Advanced ML Techniques

Unit 4: Session 01

Agenda

- Understand ML Life Cycle
- How to apply ML in real life
- Understand key concepts of ML

Steps of Machine Learning

- Data Collection
- Data Preprocessing
- Feature Engineering
- Feature Selection
- Model Selection
- Model Training
- Model Evaluation
- Model Deployment

Data Collection

- Data Collection Methods
- Data Collection Tools

Data Collection Methods

- Web Scraping
- API
- Survey
- Hugging Face Datasets
- Kaggle
- UCI Machine Learning Repository
- Government Data
- Bangladesh bank

Data Collection Tools

- Scrapy
- Beautiful Soup
- Pandas
- Requests

Checklist for Data Collection

- ✓ What is the problem you are trying to solve?
- ✓ What data is required?
- ✓ What data is available?
- ✓ What data is missing?
- ✓ How much data is required?
- ✓ What is the cost of data collection?
- ✓ What are the data sources?
- ✓ What are the data formats?
- ✓ How frequently the data is updated / collected?

Data Preprocessing

- Understand the data
- Clean the data

How data is structured?

- Structured Data e.g. CSV, Excel, Database
- Unstructured Data e.g. Text, Image, Audio, Video
- Semi-structured Data e.g. JSON, XML, HTML, YAML

What are the data types?

- Quantitative / Numerical
  - Discrete e.g. Count, Number of children, Number of cars
  - Continuous e.g. Height, Weight, Age
- Qualitative / Categorical
  - Ordinal e.g. Rating, Ranking
  - Nominal e.g. Gender, Color, Country, City

Inspect the data

Inspect	Method
First few rows	`df.head()`
Last few rows	`df.tail()`
Data types	`df.dtypes`
Data shape	`df.shape`
Data summary	`df.describe()`

Inspect the data

Inspect	Method
Missing values	`df.isnull().sum()`
Unique values	`df.nunique()`
Size of data	`df.size`
Data columns	`df.columns`
Data index	`df.index`
Data info	`df.info()`

Data Cleaning

Handle Missing value

- Drop the rows with missing values
- Impute the missing values
  - Mean, Median, Mode
  - Forward fill, Backward fill
  - Interpolation
  - KNN Imputer

Data Cleaning

Handle Outliers

- Detect outliers
  - Boxplot
  - Scatter plot
  - Z-score
  - IQR
- Remove / Process outliers based on the ML model

Note: Keep note of the outliers and how you handled them.

Feature Engineering

Create new features from existing features

Feature Engineering Methods

- Aggregation
- Transformation
- Derivation
- Binning
- Encoding Categories

Aggregation

Combine multiple features to create a new feature. e.g. Total Sales = Quantity * Price

Transformation

Apply mathematical transformation to the feature. e.g. Log Transformation

Derivation

Create new feature from existing feature. e.g. Age from Date of Birth

Binning

Create bins from continuous feature. e.g. Age Group

Age	Age Group
0-10	Child
11-20	Teen
21-30	Young Adult
31-40	Adult

One Hot Encoding

Color	Color_Red	Color_Green	Color_Blue
Red	1	0	0
Green	0	1	0
Blue	0	0	1

Label Encoding

Color	Color_Label
Red	0
Green	1
Blue	2

Time Series Data

- Detrending (Remove Trend)
  - Moving Average
  - Differencing
  - Regression
- Seasonal Adjustment (Remove Seasonality)
  - Additive
  - Multiplicative

Numerical Data Transformations

- Standardization
- Normalization
  - Min-Max Scaling
  - Robust Scaling
- Log Transformation
- Box-Cox Transformation
- Discretization
  - Equal Width Binning
  - Equal Frequency Binning

