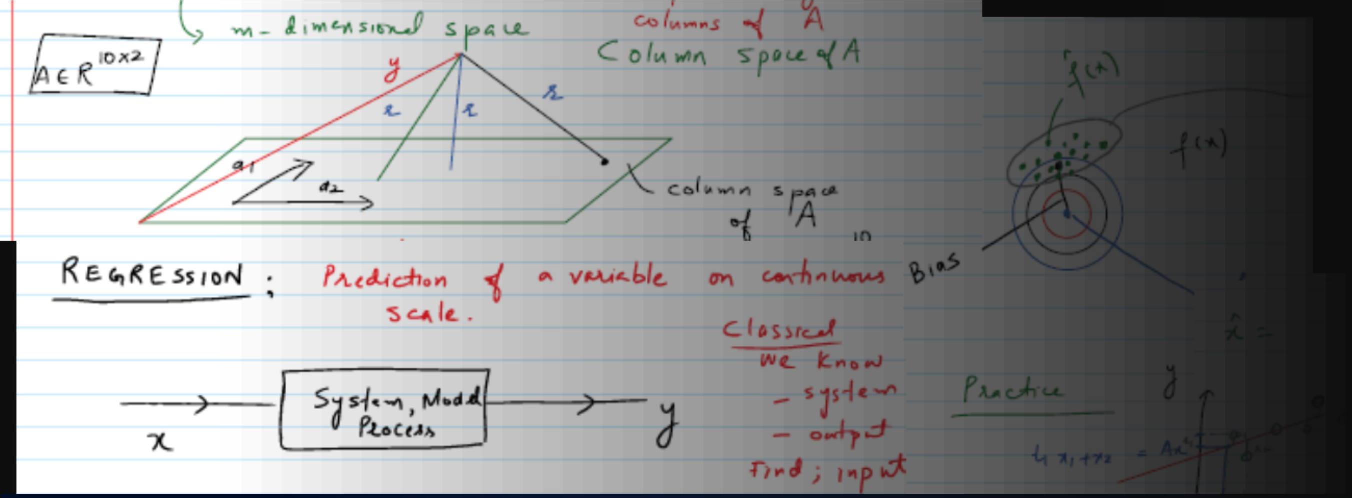


Machine Learning

EE514 – CS535

Overview

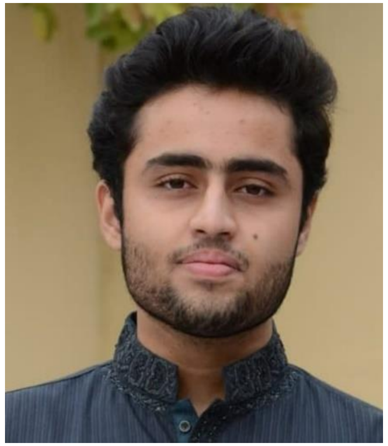
Zubair Khalid

School of Science and Engineering
Lahore University of Management Sciences

https://www.zubairkhalid.org/ee514_2023.html

LUMS
A Not-for-Profit University

About us!



Omer

Zubair

Ali

Sharjeel

Abdul Muizz

Fatima

LUMS
A Not-for-Profit University

About the Instructor

- Associate Professor, LUMS
- Post-doctorate – 2013-2015, Australian National University (ANU)
- PhD, Australian National University (ANU) - 2013

Affiliations:

- CITY – Centre for Urban Informatics, Technology and Policy (www.city.lums.edu.pk)
- Applied Signal Processing Group, ANU
- Smart Data, Systems and Applications Lab (www.sdsa.lums.edu.pk)

Collaborations: Princeton, UCL, University of Edinburgh, EPFL, ANU, KAUST

PhD Students: 7 (5 graduated)

Publications: More than 75 (23 Transactions/Journals, 53 Conference proceedings)

Service: Senior Member IEEE and Associate Editor, IEEE Signal Processing Letters

What is this course about?

Introductory course in Machine Learning (ML) – Fundamental topics in

- Supervised learning
- Unsupervised learning

Course Objectives:

- To provide a thorough introduction to ML methods
- To build mathematical foundations of ML and provide an appreciation for its applications
- To provide experience in the implementation and evaluation of ML algorithms
- To develop research interest in the theory and application of ML

Is this course a right choice for you?

<u>Undergraduate students</u>

- *Interested in pursuing AI, Deep Learning and/or Machine Learning in their **grad school***
- *Interesting in pursuing a **professional career** focused on the development of Machine Learning solutions*

<u>Graduate students</u>

- *Want to do fundamental research in the area of Machine Learning*
- *Wish to apply Machine Learning in their research work*

Course Prerequisites

Undergraduate students

- Linear Algebra (MATH120)
- Probability (MATH230, DISC203, CS501)
- Programming (CS200, EE201)

Graduate students

- Encouraged to revise Linear Algebra and Probability concepts (on-the-fly)

We expect all the students to have good programming skills (in C/Python/MATLAB)

Note on Assignment 0!

Learning Interface

Communication:

Course Page: https://www.zubairkhalid.org/ee514_2023.html

Slack: Course-related questions or discussions. We will try to respond to the queries ASAP.

Office Hours: Posted on course page; distributed throughout the week

Email Policy:

Subject:

- 'ML-URGENT-Assignment Clarification'
- 'ML-NOT URGENT-Extend Assignment deadline'

Please **do not** email to verify whether we have received your submission via LMS or the submission is late due to last-minute connectivity issues.

