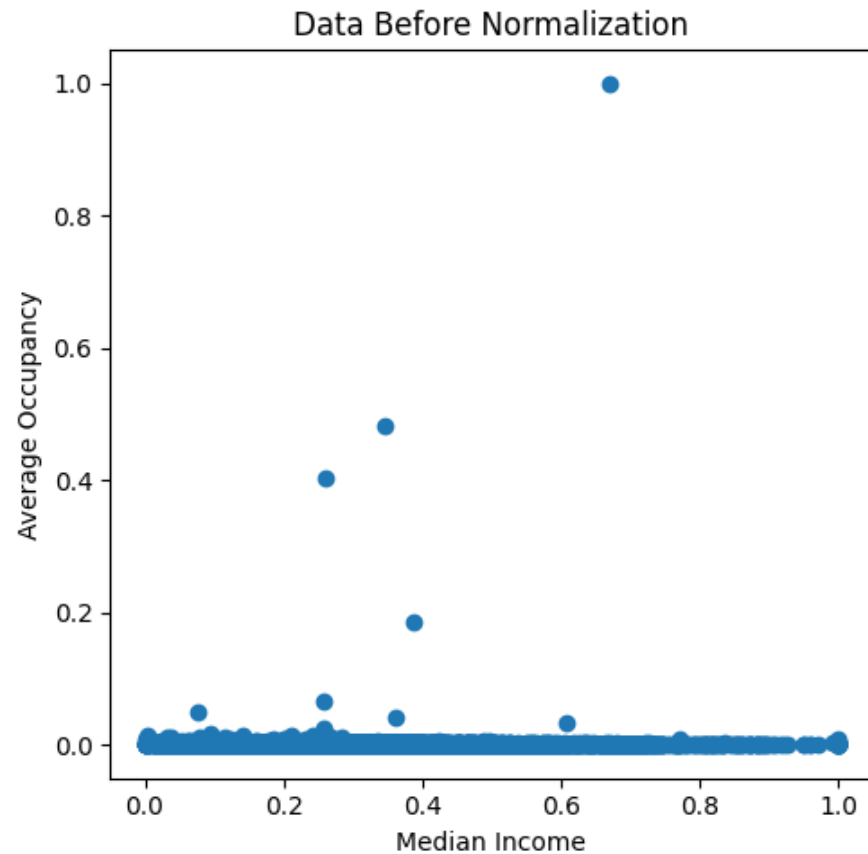
Data preprocessing example: Standard Scaler

Data preprocessing example: Standard Scaler

Data After StandardScaler
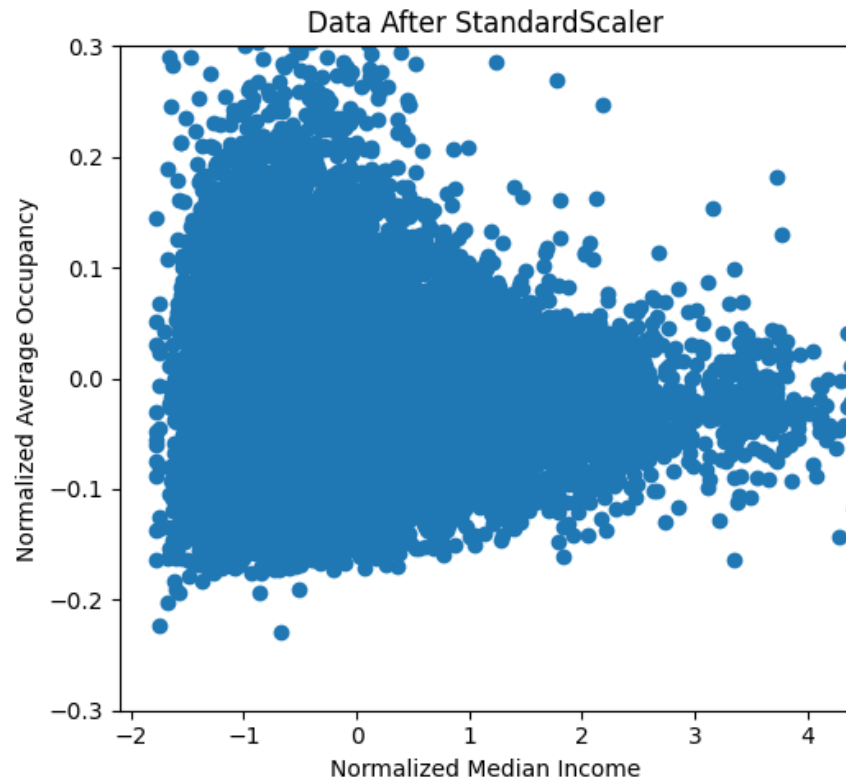
Standard Scaler
Average Occupancy:[-0.2, 0.2]
Median Income [-2, 4]

Features scales differently

Balanced feature scales cannot be guaranteed with outliers
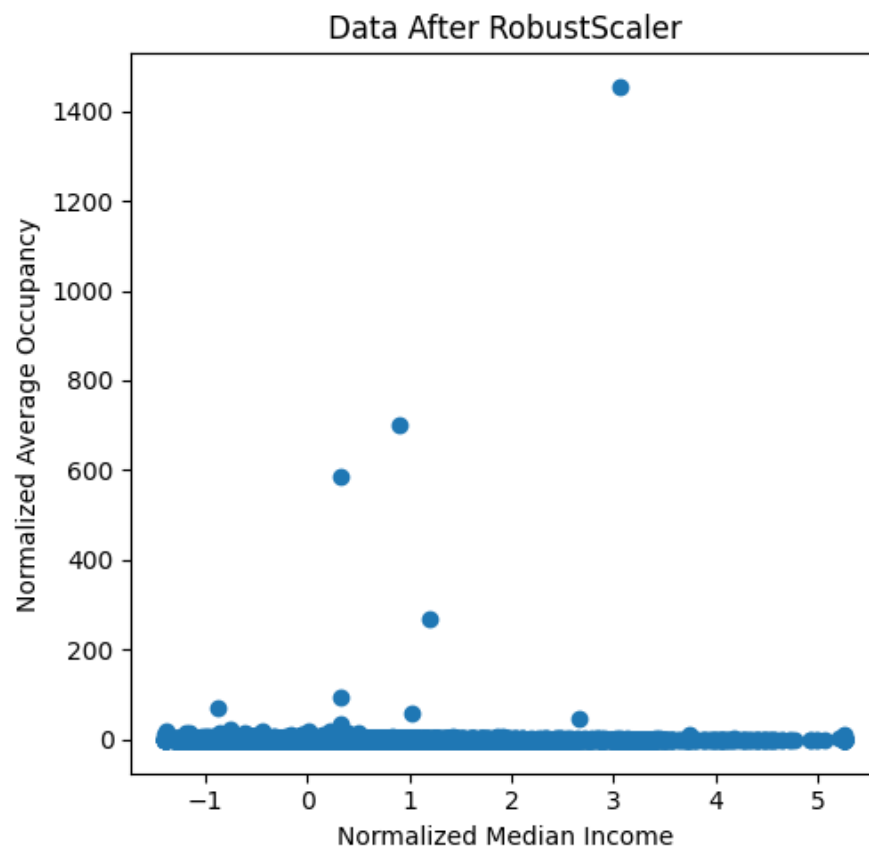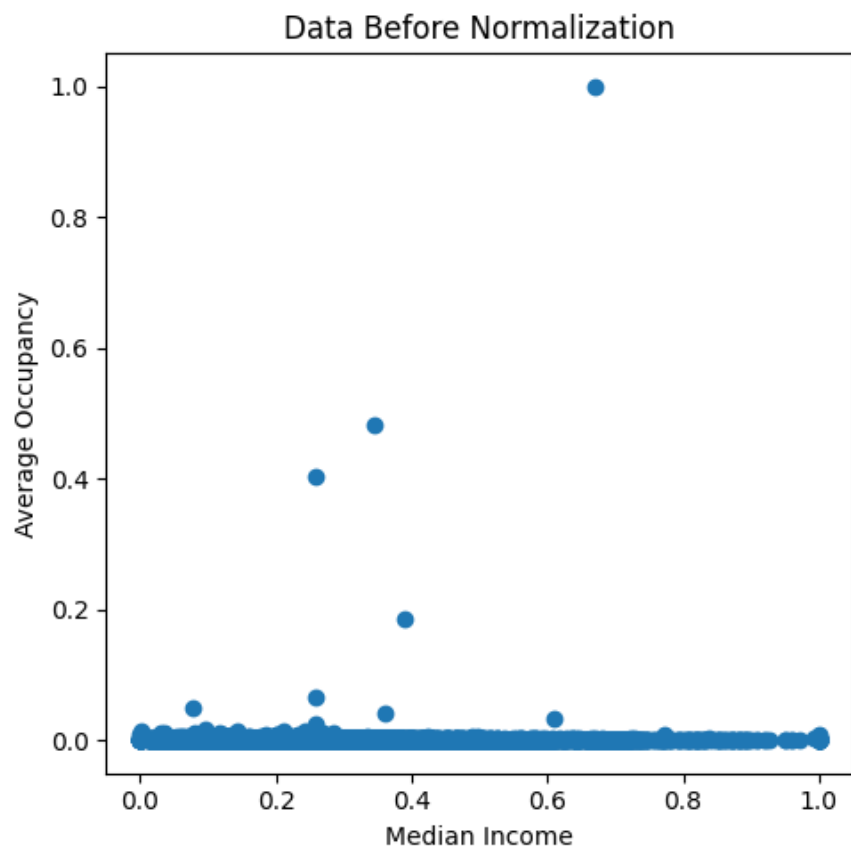
Data preprocessing example: Robust Scalar