Feature Selection

Process of selecting the most important features from the dataset

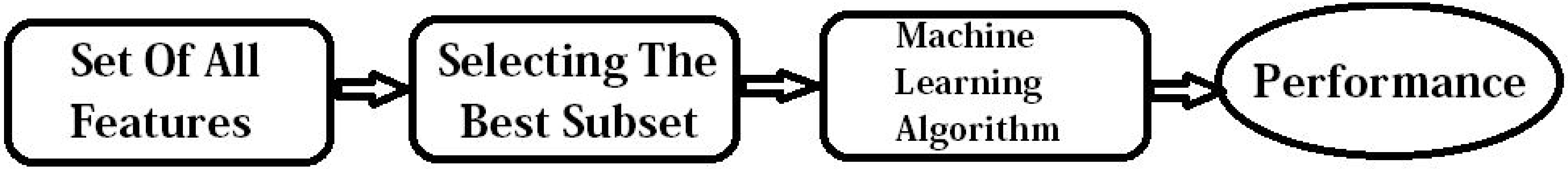
Feature Selection Methods

- Filter Methods
- Wrapper Methods
- Embedded Methods

Filter Methods

Filter methods pick up the intrinsic properties of the features measured via **univariate statistics** instead of cross-validation performance. These methods are faster and computationally less expensive than wrapper methods.

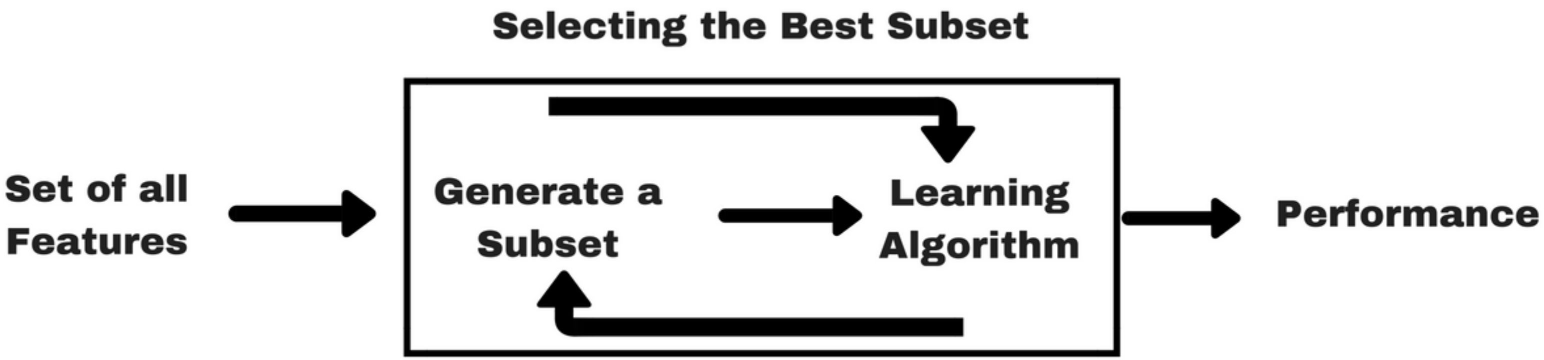
- Correlation
- Chi-Square



Wrapper Methods

Wrappers require some method to search the space of all possible subsets of features, assessing their quality by learning and evaluating a classifier with that feature subset.