Grading Distribution

- Programming Assignments and Homeworks: 35%
  - 5 Programming Assignments
  - 3 Homeworks
- Quizzes: 15% (Almost every week)
- Project: 20%
- Final Exam: 30%

Course Polices

- **Homework Late Policy**
  - 10% per day for 3 days. No submission after 3 days (72 hours)
- **Missed Quiz Policy**
  - No make-up for quiz
- ***Plagiarism will be strictly dealt with as per university policies (take it seriously).***
- *Zero Tolerance for Plagiarism and Cheating*
- Re-grading can be requested after grade reporting, within the following time limits:
  - HW and Assignments: 2 days
  - Final Exam: 3 days

Course Polices

Harassment Policy

Harassment of any kind is **unacceptable**, whether it be sexual harassment, online harassment, bullying, coercion, stalking, verbal or physical abuse of any kind. Harassment is a very broad term; it includes both direct and indirect behaviour, it may be physical or psychological in nature, it may be perpetrated online or offline, on campus and off campus. It may be one offense, or it may comprise of several incidents which together amount to sexual harassment. It may include overt requests for sexual favours but can also constitute verbal or written communication of a loaded nature. Further details of what may constitute harassment may be found in the LUMS Sexual Harassment Policy, which is available as part of the university code of conduct.

LUMS has a Sexual Harassment Policy and a Sexual Harassment Inquiry Committee (SHIC). Any member of the LUMS community can file a formal or informal complaint with the SHIC. If you are unsure about the process of filing a complaint, wish to discuss your options or have any questions, concerns, or complaints, please write to the Office of Accessibility and Inclusion (OAI, oai@lums.edu.pk) and SHIC (shic@lums.edu.pk) —both of them exist to help and support you and they will do their best to assist you in whatever way they can.

To file a complaint, please write to harassment@lums.edu.pk.

Course Polices

Help related to equity and Belonging at SSE

SSE's Council on Equity and Belonging is committed to devising ways to provide a safe, inclusive, and respectful learning, living, and working environment for its students, faculty, and staff.
For help related to any such issue, please feel free to write to any member of the school council for help or feedback.

Mental Health Support at LUMS

-

For matters relating to counselling, kindly email student.counselling@lums.edu.pk, or visit https://osa.lums.edu.pk/content/student-counselling-office for more information.

You are welcome to write to me or speak to me if you find that your mental health is impacting your ability to participate in the course. However, should you choose not to do so, please contact the Counselling Unit and speak to a counsellor or speak to the OSA team and ask them to write to me so that any necessary accommodations can be made.

Modules

1- ML Overview

Course Overview, notation
Supervised Learning Setup

Weeks: 1,2

Components:

- Programming Assignment 1: Intro to Python, Setting up Environment

Modules

2 - Classification

Classification
KNN
Evaluation Metrics, Curse of Dimensionality
Multi-class Classification

Weeks: 3,4

Components:

- Programming Assignment 2: KNN based (Using Images)
- Homework 1A

Modules

3 - Regression

Linear Regression
Gradient Descent
Multi-variate Regression
Polynomial Regression
Bias-Variance Trade-off, Regularization

Weeks: 4,5

Components:

- Programming Assignment 3: Regression
- Homework 1B

Modules

4 - Logistic Regression

Logistic Regression

Weeks: 6

Components:

- Programming Assignment 4: Logistic Regression

Modules

5 – Bayesian Framework

Bayes Theorem
Naive Bayes Classification

Weeks: 7,8

Components:

- Programming Assignment 5: Naïve Bayes Classifier (may be merged with Assignment 4)
- Homework 2

Modules

6 – Perceptron, SVM and Neural Network

Perceptron Algorithm
SVM
Neural Networks

Weeks: 9,10,11,12

Components:

- Programming Assignment 6: Neural Networks
- Homework 3

Modules

7 – Clustering

Unsupervised Learning Overview
Clustering (k-means)

Weeks: 13,14

Components:

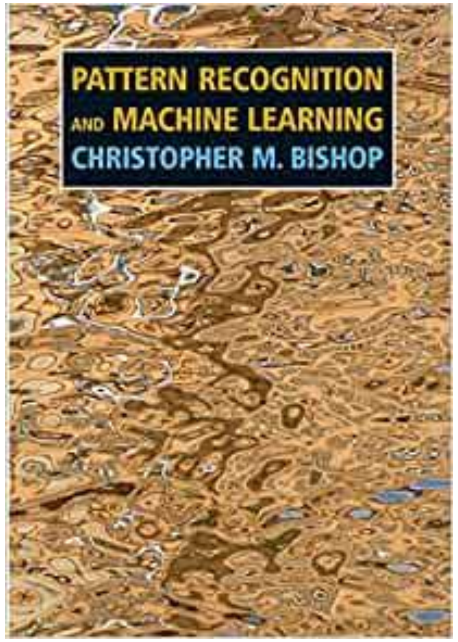
- Homework 3

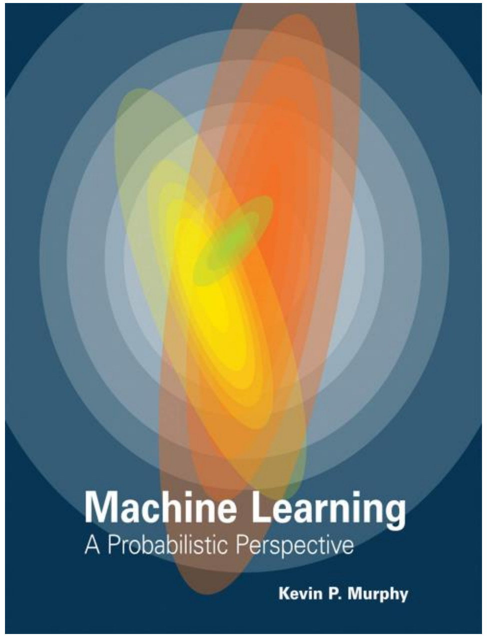
Modules

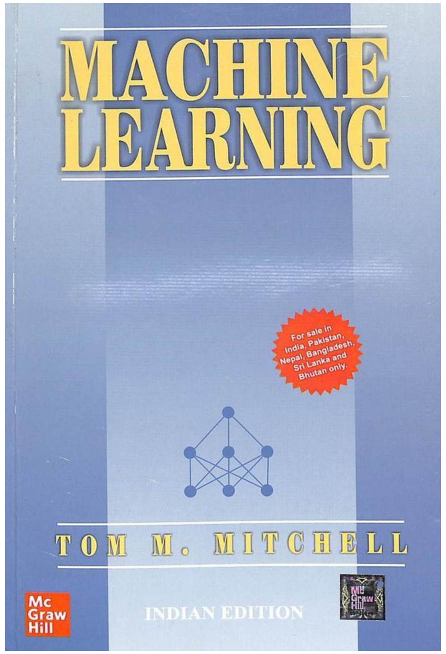
8 – Further Topics

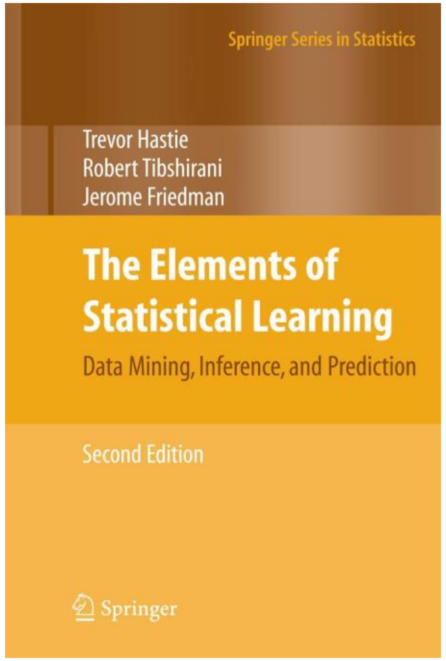
Feature Engineering, Dimensionality Reduction
Kernel Methods and Gaussian Process

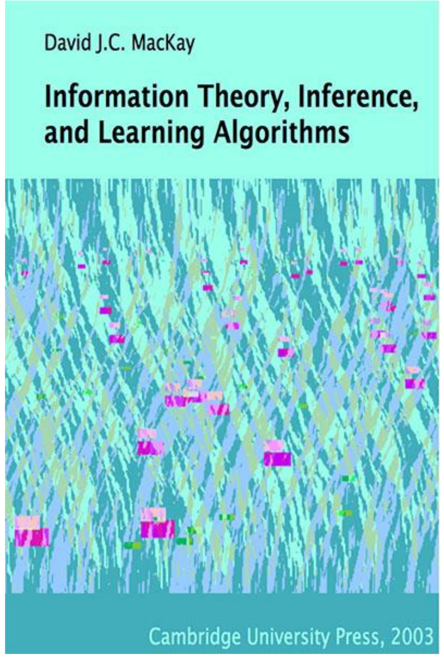
Suggested Reference Books











- (CB) Pattern Recognition and Machine Learning, Christopher M. Bishop
- (KM) Machine Learning: a Probabilistic Perspective, Kevin Murphy
- (TM) Machine Learning, Tom Mitchell
- (HTF) The Elements of Statistical Learning: Data mining, Inference, and Prediction, by Hastie, Tibshirani, Friedman
- (DM) Information Theory, Inference, and Learning Algorithms, David Mackay
- Lecture Notes/Slides will be shared.

“As to methods, there may be a million and then some, but principles are few. The man who grasps principles can successfully select his own methods.”

Ralph Waldo Emerson

Machine Learning Overview

What is Machine Learning?

"The activity or process of gaining knowledge or skill by studying, practicing, being taught, or experiencing something."

Merriam Webster dictionary

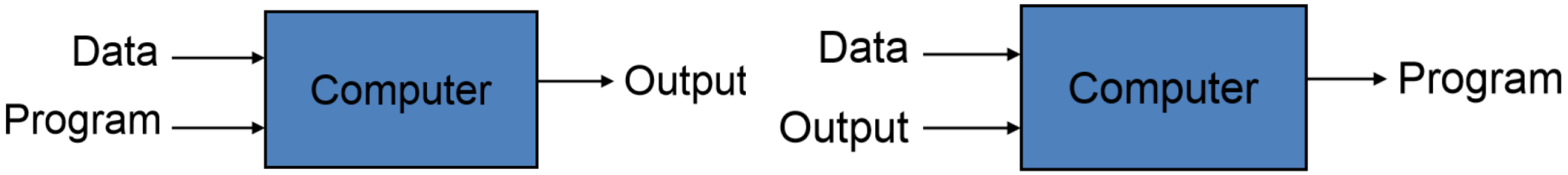
"A computer program is said to learn from experience **E** *with respect to some class of tasks* **T** *and performance measure* **P**, *if its performance at tasks in* **T**, *as measured by* **P**, *improves with experience* **E**."

Tom Mitchell

Machine Learning Overview

What is Machine Learning?

- *Automating the process of automation*
- *Getting computers to program themselves*