Data preprocessing example: Robust Scalar

Data After RobustScaler
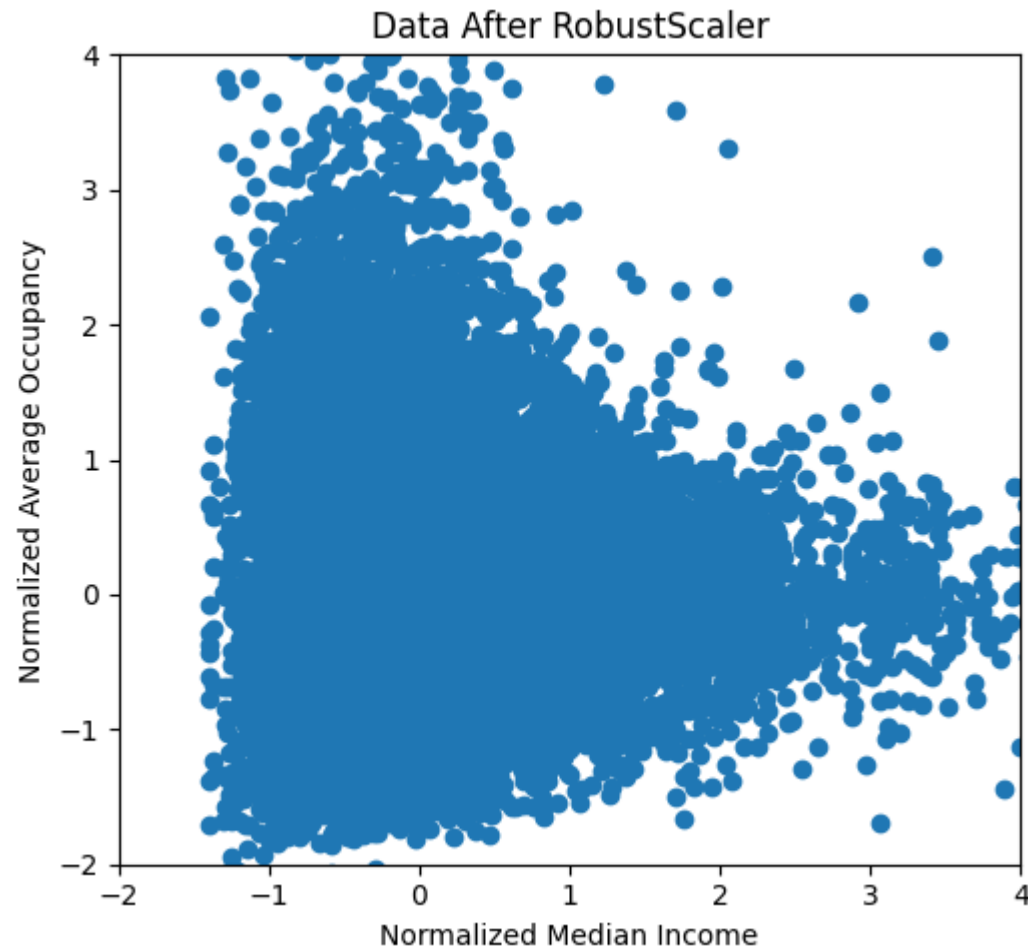
Robust Scaler
Most of data points in both features in range of [-2, 3]

Features scales similarly with outliers as well

Still outliers are there

sklearn.preprocessing.RobustScaler — scikit-learn 1.3.0 documentation

Data preprocessing example: Robust Scalar

Data After Min-Max Scaling

Data After Standard Scaling

Data After Robust Scaling
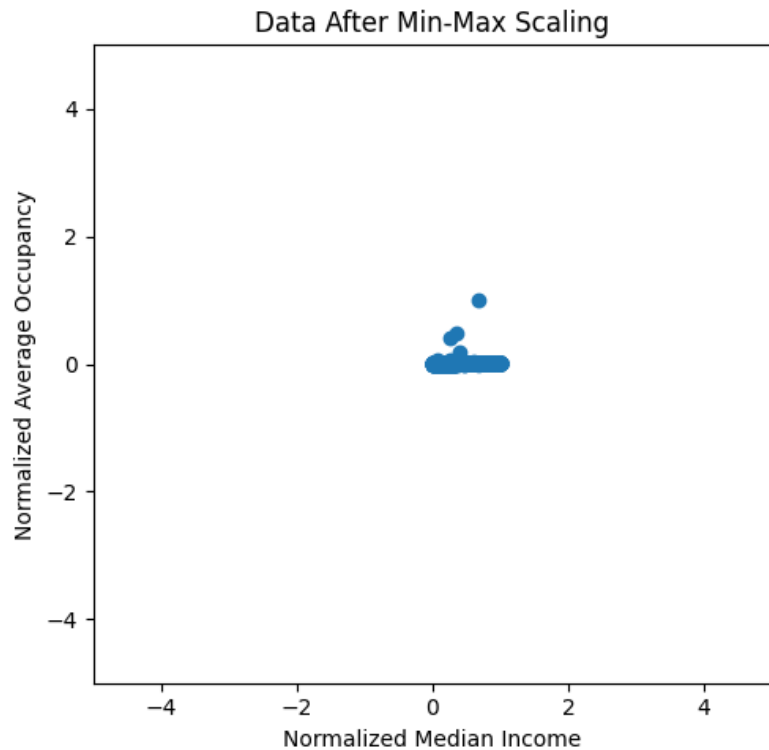
Min-max Scaler
Average Occupancy: Inliers in narrow range [0, 0.005]

Standard Scaler
Average Occupancy:[-0.2, 0.2]
Median Income [-2, 4]
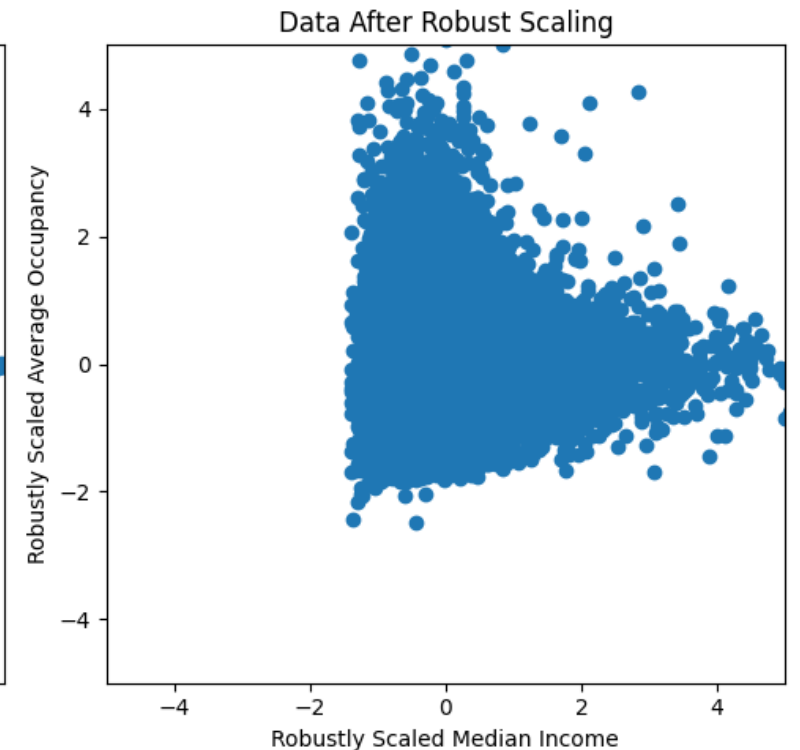
Robust Scaler
Most of data points in both features in range of [-2, 3]