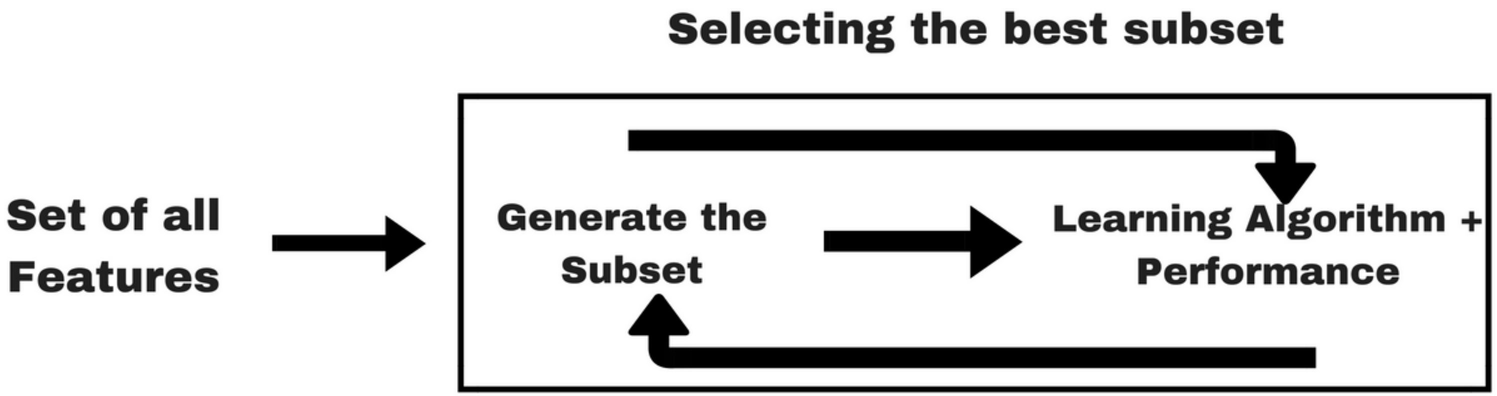
- Forward Selection
- Backward Elimination
- Exhaustive Feature Selection



Embedded Methods

These methods encompass the benefits of both the wrapper and filter methods by including interactions of features but also maintaining reasonable computational costs.

- Lasso Regression
- Ridge Regression



Model Selection

What is your goal?

- Predict value of a continuous variable
  - Regression
- Predict class of a categorical variable
  - Classification
    - Binary Classification
    - Multi-class Classification

What is your goal?

- Discover patterns in the data
  - Clustering
- Anomaly Detection
- Recommendation
  - Collaborative Filtering
  - Content Based Filtering
- Text Analysis
  - Sentiment Analysis
  - Topic Modeling
  - Text Summarization

Types of ML Algorithms

- Supervised Learning
- Unsupervised Learning

Supervised Learning

Linear Models	Tree Based Models
Linear Regression	Decision Tree
Logistic Regression	Random Forest
Ridge Regression	Gradient Boosting
Lasso Regression	XGBoost
	LightGBM

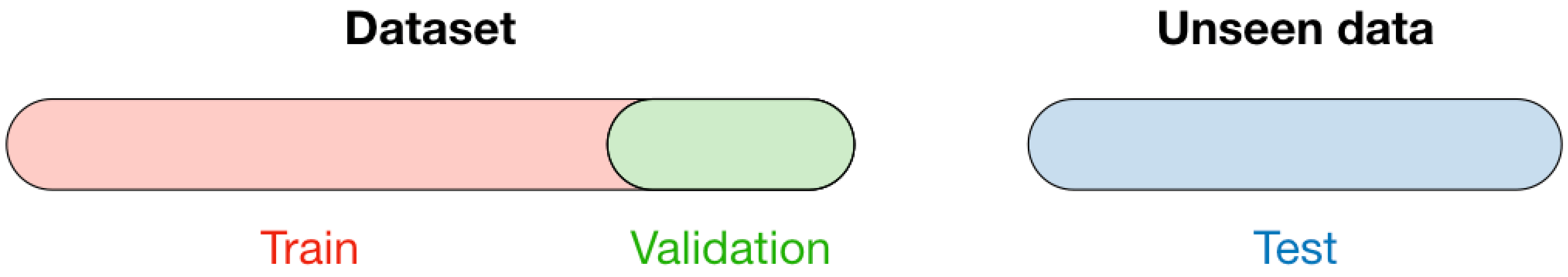
Unsupervised Learning

- Clustering
  - K-Means
  - Hierarchical Clustering
  - Gaussian Mixture Model
- Association
  - Apriori
  - FP-Growth