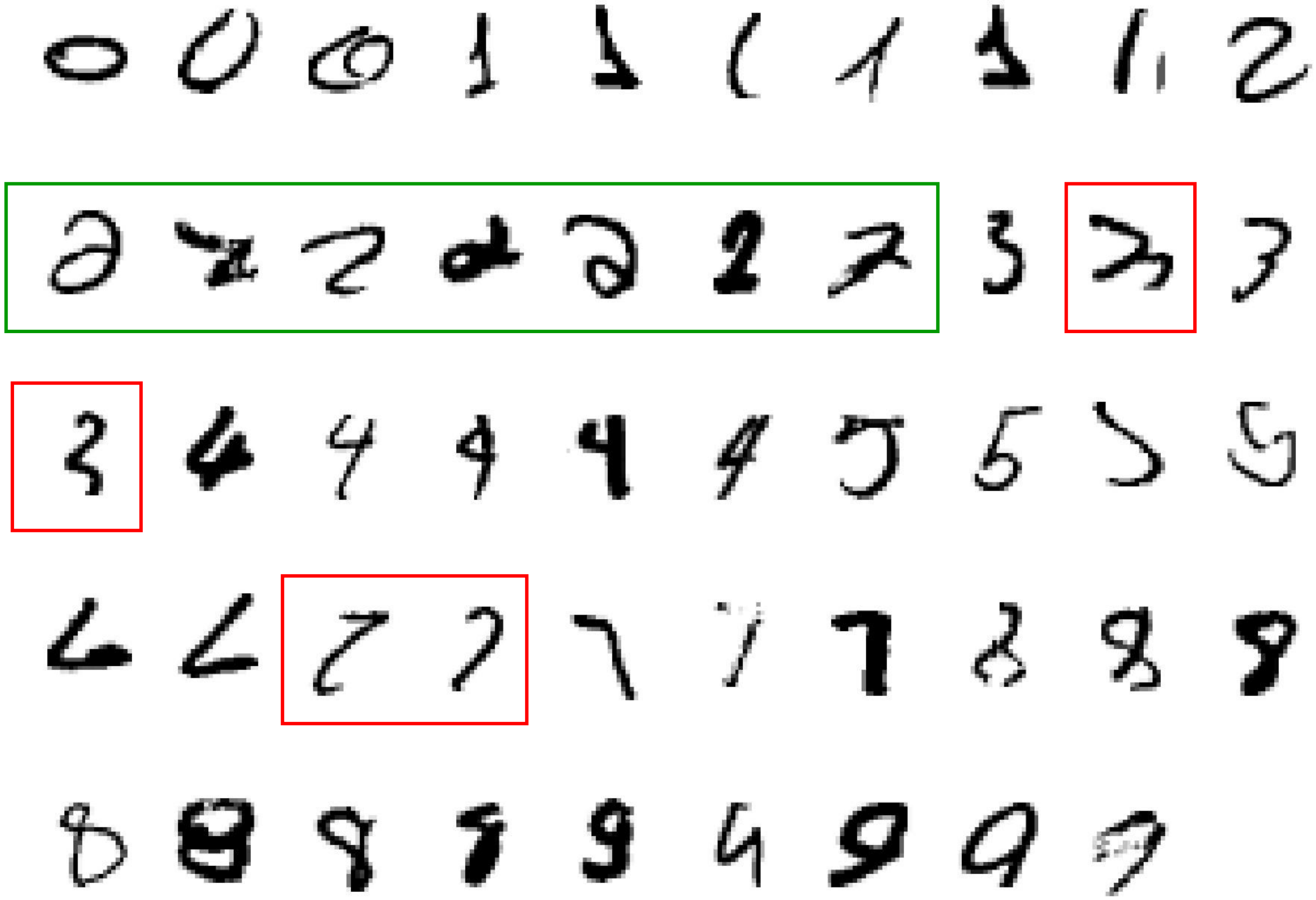


Traditional Programming **Machine Learning**

Given examples (training data), make a machine learn system behavior or discover patterns

Machine Learning Overview

Classical Example: Recognize hand-written 2!

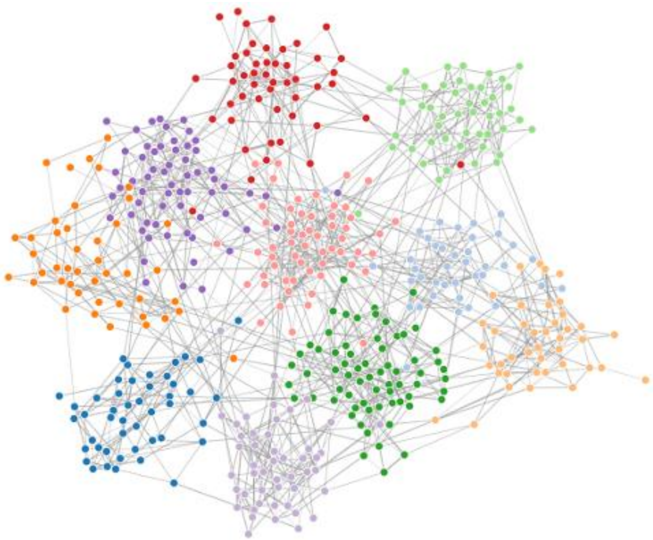


Machine Learning: Overview

What is Machine Learning?

Given examples (training data), make a machine learn system behavior or discover patterns

Data



Given to us

Algorithm

We need to design it

Model

$$f(\mathbf{x})$$

Final output which enables us to make predictions

Machine Learning: Overview

Algorithms vs Model

- *Linear regression algorithm* produces a model, that is, *a vector of values of the coefficients of the model.*

- *Decision tree algorithm* produces a model comprised of *a tree of if-then statements with specific values.*

- *Neural network along with backpropagation + gradient descent:* produces a model comprised of *a trained (weights assigned) neural network.*