Data preprocessing example: Quantiles information

Data Before Normalization — Data After Quantile Transformer Scaling (Uniform) — Data After Quantile Transformer Scaling (Normal)
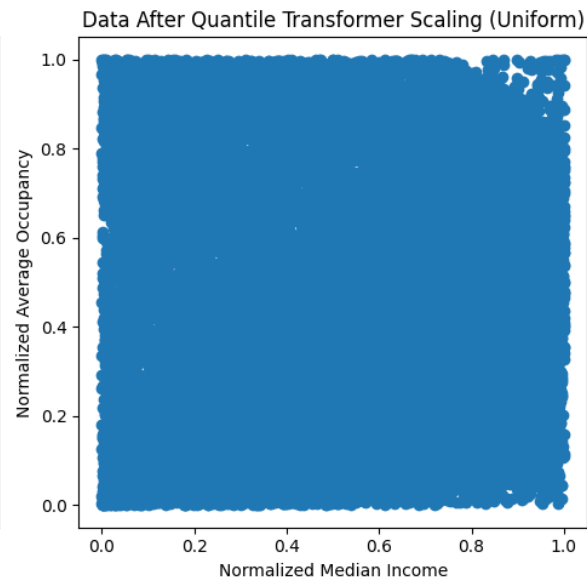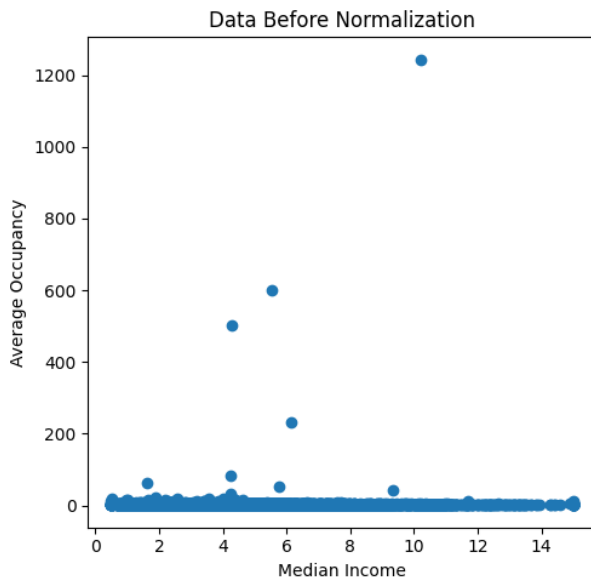
sklearn.preprocessing.Quantile Transformer — scikit-learn 1.3.0 documentation

	MedInc	AveOccup
count	20640.000	20640.000
mean	0.500	0.500
std	0.289	0.289
min	0.000	0.000
25%	0.250	0.250
50%	0.500	0.500
75%	0.750	0.750
max	1.000	1.000

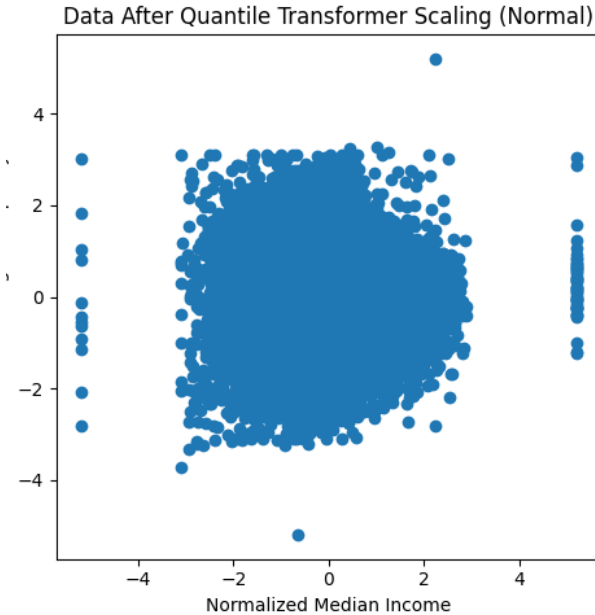|	MedInc	AveOccup
count	20640.000	20640.000
mean	0.004	-0.000
std	1.024	0.999
min	-5.199	-5.199
25%	-0.675	-0.675
50%	-0.000	-0.000
75%	0.674	0.675
max	5.199	5.199

➢ Non-linear transformation

➢ spreads out the most frequent values in each feature, aiming to follow a uniform or normal distribution.

➢ It reduces the impact of outliers, making it a robust preprocessing technique.

➢ The transformation is applied independently to each feature.

➢ It estimates the cumulative distribution function of a feature to map original values to a uniform/normal distribution.

➢ The obtained values are then mapped to the desired output distribution using the associated quantile function.
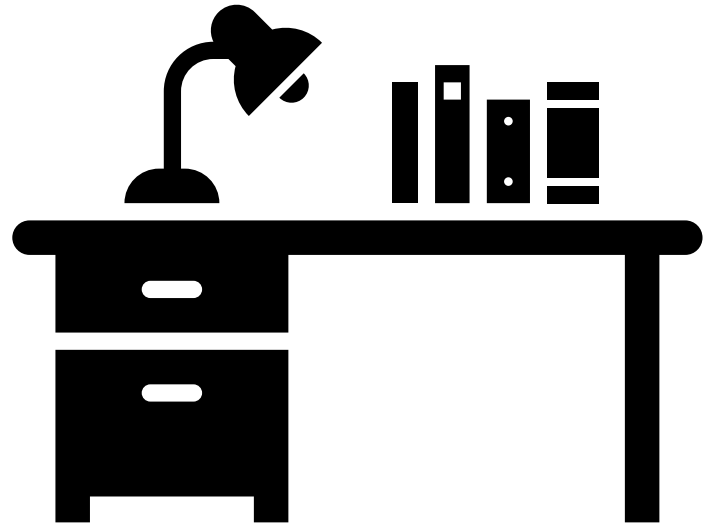
Homework

➤Task: Comparing Data Normalization Methods (See course page in Moodle)

ML Training Process (Supervised Learning)

data preparation

data splitting

Training set

Model Training

Validation set

Model Validation

Testing set

Model Testing

Performance evaluation

- Accuracy
- Precision
- Recall
- F1-Score

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- Root Mean Squared Error (RMSE)
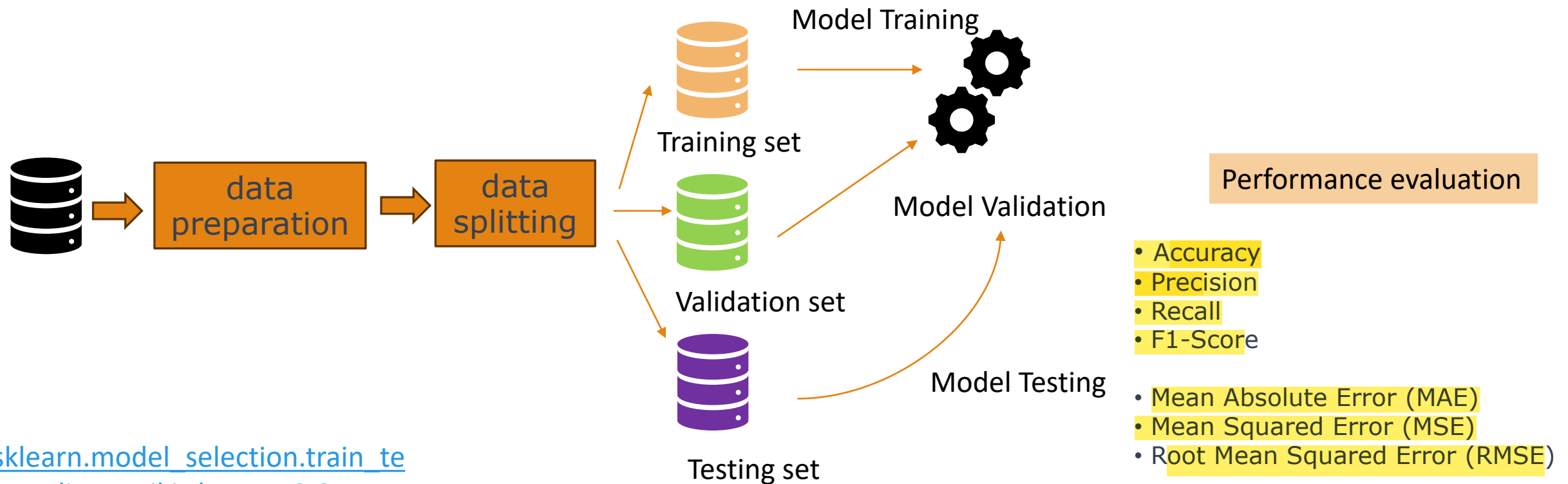
sklearn.model_selection.train_test_split — scikit-learn 1.3.0 documentation