Model Training & Evaluation

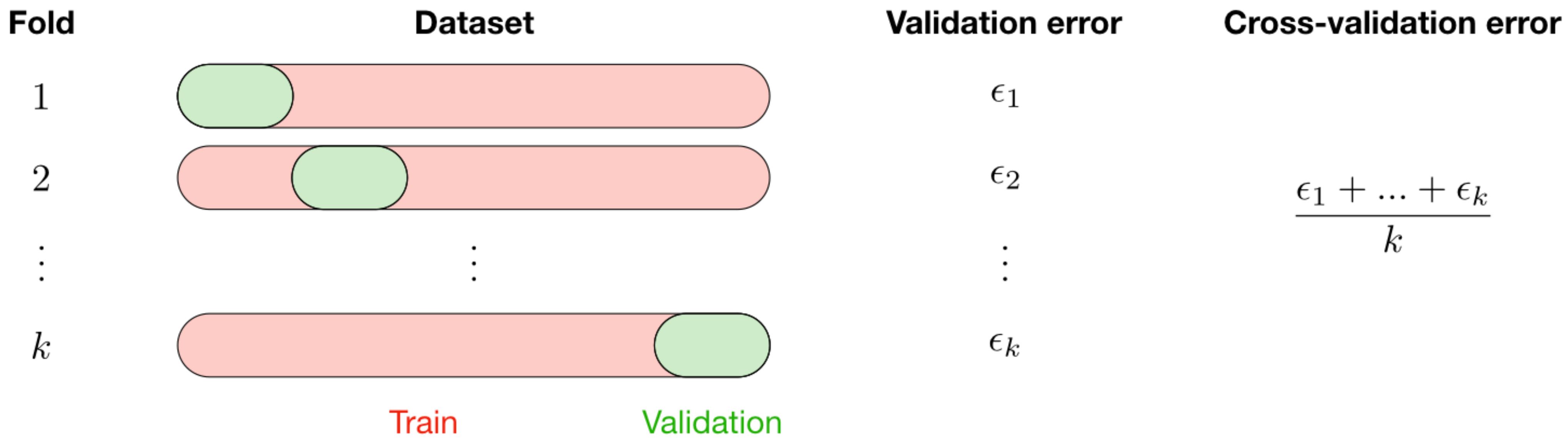
Split the data

- Training Set
- Validation Set
- Test Set



Split the data

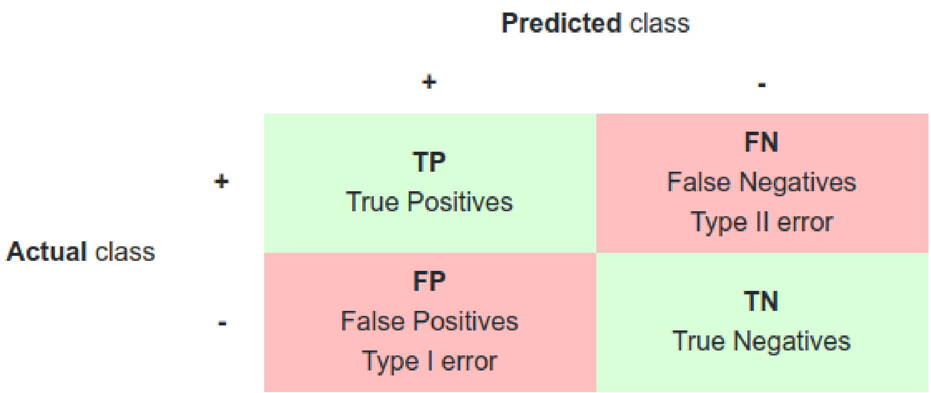
K-Fold Cross Validation



Define a loss function

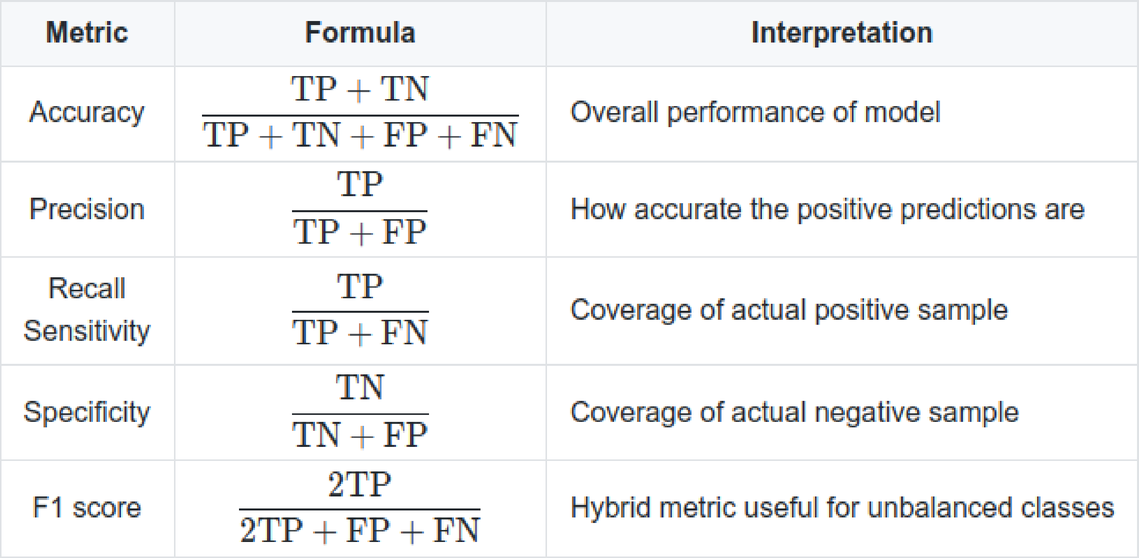
- Regression
    - Mean Squared Error (MSE)
    - Root Mean Squared Error (RMSE)
    - Mean Absolute Error (MAE)
- Classification
    - Accuracy
    - Precision
    - Recall
    - F1 Score
    - ROC AUC

Confusion Matrix



		Predicted class	
		+	-
Actual class	+	**TP** True Positives	**FN** False Negatives Type II error
	-	**FP** False Positives Type I error	**TN** True Negatives

Main Metrics



Metric	Formula	Interpretation
