CSE 4553
Machine Learning
Lecture 4: Basic Practices in ML

Winter 2024

Dr. Hasan Mahmud | hasan@iut-dhaka.edu

Contents

- Feature Engineering
- Selection of ML algorithms
- Training, Validation, and Test set
- Underfitting and Overfitting
- Regularization
- ML Model Performance Analysis
  - Confusion Matrix
  - Accuracy
  - Precision, Recall
  - ROC, AUC curves
- Hyperparameter Tuning, Cross-Validation

Feature Engineering

- The problem of transforming raw data into a dataset is called feature engineering.
- Logs of user interaction with a computer system may contain the following features:
  - Price of the subscription
  - Frequency of connection per day, per week, and per year
  - Average session duration in seconds
  - Average response time and so on.
- Informative features: Help learning algorithm to build a model that predicts well labels of the data used for training.
- A model has a low bias when it predicts well the training data.

Feature Engineering…

- One-Hot Encoding
  - Used for categorical features when order of the feature is not important
  - E.g. $red = [1, 0, 0],$ $yellow = [0, 1, 0],$ $green = [0, 0, 1]$

Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

Human-Readable

Pet
Cat
Dog
Turtle
Fish
Cat

Machine-Readable

Cat	Dog	Turtle	Fish
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

Feature Engineering…

- Binning
  - Used to convert numerical feature in to categorical feature.
  - Binning (also called bucketing) is the process of converting a continuous feature into multiple binary features called bins or buckets, typically based on value range
  - Three common approach of binning: Equal width binning, Equal frequency binning, and a K-means approach
  - For example, instead of representing age as a single real-valued feature, the analyst could chop ranges of age into discrete bins: all ages between 0 and 5 years-old could be put into one bin, 6 to 10 years-old could be in the second bin, 11 to 15 years-old could be in the third bin, and so on.

Sorted Data of price (in dollars):
4, 8, 15, 21, 21, 24, 25, 28, 34

Distribute the sorted data into bins
Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing By Means
Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by Bin Boundaries
Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

Binning Methods for Smoothing Data

Feature Engineering…

- Normalization
  – Normalization is the process of converting an actual range of values which a numerical feature can take, into a standard range of values, typically in the interval [−1, 1] or [0, 1].

$$x_{\text{norm}} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

#	Emp	Age	Salary
1	Emp1	44	73000
2	Emp2	27	47000
3	Emp3	30	53000
4	Emp4	38	62000
5	Emp5	40	57000
6	Emp6	35	53000
7	Emp7	48	78000

Normalization

Age	Normalized Age	Salary	Normalized Salary
44	0.80952381	73000	0.838709677
27	0	47000	0
30	0.142857143	53000	0.193548387
38	0.523809524	62000	0.483870968
40	0.619047619	57000	0.322580645
35	0.380952381	53000	0.193548387
48	1	78000	1
	Range 0-1		**Range 0-1**

How to calculate Normalized value?
X = 35, min = 27, max = 48 for column Age.
Xnorm(for 35) = $\frac{35-27}{48-27}$ = 0.3809

More generally, the normalization formula looks like this:

$$\bar{x}^{(j)} = \frac{x^{(j)} - min^{(j)}}{max^{(j)} - min^{(j)}},$$

where $min^{(j)}$ and $max^{(j)}$ are, respectively, the minimum and the maximum value of the feature j in the dataset.

Feature Engineering…

- Standardization
  - Standardization (or z-score normalization) is the procedure during which the feature values are rescaled so that they have the properties of a standard normal distribution with $\mu = 0$ and $\sigma = 1$, where μ is the mean (the average value of the feature, averaged over all examples in the dataset) and σ is the standard deviation from the mean.
  - Standard scores (or z-scores) of features are calculated as follows:

$$\hat{x}^{(j)} = \frac{x^{(j)} - \mu^{(j)}}{\sigma^{(j)}}.$$

Feature Engineering…

- Dealing with missing features:

  The typical approaches of dealing with missing values for a feature include:

  - Removing the examples with missing features from the dataset. This can be done if your dataset is big enough so you can sacrifice some training examples.
  - Using a learning algorithm that can deal with missing feature values (depends on the library and a specific implementation of the algorithm).
  - Using a data imputation technique.