ML Training Process (Supervised Learning)

Model Training

Training set

Model Validation

Validation set

Model Testing

Testing set

data preparation

data splitting

Loss Functions

Used to see how different the guesses made by a machine learning model are from the actual correct answers

➢ Mean Squared Error (MSE)
➢ Mean Absolute Error (MAE)
➢ Binary Cross-Entropy (Log Loss)
➢ Categorical Cross-Entropy
➢ Sparse Categorical Cross-Entropy
➢ Hinge Loss
➢ Kullback-Leibler Divergence (KL Divergence)
➢ Huber Loss
➢ Triplet Loss
➢ Ranking Losses (e.g., Hinge Rank Loss, RankNet Loss))

ML Training Process (Supervised Learning)

- ➤ Mean Squared Error (MSE)
- ➤ Mean Absolute Error (MAE)
- ➤ Binary Cross-Entropy (Log Loss) (BCE)
- ➤ Categorical Cross-Entropy (CCE)
- ➤ Sparse Categorical Cross-Entropy
- ➤ Hinge Loss
- ➤ Kullback-Leibler Divergence (KL Divergence)
- ➤ Huber Loss
- ➤ Triplet Loss
- ➤ Ranking Losses (e.g., Hinge Rank Loss, RankNet Loss))

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

$$BCE = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)]$$

$$CCE = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{C} y_{i,j}\log(\hat{y}_{i,j})$$

Plot of x and -log(x)

— -log(x)

y_i	True target value
$\hat{y}_i$	Predicted value
n	Number of samples in the dataset
C	Number of classes
$y_{i,j}$	One-hot encoded true label for class j
$\hat{y}_{i,j}$	Predicted probability for class j

How to evaluate a model

➢ Accuracy, Precision, Recall, and F-Score

$$\text{Precision} = \frac{TP}{TP + FP}$$

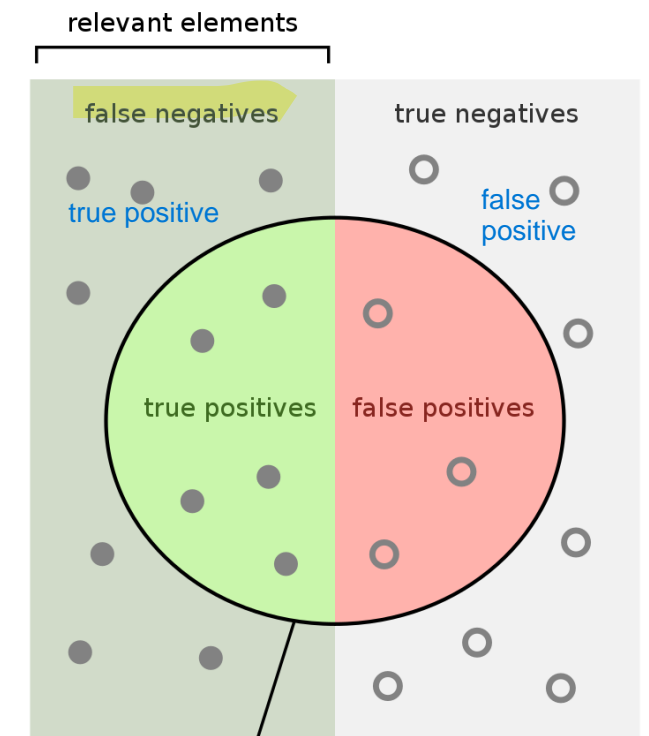$$\text{Recall} = \frac{TP}{TP + FN}$$ all positive

$$F_1 = \frac{2}{\left(\dfrac{1}{\text{Precision}} + \dfrac{1}{\text{Recall}}\right)}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$
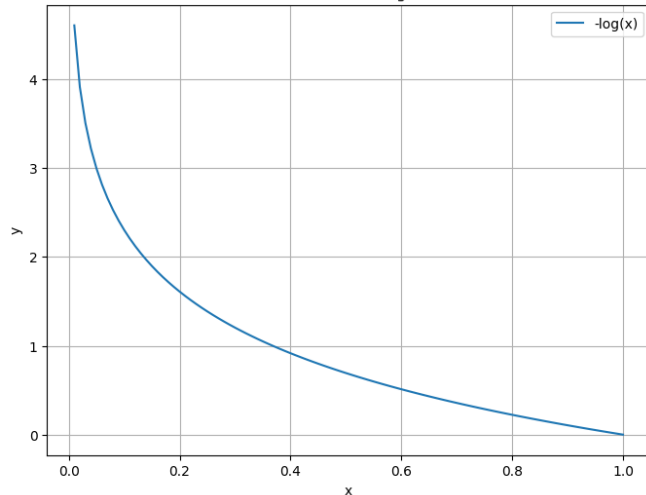
		Predicted class		
		Positive (+)	Negative (-)	Total
True class	Positive (+)	True Pos. (TP)	False Neg. (FN)	P
	Negative (-)	False Pos. (FP)	True Neg. (TN)	N
	Total	P*	N*	

type I error, false alarm   type II error, miss

Accuracy can be a misleading metric for imbalanced data sets.

relevant elements

false negatives   true negatives

true positive

false positive

true positives   false positives

retrieved elements

How many retrieved items are relevant?   How many relevant items are retrieved?

Precision =   Recall =

A higher precision indicates a lower rate of false positives, which means the model is making fewer incorrect positive predictions.   suitable for chose for surgery

A higher recall indicates a lower rate of false negatives, meaning the model is correctly identifying more positive instances   suitable for finding about a virous affected people

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

How to evaluate a model

➢ Accuracy, Precision, Recall, and F-Score

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + FP}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F_1 = \frac{2}{\left(\dfrac{1}{\text{Precision}} + \dfrac{1}{\text{Recall}}\right)}$$

$$\text{Accuracy} = \frac{\text{TP} + TN}{\text{TP} + \text{TN} + FP + FN}$$
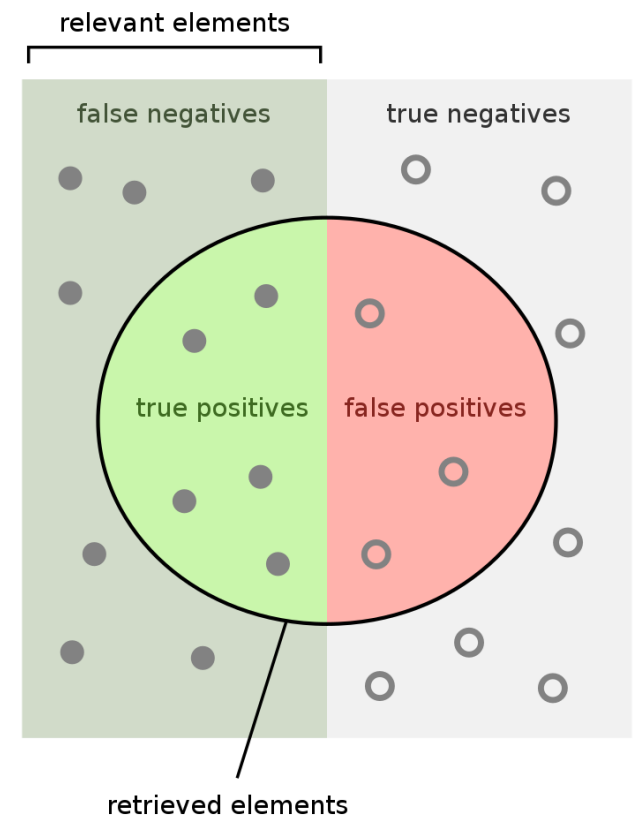
Higher F1-score values indicate a better balance between precision and recall

Higher precision and recall values are generally desired (Application dependent)

➢ Some scenarios, higher precision (lower rate of false positives (false alarm)) may be more critical (e.g., medical diagnosis)

➢ Some scenarios, higher recall (lower rate of false negatives (miss)) may be more important (e.g., fraud detection).

relevant elements

false negatives — true negatives

true positives — false positives

retrieved elements

How many retrieved items are relevant?

How many relevant items are retrieved?

Precision =

Recall =

Model selection

➤ Model selection is the process of choosing the best model from a set of candidate models for a specific task.