Accuracy	$\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$	Overall performance of model
Precision	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$	How accurate the positive predictions are
Recall Sensitivity	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	Coverage of actual positive sample
Specificity	$\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}$	Coverage of actual negative sample
F1 score	$\frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}$	Hybrid metric useful for unbalanced classes

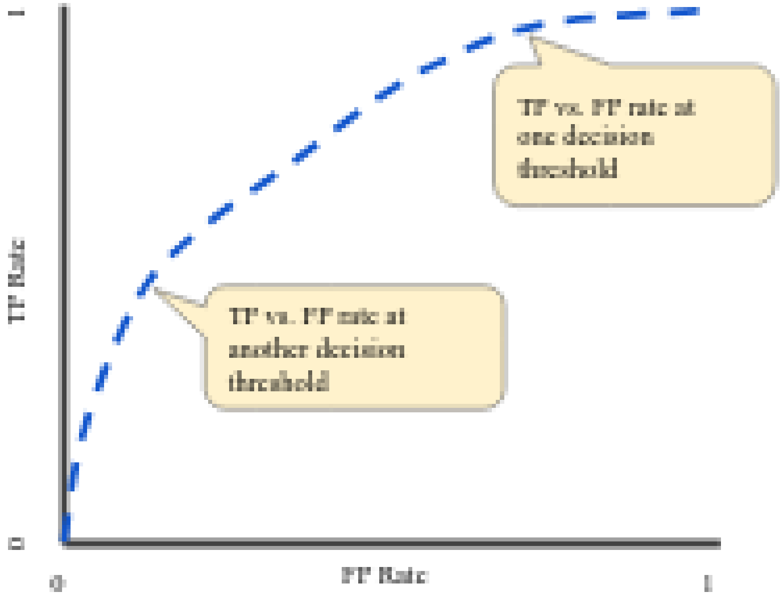
ROC

A Receiver Operating Characteristic (ROC) Curve is a plot of the true positive rate against the false positive rate.