Machine Learning: Overview

Example Applications

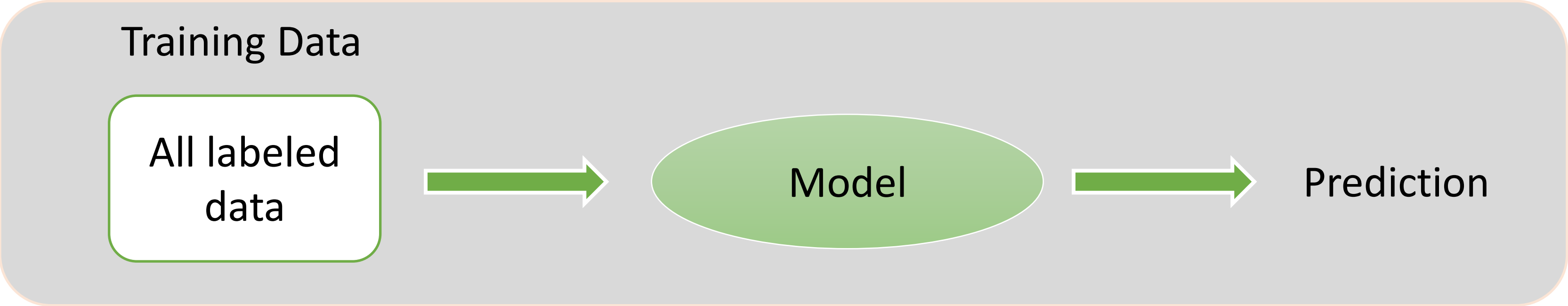
- Medical Diagnosis
- Autonomous Driving
- Information extraction
- Computer/Machine Vision
- Finance
- Web Search
- Robotics
- Social networks
- Production Industry
- Logistics
- Waste Management
- [Your research/favorite area]

Machine Learning: Overview

Nature of ML Problems

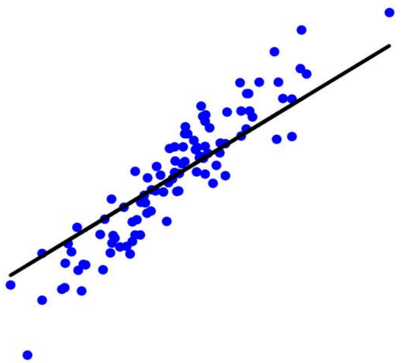
1. Supervised Learning

The learning algorithm would receive a set of inputs along with the corresponding correct outputs to train a model



Supervised Learning

Regression



Regression: Quantitative Prediction on a continuous scale

Examples: Prediction of

- Age of a person from his/her photo
- Price of 10 Marla, 5-bedroom house in 2050
- USD/PKR exchange rate after one week
- Efficacy of vaccine or medicine
- Average temperature/Rainfall during monsoon
- Cumulative score in ML course
- Probability of decrease in the electricity prices in Pakistan
- No. of steps per day

What do all these problems have in common?

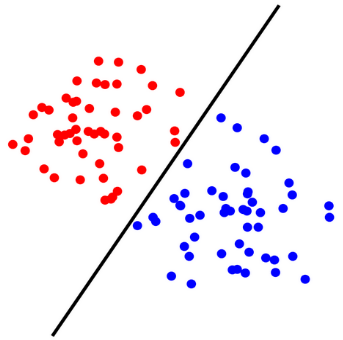
Continuous outputs

Predicting continuous outputs is called regression

Supervised Learning

Classification

Classification: *Given a data sample, predict its class* **(discrete)**



Examples: Prediction of

- *Gender of a person using his/her photo or hand-writing style*
- *Spam filtering*
- *Object or face detection in a photo*
- *Temperature/Rainfall normal or abnormal during monsoon*
- *Letter grade in ML course*
- *Decrease expected in electricity prices in Pakistan next year*
- *More than 10000 Steps taken today*

What do all these problems have in common?

Discrete outputs: Categorical

Yes/No (Binary Classification)

Multi-class classification: multiple classes

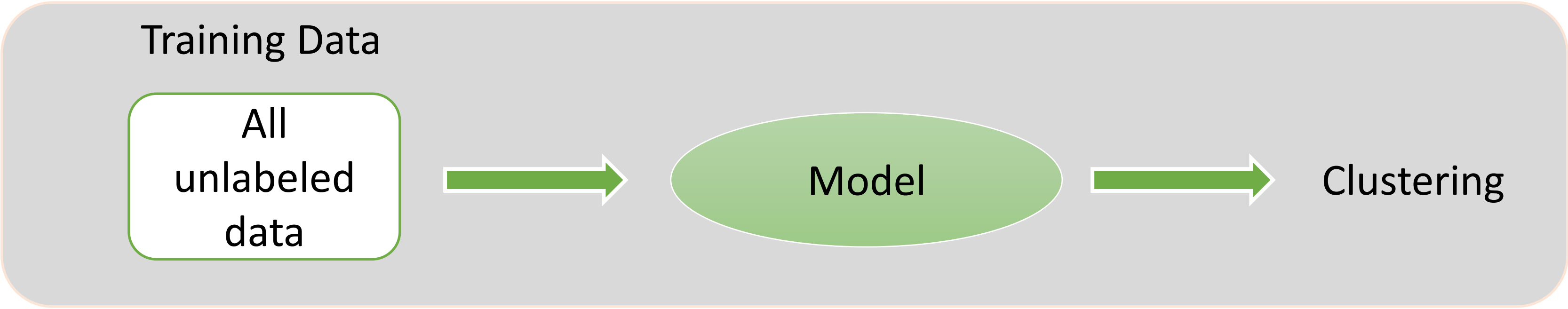
Predicting a categorical output is called classification

Machine Learning: Overview

Nature of ML Problems

2. Unsupervised Learning

The learning algorithm would receive unlabeled raw data to train a model and to find patterns in the data