Sample of the MNIST dataset of handwritten digits (https://en.wikipedia.org/wiki/MNIST_database)

ML Model

Convolutional Neural Networks (CNNs)

Decision Trees

k-Nearest Neighbors (k-NN)

Model selection

Degree 1
MSE = 4.08e-01(+/- 4.25e-01)

Degree 4
MSE = 4.32e-02(+/- 7.08e-02)

Degree 15
MSE = 1.81e+08(+/- 5.42e+08)
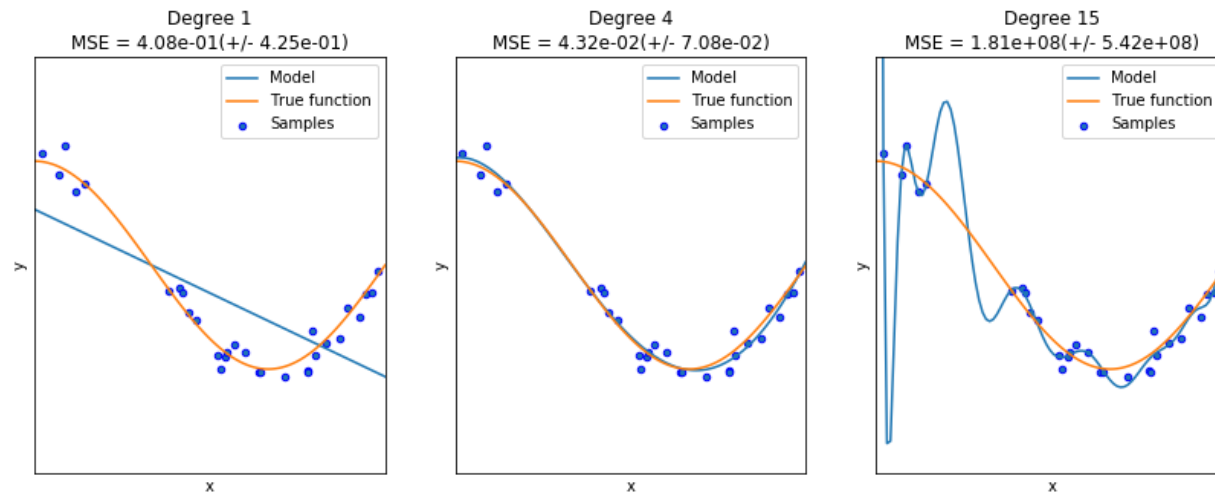
➢ Hyperparameters (parameters that are set before the training process begins)
  ➢ E.g., Learning Rate

➢ Hyperparameters not learned from data

Model selection

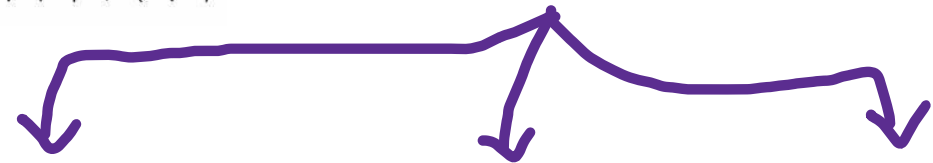➢ Model selection is the process of choosing the best model from a set of candidate models for a specific task.

Training set

Model Training

Model Re-training

Testing set

Model/Models

Model Evaluation

Model Parameter (hyperparameters) Tunning

hyperparameters not learned from data)

Select the best model e.g., using performance metrics like accuracy, F1-score
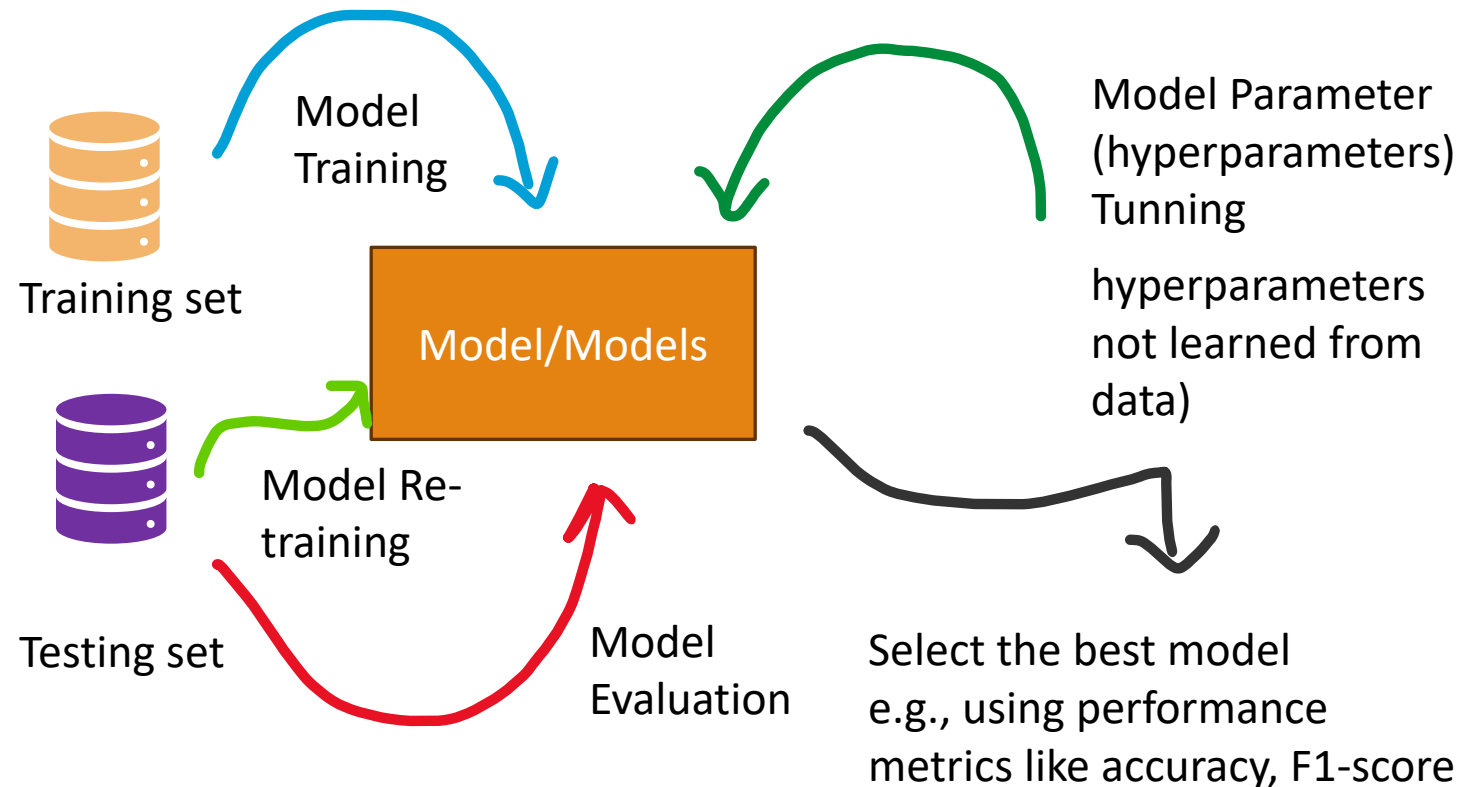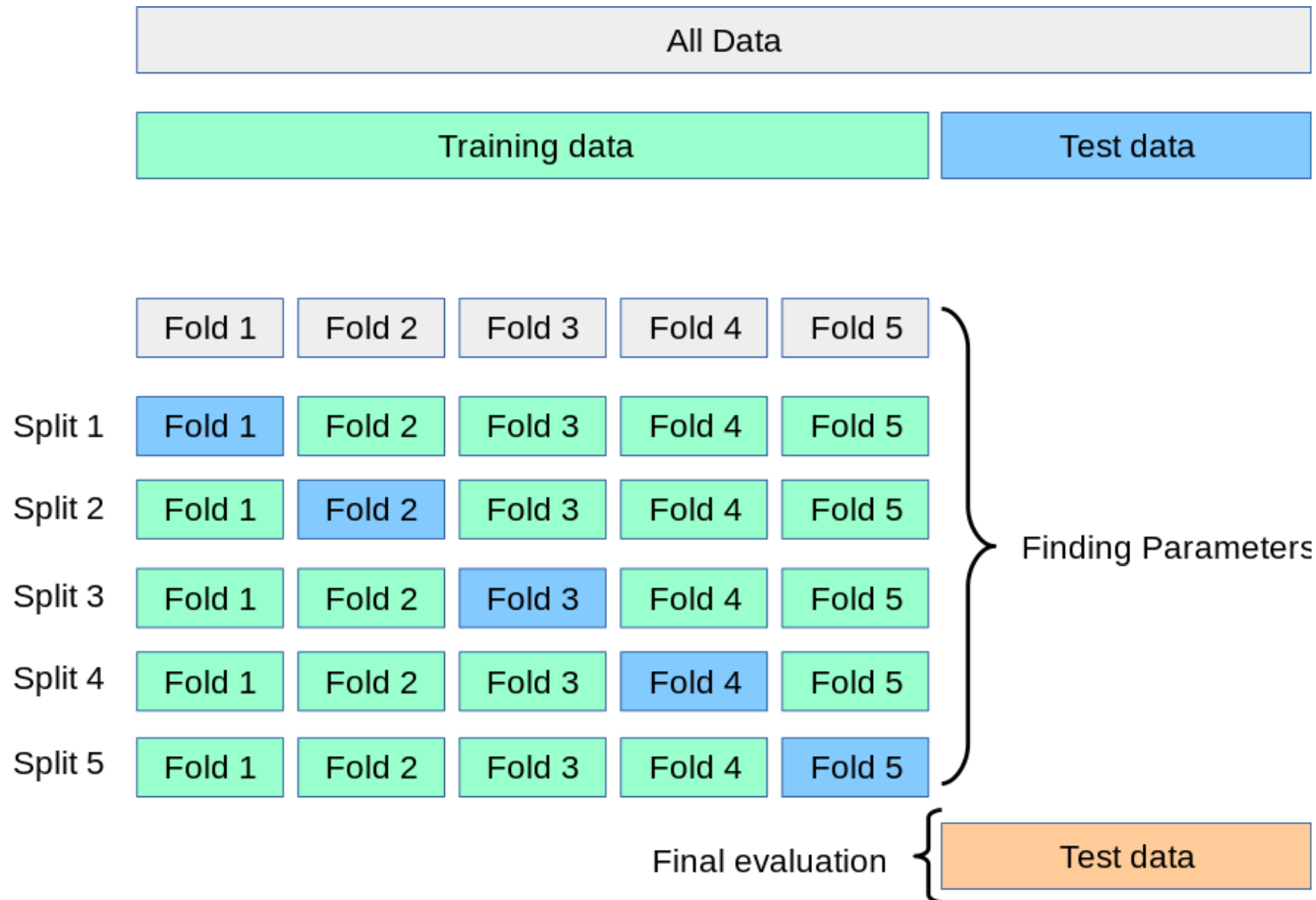
Model selection