The relationship between sensitivity and specificity. For example, a decrease in sensitivity results in an increase in specificity.

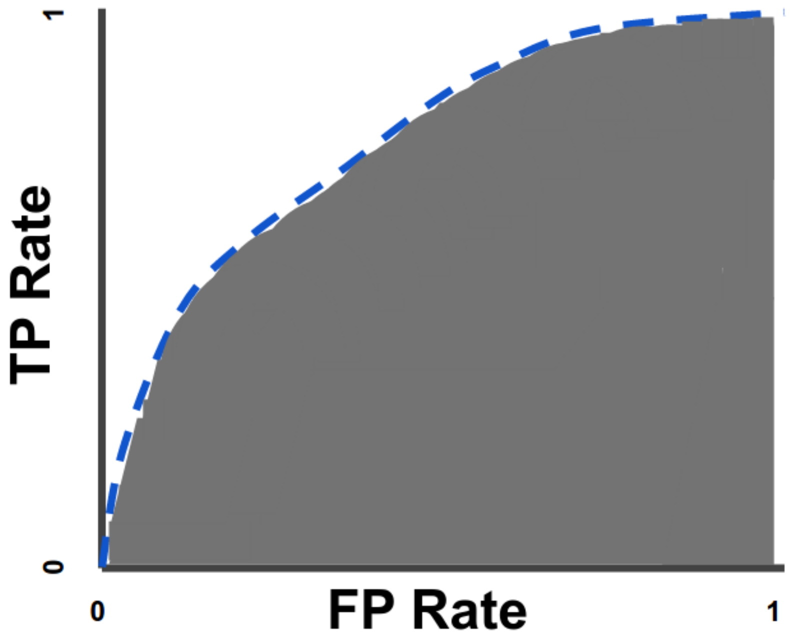
Test accuracy; the closer the graph is to the top and left-hand borders, the more accurate the test. Likewise, the closer the graph to the diagonal, the less accurate the test. A perfect test would go straight from zero up to the top-left corner and then straight across the horizontal.

- True Positive Rate (TPR) $TPR = \frac{TP}{TP+FN}$
- False Positive Rate (FPR). $FPR = \frac{FP}{FP+TN}$



AUC

AUC stands for "Area under the ROC Curve." That is, AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1).



Hyperparameter Tuning

- Learning Rate
- Number of Trees
- Number of Layers
- Number of Neurons
- Number of Cluster
- Number of Neighbors

Bias-Variance Tradeoff

Bias

Differences between expected values and the predicted values are known as error or bias error or error due to bias.

Low Bias: Low bias value means fewer assumptions are taken to build the target function. In this case, the model will **closely match the training dataset**. (Overfitting)