Feature Engineering…

- Data Imputation Techniques
- One technique consists in replacing the missing value of a feature by an average value of this feature in the dataset:

$$\hat{x}^{(j)} = \frac{1}{N} x^{(j)}.$$

- Another technique is to replace the missing value by the same value outside the normal range of values. For example, if the normal range is [0, 1], then you can set the missing value equal to 2 or −1.

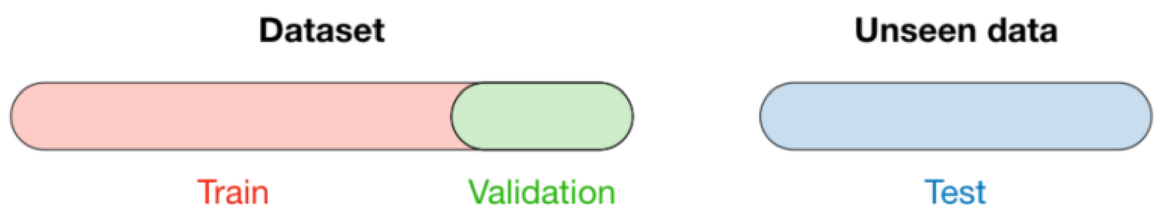
Learning Algorithm Selection

- Explainability
  - ML Models need to be explainable to the non-technical audience.
  - Neural Network/DNN VS KNN, linear regression/logistic regression
- In-memory vs out-of-memory
  - Can the dataset be loaded fully into the RAM or incremental learning procedure should be applied.
- Number of features and examples
- Categorical vs numerical features
- Nonlinearity of the data
- Training speed
- Prediction speed

Training set, validation set, test set

Training set	Validation set	Testing set
- Model is trained - Usually 80% of the dataset	- Model is assessed - Usually 20% of the dataset - Also called hold-out or development set	- Model gives predictions - Unseen data

Once the model has been chosen, it is trained on the entire dataset and tested on the unseen test set. These are represented in the figure below:

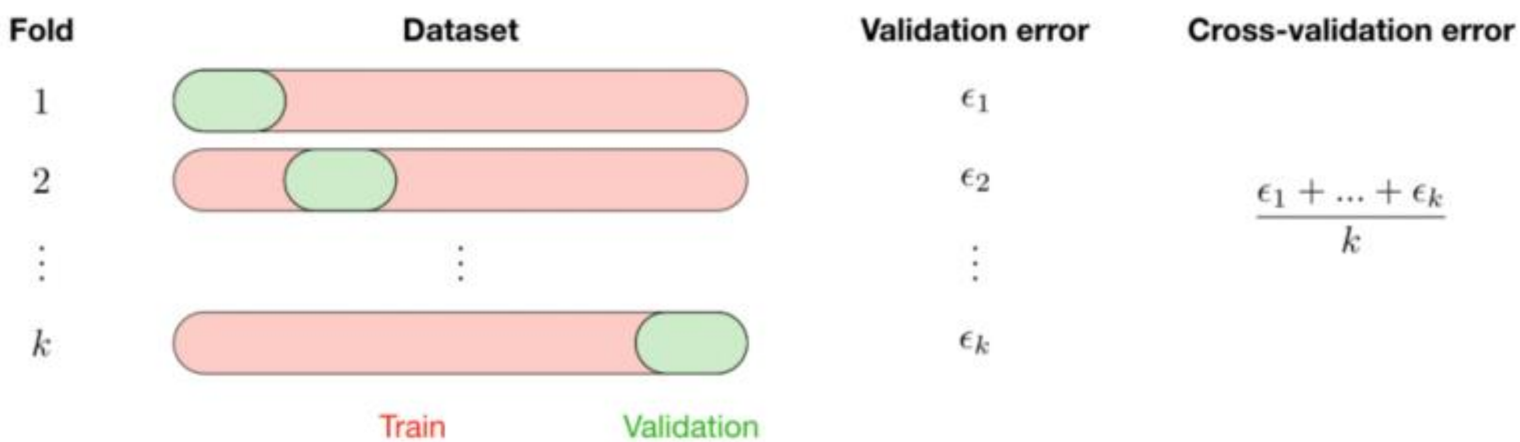


Cross-validation

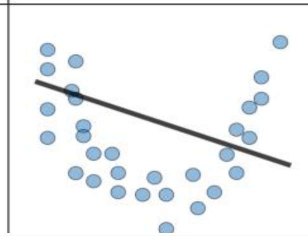
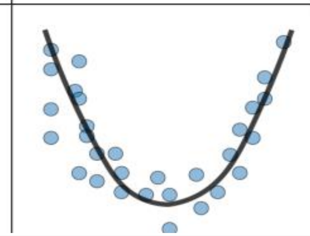
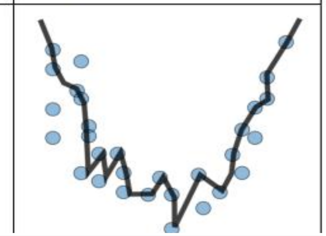
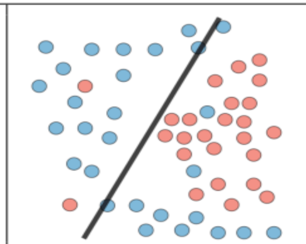
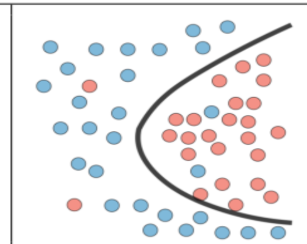
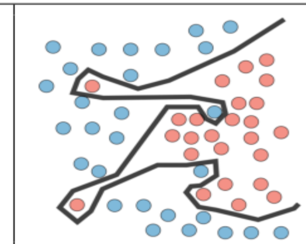
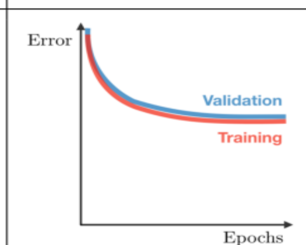
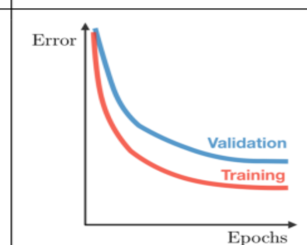
❒ **Cross-validation** – Cross-validation, also noted CV, is a method that is used to select a model that does not rely too much on the initial training set. The different types are summed up in the table below:

k-fold	**Leave-p-out**
- Training on $k-1$ folds and assessment on the remaining one - Generally $k=5$ or 10	- Training on $n-p$ observations and assessment on the p remaining ones - Case $p=1$ is called leave-one-out

The most commonly used method is called k-fold cross-validation and splits the training data into k folds to validate the model on one fold while training the model on the $k-1$ other folds, all of this k times. The error is then averaged over the k folds and is named cross-validation error.



Overfitting vs underfitting

	Underfitting	Just right	Overfitting
Symptoms	- High training error - Training error close to test error - High bias	- Training error slightly lower than test error	- Low training error - Training error much lower than test error - High variance
Regression			
Classification			
Deep learning			
Remedies	- Complexify model - Add more features - Train longer		- Regularize - Get more data

Model Performance Assessment

- Confusion Matrix

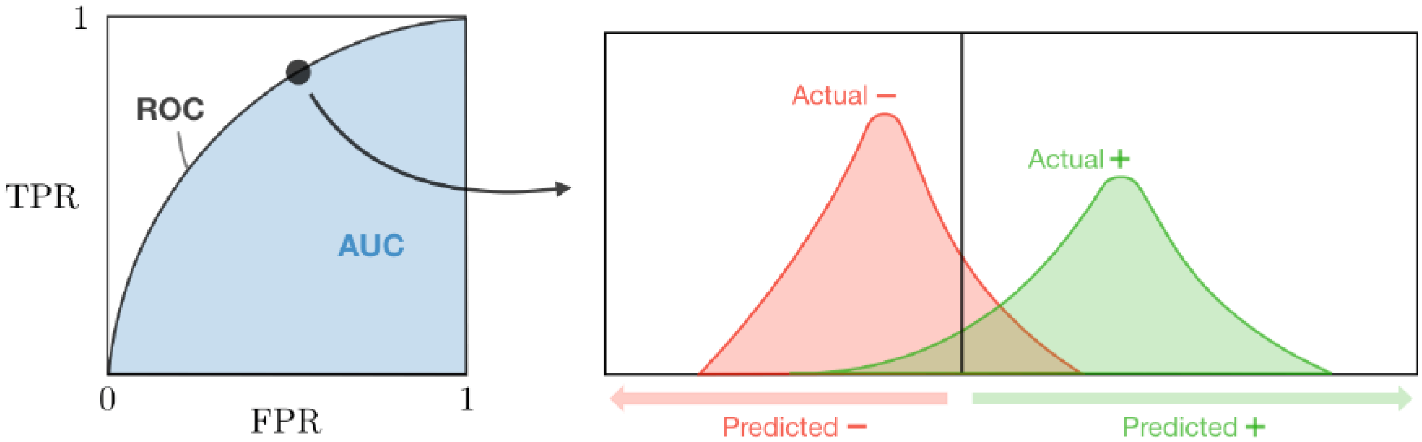
Metric	Formula	Interpretation
Accuracy	$\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}$	Overall performance of model
Precision	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$	How accurate the positive predictions are
Recall Sensitivity	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	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