Resampling technique:- k-fold cross validation (k-CV)

- The dataset is divided into k subsets (folds) of approximately equal size.

- The model is trained and evaluated k times, each time using a different fold as the test set and the rest as the training set.

- For each iteration, the model is trained on (k-1) folds and evaluated on the remaining fold.

Model selection:

Resampling technique:- k-fold cross validation (k-CV)

➢ Reduced Overfitting: Mitigates overfitting by testing the model on unseen data subsets. Ensures better generalization to new data.

➢ Evaluates model performance for various hyperparameter settings.

➢ Helps to identify the best hyperparameters for optimal model performance

➢ Allows fair and consistent evaluation of multiple models

➢ Maximizes data utilization for both training and testing

➢ Ensures all available data contributes to model evaluation

Model selection:- Hyper parameter tuning-Grid Search

➢ Grid Search involves defining a grid of hyperparameter values to explore.

➢It systematically evaluates all possible combinations from the grid to identify the best-performing one

➢To avoid overfitting during Grid Search, cross-validation is commonly used.

➢Grid Search can be computationally expensive when the hyperparameter space is large.

Model Selection: Probabilistic

- Statistical modeling is used to choose the most appropriate model among a set of candidate models.

- Model comparison

  - Akaike Information Criterion (AIC).
  - Bayesian Information Criterion (BIC).
  - Minimum Description Length (MDL).

"Information theory perspective"

Model Selection: Probabilistic

➢ Akaike Information Criterion (AIC).

$$\ln p(\mathcal{D}|\mathbf{w}_{\mathrm{ML}}) - M$$

Best-fit log likelihood

Number of adjustable parameters of the model

✓ Select the model with the largest value
✓ Both model complexity and model performance is considered

➢Bayesian Information Criterion (BIC)- variation of AIC

✓ Generally, more penalize on model complexity than AIC ➜ more complex models less like to select

try to chose with simple and best performance model

Bias-variance trade-off.

➢ Given a dataset with samples denoted as $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$

➢ A model $f(x)$ that maps input features x to predicted output y

$y = f(x) + e$
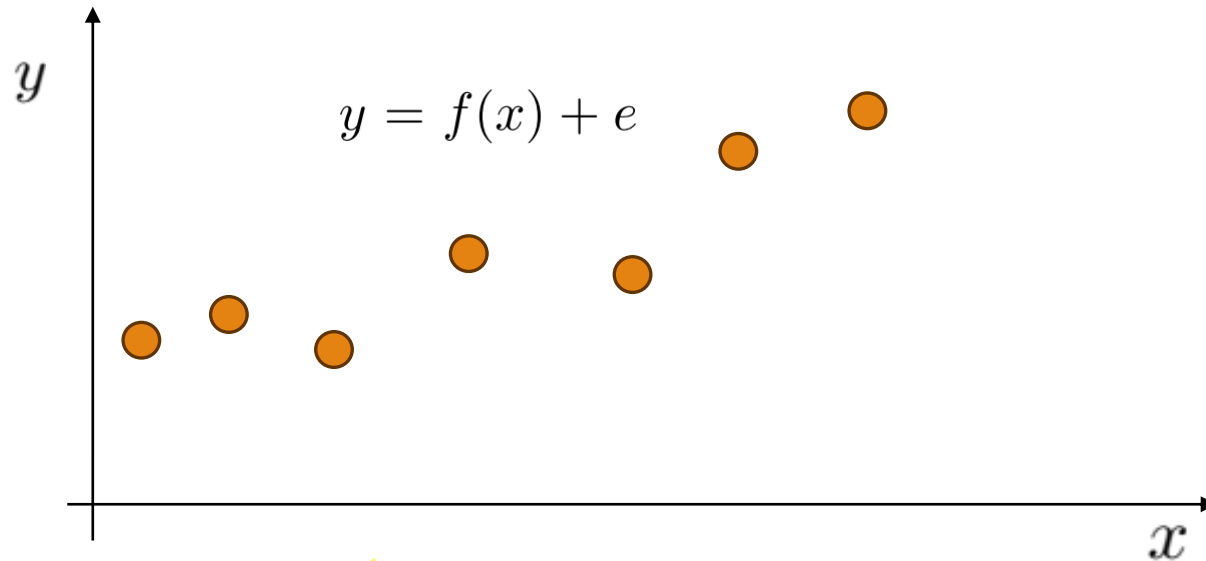
Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_{\text{true},i} - y_{\text{pred},i})^2$$
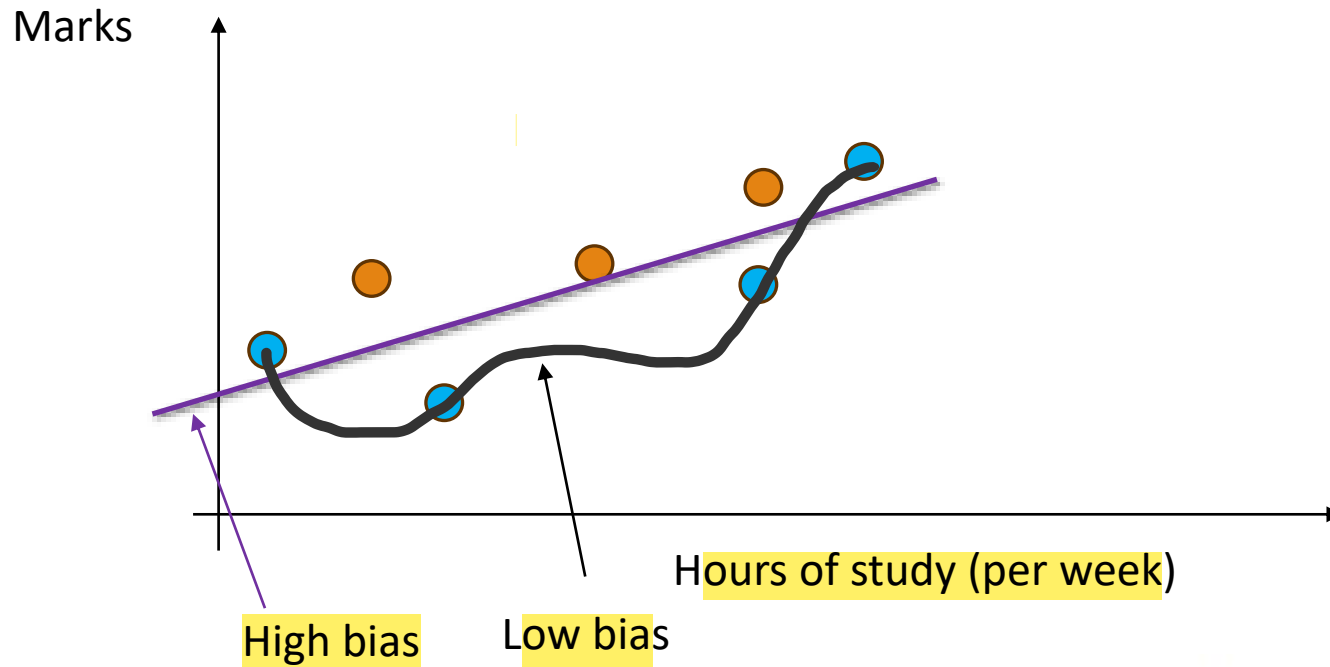
➢ Learned model $\hat{f}(x)$