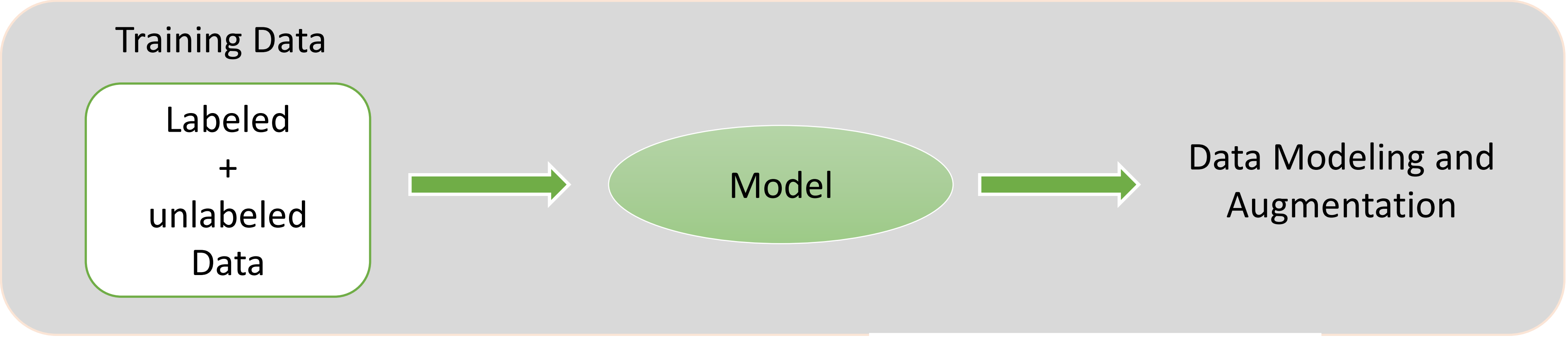


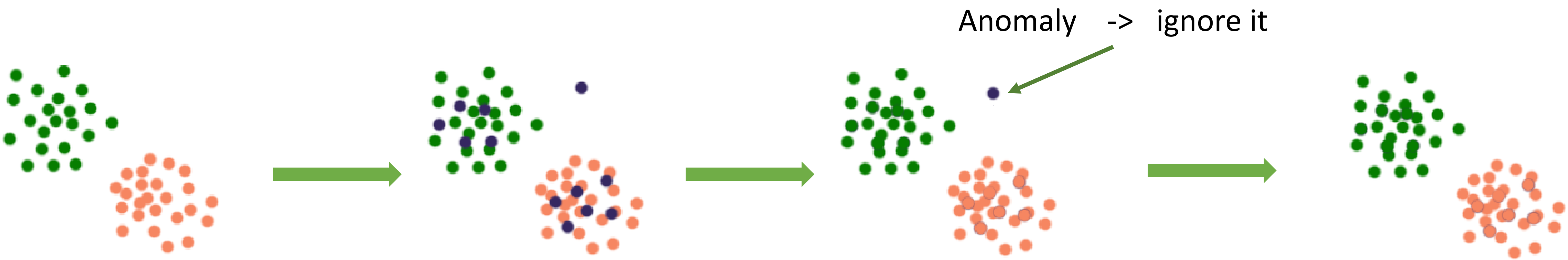
Machine Learning: Overview

Nature of ML Problems

3. Semi-supervised Learning

- *The learning algorithm receives labeled and unlabeled raw data to train a model*
- *Main objective is to efficiently accommodate the unlabeled data*





Use labeled data to build a model

Place unlabeled data with model

Use the model to label the unlabeled data

Fit the model again with the combined data

Machine Learning: Overview

Training Data Collection

$\mathbf{x}$

Input

→ ***Model or Process or System*** →

y

Observed Output

PROCESS or ***SYSTEM*** : Underlying physical or logical phenomenon which maps our input data to our observed output

Collect the training data by observing our unknown **PROCESS** or **SYSTEM**

Machine Learning: Overview

Example Systems

- Previous Sales
- Prices
- Inflation
- Pandemic

→ ***Model*** → Future sales

Image → ***Model*** → Object detection Or recognition

Machine Learning: Overview

Example Systems

Consumer Transaction Data → ***Model*** → Market segmentation based on consumers' spending patterns

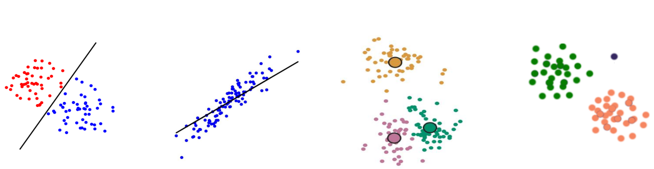
- Income
- Credit History
- Employment
- Marital Status

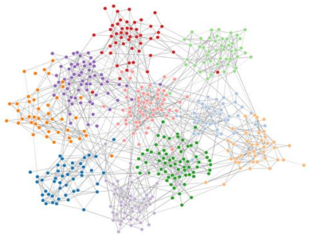
→ ***Model*** → Loan Approval

Machine Learning: Overview

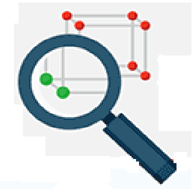
Typical Flow

Problem Nature Identification → **Training Data Collection** → **Model Training** → **Model Validation** → **Model Deployment**





$f(\mathbf{x})$



Supervised Learning Setup

Nomenclature

In these regression or classification problems, we have

- Inputs – referred to as Features
- Output – referred to as Label
- Training data – (input, output) for which the output is known and is used for training a model by ML algorithm
- A Loss, an objective or a cost function – determines how well a trined model approximates the training data
- Test data – (input, output) for which the output is known and is used for the evaluation of the performance of the trained model

Supervised Learning Setup

Nomenclature - Example

Predict Stock Index Price

- *Features (Input)*
- *Labels (Output)*
- *Training data*

Interest_Rate	Unemployment_Rate	Stock_Index_Price
2.75	5.3	1464
2.5	5.3	1394
2.5	5.3	1357
2.5	5.3	1293
2.5	5.4	1256
2.5	5.6	1254
2.5	5.5	1234
2.25	5.5	1195
2.25	5.5	1159
2.25	5.6	1167
2	5.7	1130
2	5.9	1075
2	6	1047
1.75	5.9	965
1.75	5.8	943
1.75	6.1	958
1.75	6.2	971
1.75	6.1	949
1.75	6.1	884
1.75	6.1	866
1.75	5.9	876
1.75	6.2	?
1.75	6.2	?
1.75	6.1	?