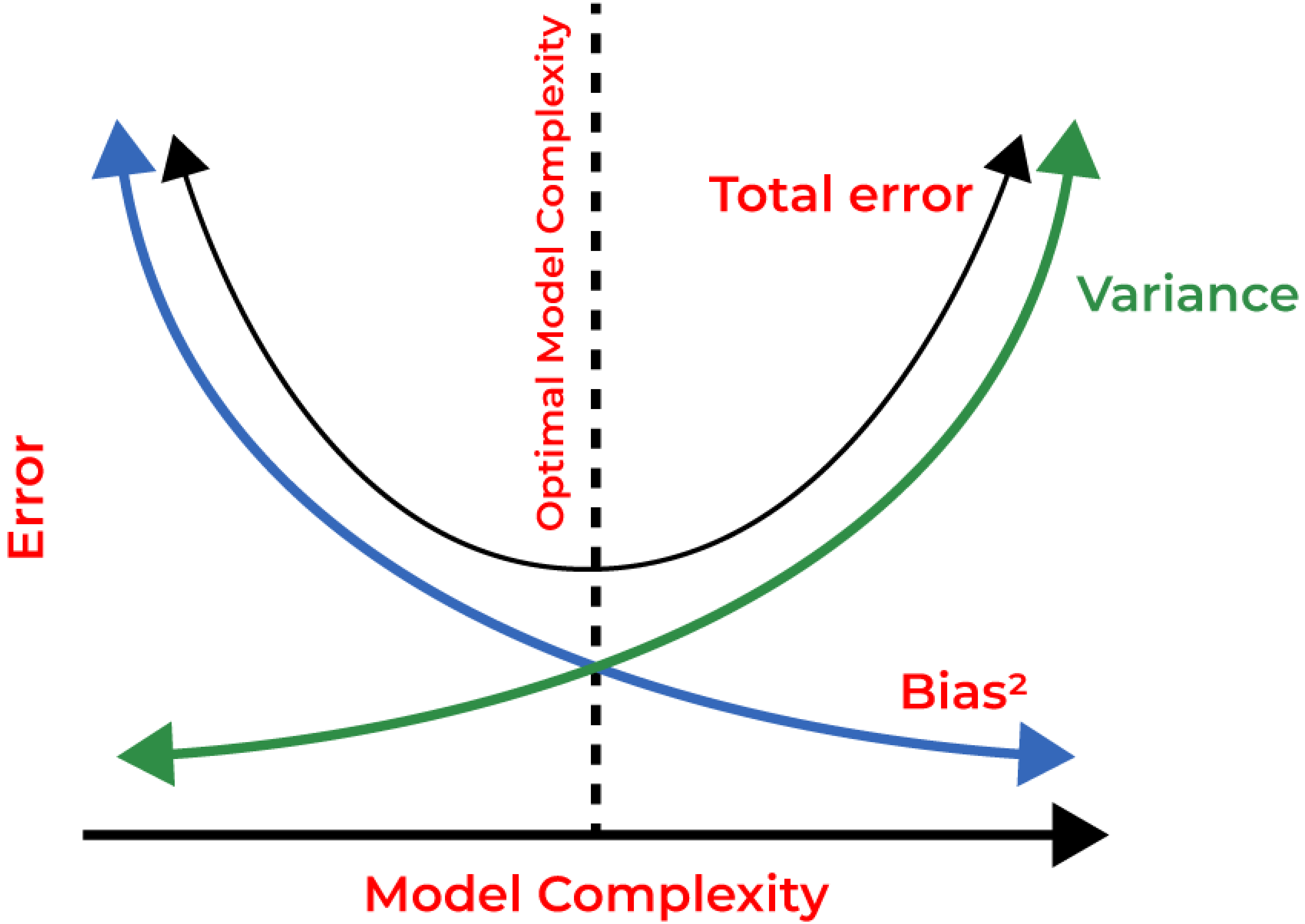
High Bias: High bias value means more assumptions are taken to build the target function. In this case, the model will not match the training dataset closely. (Underfitting)