Bias-variance trade-off

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_{\text{true},i} - y_{\text{pred},i})^2$$

$$= \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + E((y - f(x))^2)$$

$$= \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma_e^2$$

$$= \text{Bias}^2 + \text{Variance} + \boxed{\text{Irreducible Error}}$$

cannot be reduced by any model

Bias-variance trade-off

➢ High bias: Inability to capture the true relationship between input data and output

Marks

high bias is a line and low bias is tries to cover all data values
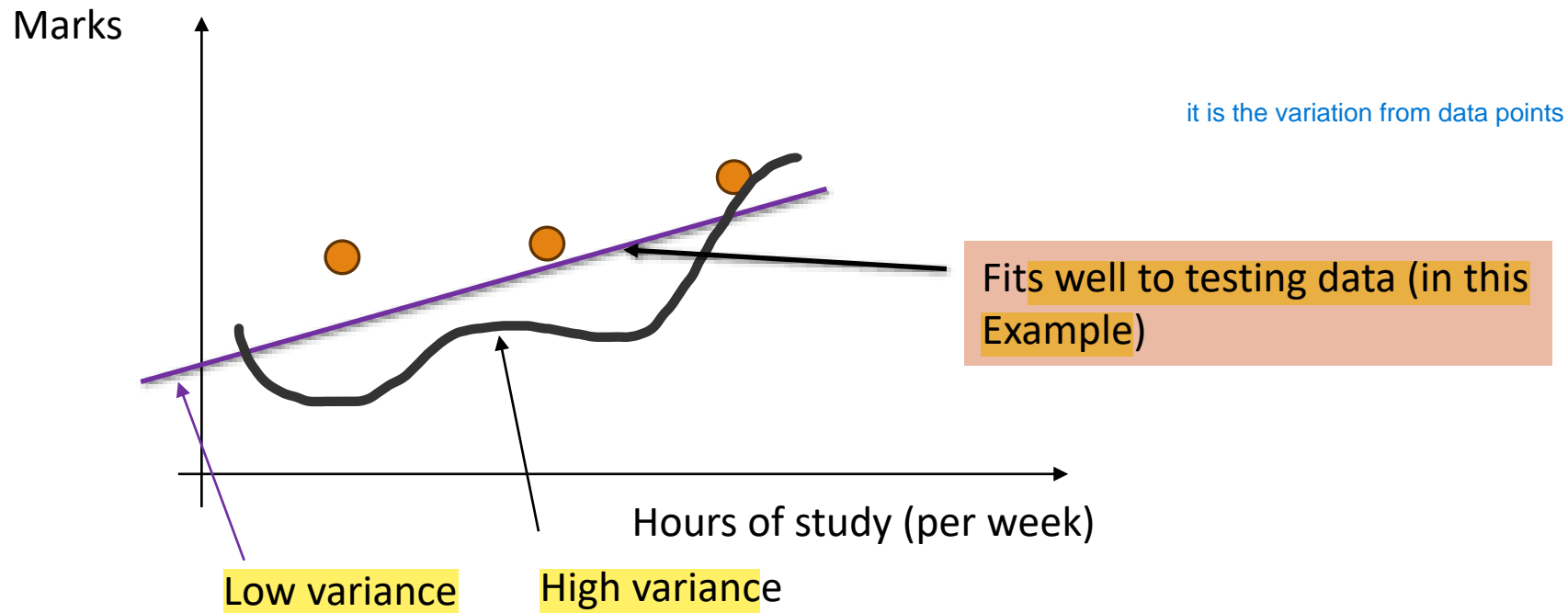
low bias means model catches all data points

Hours of study (per week)

High bias   Low bias

Fits well to training data (in this Example)

● Training samples

$$\left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma_e^2$$

$$\mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

Bias-variance trade-off

➢High variance: Inability to fit different data sets

Marks

it is the variation from data points

Fits well to testing data (in this Example)

Hours of study (per week)

Low variance   High variance
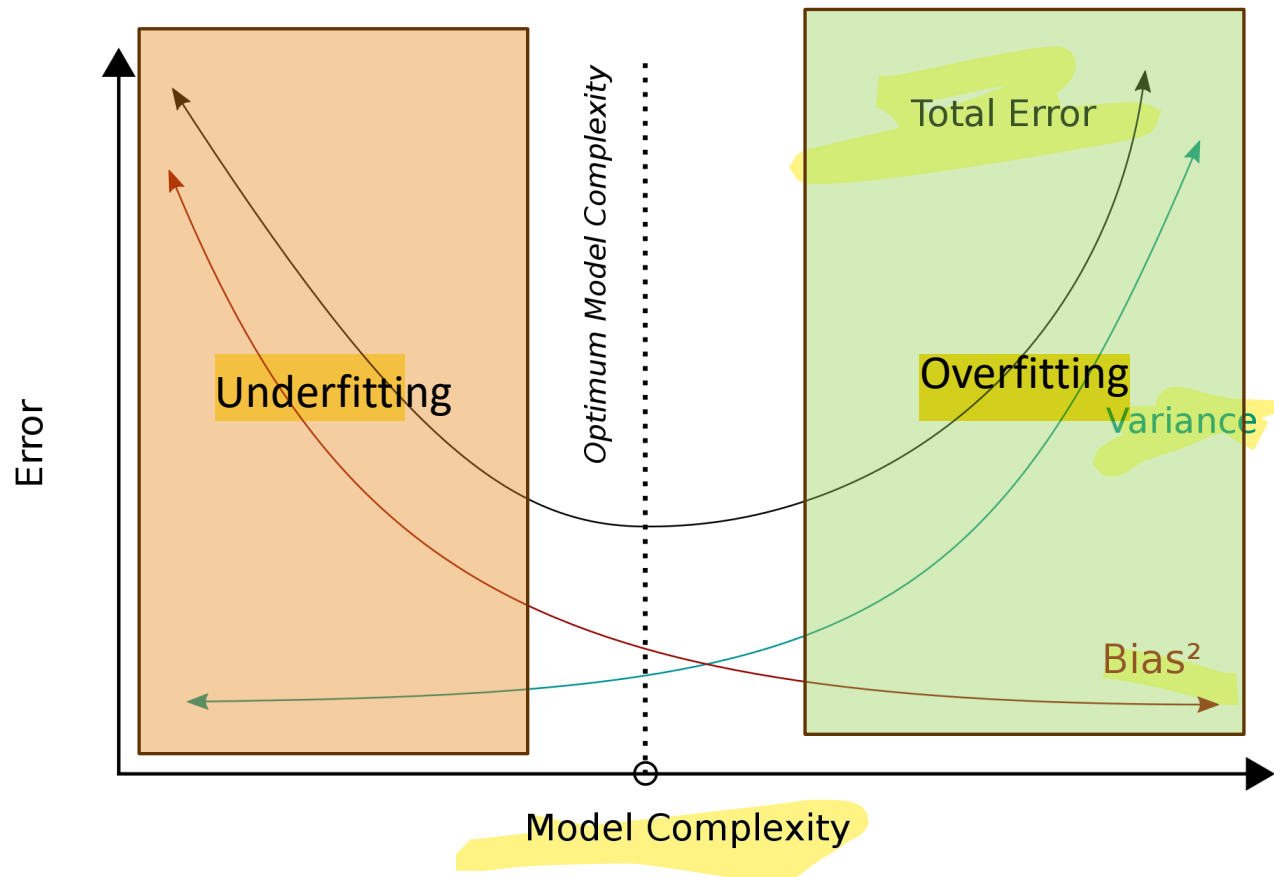
⬤ Testing samples