Supervised Learning Setup

Formulation

We assume that we have d columns (features) of the input. In this example, we have two features; interest rate and unemployment rate, that is, $d = 2$.

In general, we use $\mathbf{x_i}$ to refer to featues of the i-th sample, that is,

$$\mathbf{x_i} = [x_{i,1}, x_{i,2}, x_{i,3}, \ldots x_{i,d}]$$

If y_i is the label associated with the i-th sample $\mathbf{x}_i$, we formulate training data in pairs as

$$(\mathbf{x_i}, y_i), \quad i = 1, 2, \ldots, n$$

Here, n denotes the number of samples in the training data. In this example, we have $n = 21$

Interest_Rate	Unemployment_Rate	Stock_Index_Price
2.75	5.3	1464
2.5	5.3	1394
2.5	5.3	1357
2.5	5.3	1293
2.5	5.4	1256
2.5	5.6	1254
2.5	5.5	1234
2.25	5.5	1195
2.25	5.5	1159
2.25	5.6	1167
2	5.7	1130
2	5.9	1075
2	6	1047
1.75	5.9	965
1.75	5.8	943
1.75	6.1	958
1.75	6.2	971
1.75	6.1	949
1.75	6.1	884
1.75	6.1	866
1.75	5.9	876
1.75	6.2	?
1.75	6.2	?
1.75	6.1	?

Supervised Learning Setup

Formulation

Using the adopted notation, we can formalize the supervised machine learning setup. We represent the entire training data as

$$D = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \ldots, (\mathbf{x_n}, y_n)\} \subseteq \mathcal{X}^d \times \mathcal{Y}$$

Here $\mathcal{X}^d$ - d dimensional feature space and $\mathcal{Y}$ is the label space.

Regression: $\mathcal{Y} = \mathbf{R}$ (prediction on continuous scale)

Classification: $\mathcal{Y} = \{0, 1\}$ or $\mathcal{Y} = \{-1, 1\}$ or $\mathcal{Y} = \{1, 2\}$ (Binary classification)

$\mathcal{Y} = \{1, 2, \ldots, M\}$ (M-class classification)

Supervised Learning Setup

Example

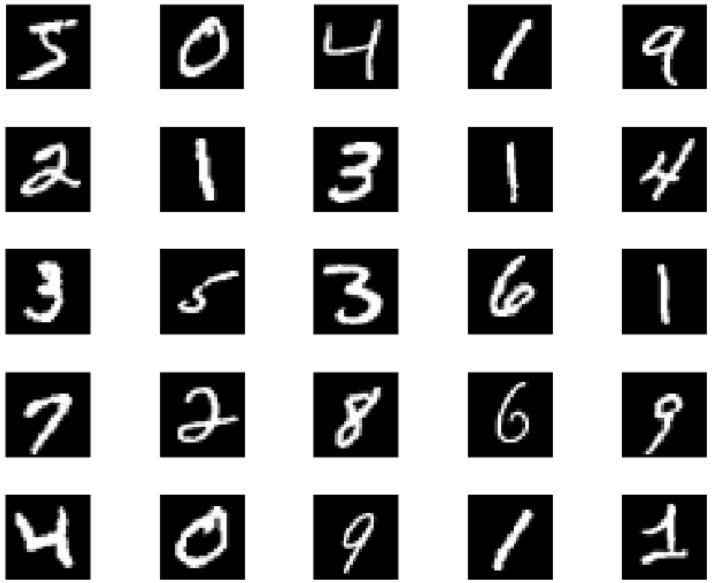
Data of 200 Patients:

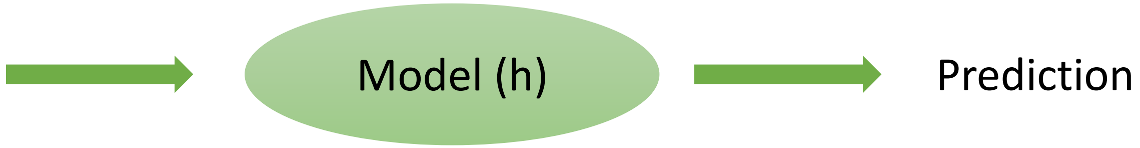
- Age of the patient
- Cholesterol levels
- Glucose levels
- BMI
- Height
- Heart Rate
- Calories intake
- No. of steps taken

→ Model (h) → Prediction of Oxygen Saturation

Supervised Learning Setup

Example





MNIST Data:

- Each sample 28x28 pixel image
- 60,000 training data
- 10,000 testing data

Supervised Learning Setup

Learning

Recall a problem in hand. We want to develop a model that can predict the label for the input for which label is unknown.

We assume that the data points $(\mathbf{x_i}, y_i)$ are drawn from some (unknown) distribution $P(X, Y)$.

Our goal is to learn the machine (model, function or hypothesis) h such that for a new pair $(\mathbf{x}, y)$ P, we can use h to obtin

$$h(\mathbf{x}) = y$$

with high probability or

$$h(\mathbf{x}) \approx y$$

in some optimal sense.

Supervised Learning Setup

Hypothesis Class

We call the set of possible functions or candidate models (linear model, neural network, decision tree, etc.) "the hypothesis class".

Denoted by $\mathcal{H}$

For a given problem, we wish to select hypothesis (machine) $h \in \mathcal{H}$.

Q: How?

A: Define hypothesis class $\mathcal{H}$ for a given learning problem.

Evaluate the performance of each candidate function and choose the best one.

Supervised Learning Setup

Q: How do we evaluate the performance?

A: Define a loss function to quantify the accuracy of the prediction.

Loss Function

Loss function should quantify the error in predicting y using hypothesis function h and input $\mathbf{x}$.

Denoted by $\mathcal{L}$.