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

ROC, AUC

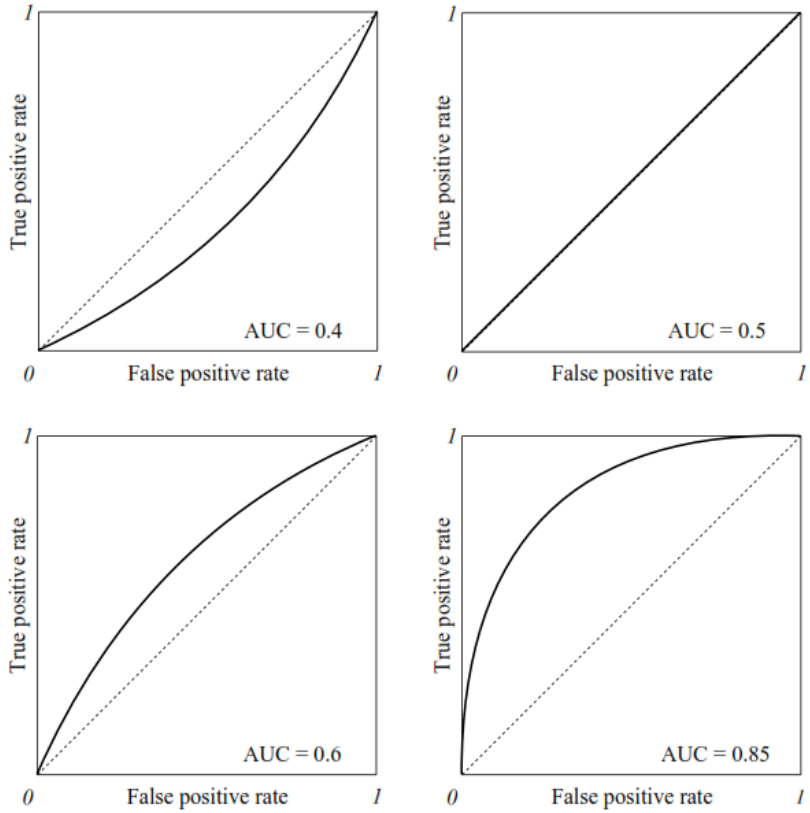
- ROC: The receiver operating curve, also noted ROC, is the plot of TPR versus FPR by varying the threshold.
- AUC: The area under the receiving operating curve, also noted AUC or AUROC

Metric	Formula	Equivalent
True Positive Rate TPR	$\frac{TP}{TP + FN}$	Recall, sensitivity
False Positive Rate FPR	$\frac{FP}{TN + FP}$	1-specificity



ROC, AUC

- ROC: The receiver operating curve, also noted ROC, is the plot of TPR versus FPR by varying the threshold.
- AUC: The area under the receiving operating curve, also noted AUC or AUROC



Hyperparameter Tuning

- Hyperparameters are defined as the parameters that are explicitly defined by the user to control the learning process.
- These are external to the model, and their values cannot be changed during the training process.
- Few examples:
  - The k in kNN or K-Nearest Neighbour algorithm
  - Learning rate for training a neural network
  - Train-test split ratio
  - Batch Size
  - Number of Epochs
  - Branches in Decision Tree
  - Number of clusters in Clustering Algorithm