$$\left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma_e^2$$

$$\text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

Bias-variance trade-off

➢ Too simple model

    will have high bias but low variance

➢ A highly complex model
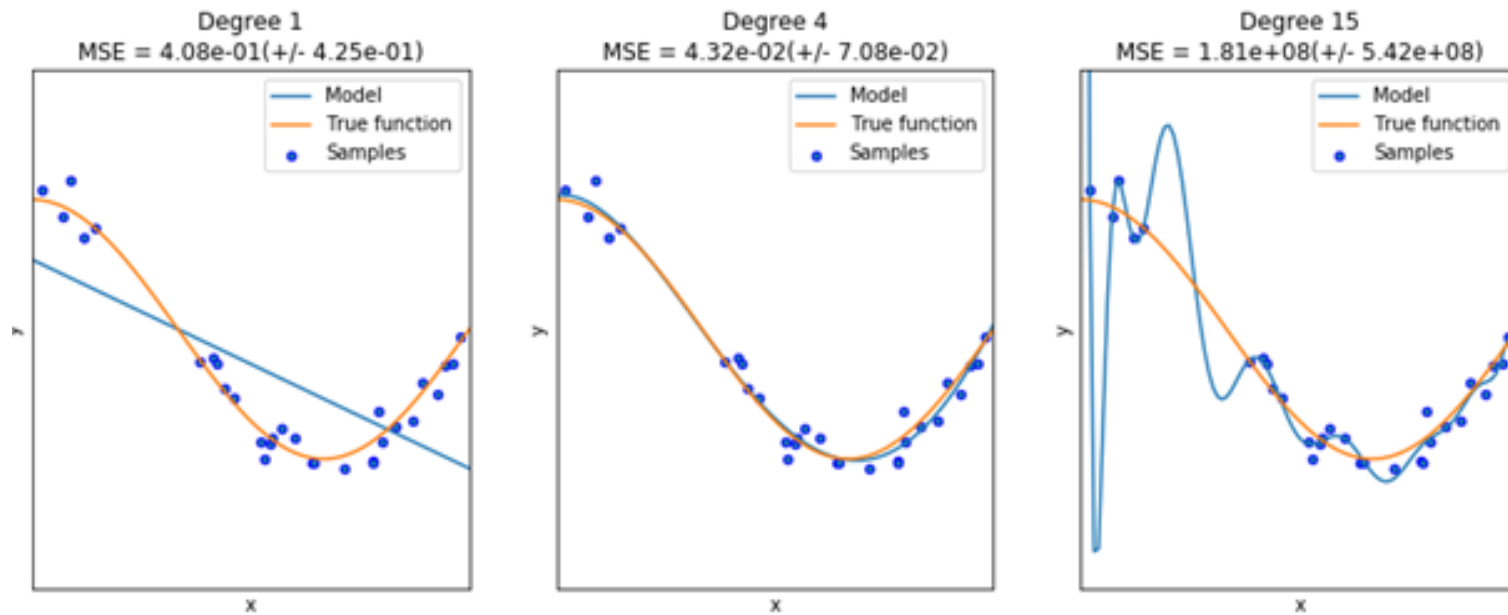
    will have low bias but high variance

variance increase the overfitting

bias reduce the over fitting but increases the under fitting

$$\left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma_e^2$$

$$\text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

Bias-variance trade-off

Degree 1
MSE = 4.08e-01(+/- 4.25e-01)

Degree 4
MSE = 4.32e-02(+/- 7.08e-02)

Degree 15
MSE = 1.81e+08(+/- 5.42e+08)

- Model
- True function
- Samples

➤ Too simple model

   will have high bias but low variance

➤ A highly complex model

   will have low bias but high variance

Underfitting

Overfitting
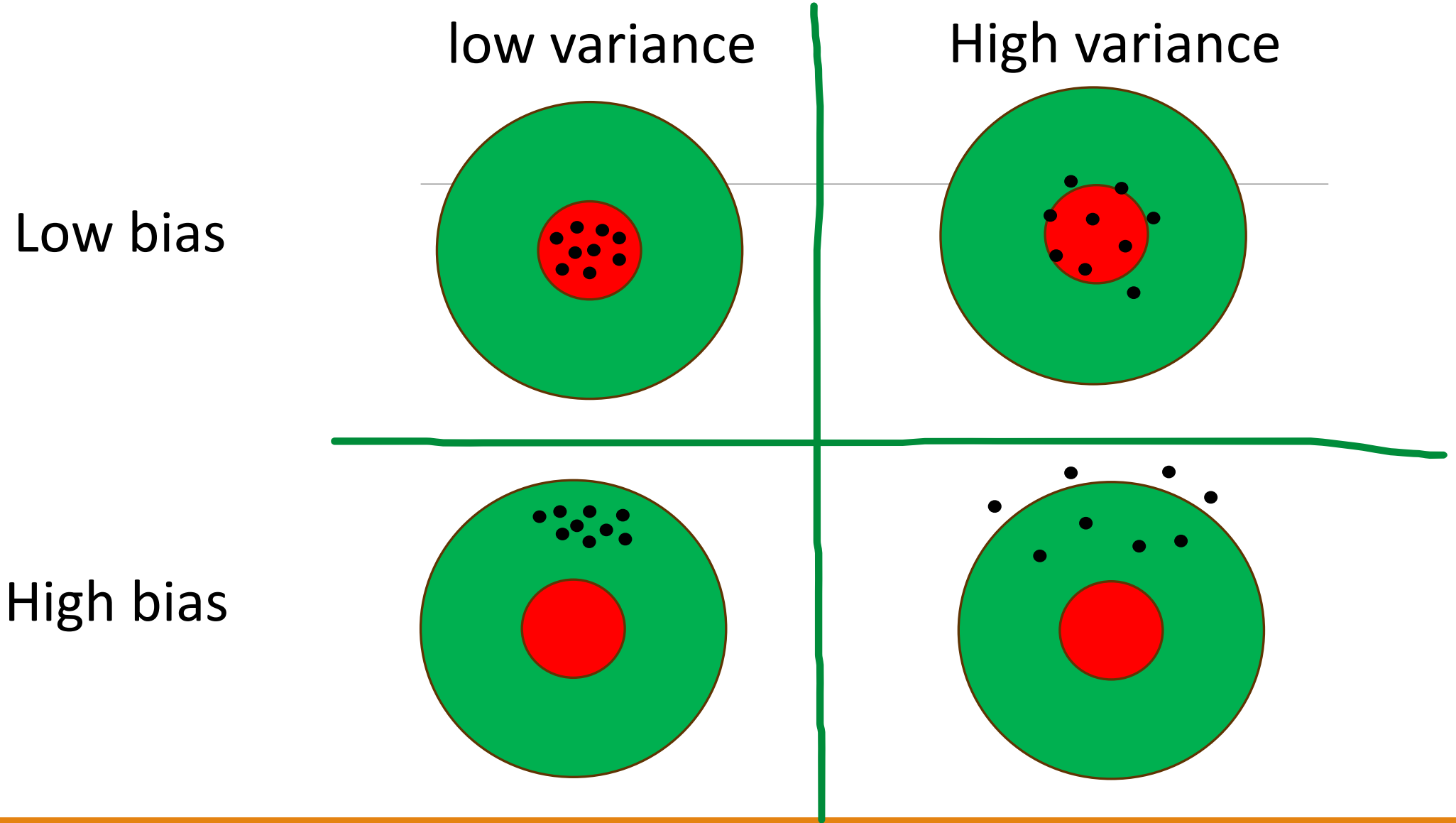
high bias but low variance

low bias but high variance

Bias-variance trade-off
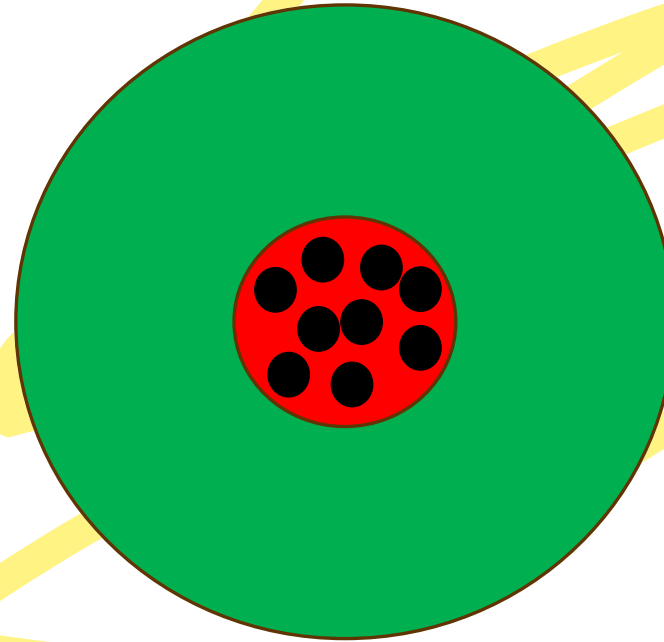
Low bias and low variance

Bias-variance trade-off

	low variance	High variance
Low bias		
High bias		

Thank You
Q & A