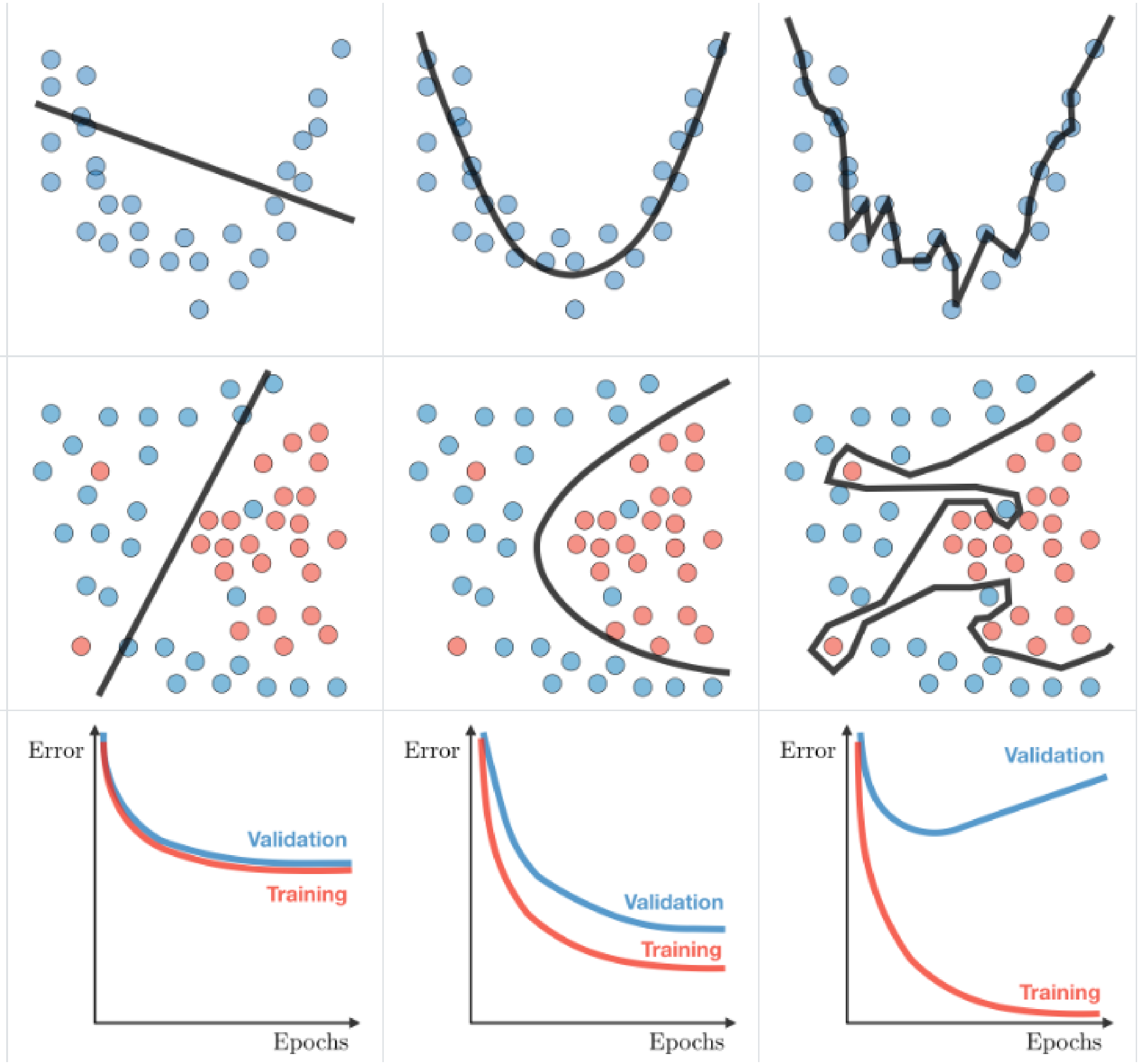
Variance

Variance is the measure of spread in data from its mean position.

Low variance: Low variance means that the model is less sensitive to changes in the training data and can produce consistent estimates of the target function with different subsets of data from the same distribution. This is the case of **underfitting** when the model fails to generalize on both training and test data.

High variance: High variance means that the model is very sensitive to changes in the training data and can result in significant changes in the estimate of the target function when trained on different subsets of data from the same distribution. This is the case of **overfitting** when the model performs well on the training data but poorly on new, unseen test data. It fits the training data too closely that it fails on the new training dataset.


Optimal Model Complexity
Total error
Variance
Error
Bias²
Model Complexity



Underfitting

- High training error
- Training error and validation error are close
- High bias

Overfitting

- Low training error
- Training error and validation error are far apart

Model Deployment

Model Deployment

Before	After
- Export the model - Build API - FAST API - Flask - Build Web App - Streamlit - Dash	- Security - Scalability - Monitoring - Logging - Versioning - Audit

Tools:

- Tensorflow Extended
- MLFlow
- Streamlit
- Dash

Thank You

Happy Learning :)