Supervised Learning Setup

0/1 Loss Function:

Zero-one loss is defined as

$$\mathcal{L}_{0/1}(h) = \frac{1}{n}\sum_{i=1}^{n} 1 - \delta_{h(\mathbf{x_i})-y_i}$$

Here $\delta_{h(\mathbf{x_i})-y_i}$ is the delta function defined as

$$\delta_k = \begin{cases} 1, & k = 0 \\ 0 & \text{otherwise} \end{cases}$$

Interpretation:

- *Note normalization by the number of samples. This makes it the loss per sample.*
- *Loss function counts the number of mistakes made by hypothesis function on D.*
- ***Not used frequently due to non-differentiability and non-continuity.***

Supervised Learning Setup

Squared Loss Function:

Squared loss is defined as (also referred to as **mean-square error, MSE**)

$$\mathcal{L}_{\text{sq}}(h) = \frac{1}{n}\sum_{i=1}^{n}\left(h(\mathbf{x_i}) - y_i\right)^2$$

Interpretation:

- Again note normalization by the number of samples.
- Loss grows quadratically with the absolute error amount in each sample.

Root Mean Squared Error (RMSE):

RMSE is just square root of squared loss function: $$\mathcal{L}_{\text{rms}}(h) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(h(\mathbf{x_i}) - y_i\right)^2}$$

Supervised Learning Setup

Absolute Loss Function:

Absolute loss is defined as

$$\mathcal{L}_{\text{abs}}(h) = \frac{1}{n}\sum_{i=1}^{n} |h(\mathbf{x_i}) - y_i|$$

Interpretation:

- Loss grows linearly with the absolute of the error in each prediction.
- Used in regression and suited for noisy data.

* All of the losses are non-negative

Supervised Learning Setup

Learning

We wish to select hypothesis (machine) $h \in \mathcal{H}$ such that

$$h^* = \min_{h \in \mathcal{H}} \mathcal{L}(h)$$

(Optimization problem)

Recall We assume that the data points $(\mathbf{x_i}, y_i)$ are drawn from some $\boxed{\text{(unknown)}}$ distribution $P(X, Y)$.

We can come up with a function h after solving this minimization problem that gives low loss on our data.

Q: How can we ensure that hypothesis *h* will give low loss on the input not in *D*?

Supervised Learning Setup

To illustrate this, let us consider a model h trained on every input in D, that is, giving zero loss. Such function is referred to as memorizer and can be formulated as follows

$$h(\mathbf{x}) = \begin{cases} y_i, & \exists\ (\mathbf{x}_i, y_i) \in D, \quad \mathbf{x}_i = \mathbf{x}, \\ 0, & \text{otherwise} \end{cases}$$

Interpretation:

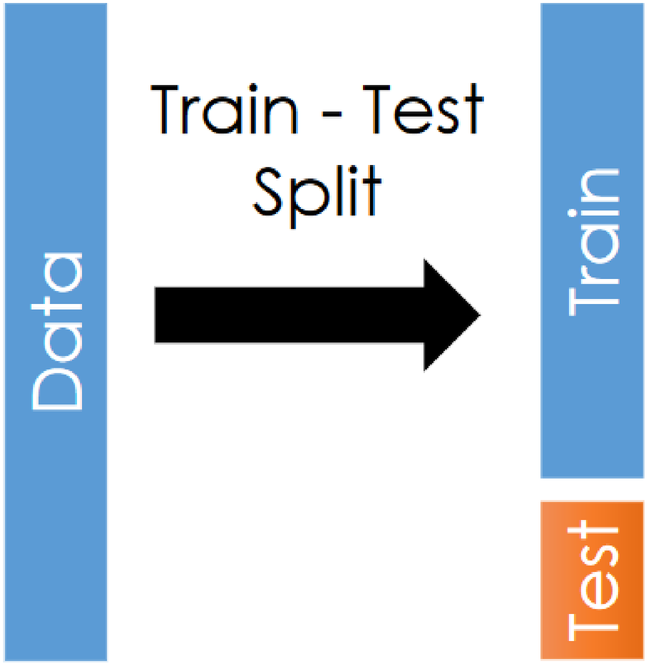
- *0% loss error on the training data (Model is fit to every data point in D).*
- *Large error for some input not in D*
- *First glimpse of **overfitting.***

Revisit:

Q: How can we ensure that hypothesis *h* will give low loss on the input not in *D*?

A: Train/Test Split

Supervised Learning Setup



Generalization: The Train-Test Split

To resolve the overfitting issue, we usually split D into train and test subsets:

- D_{TR} as the training data, (70, 80 or 90%)
- D_{TE} as the test data, (30, 20, or 10%)

How to carry out splitting?

- Split should be capturing the variations in the distribution.
- Usually, we carry out splitting using i.i.d. sampling and time series with respect to time

You can only use the test dataset once after deciding on the model using training dataset

Supervised Learning Setup

Learning (Revisit after train-test split)

We had the following optimiztion problem as

$$h^* = \min_{h \in \mathcal{H}} \mathcal{L}(h)$$

We generalize it as

$$h^* = \min_{h \in \mathcal{H}} \frac{1}{|D_{\mathrm{TR}}|} \sum_{(\mathbf{x},y) \in D_{\mathrm{TR}}} \mathcal{L}(\mathbf{x}, y)|h)$$

Evaluation

Loss on the testing data is given by

$$\epsilon_{\mathrm{TE}} = \frac{1}{|D_{\mathrm{TE}}|} \sum_{(\mathbf{x},y) \in D_{\mathrm{TE}}} \mathcal{L}(\mathbf{x}, y)|h*)$$

Supervised Learning Setup

Generalization loss

We define the generalized loss on the distribution P from which the D is drawn as the expected value (average value, probability weighted average to be precise) of the loss for a given h^* s

$$\epsilon = E\left[\mathcal{L}(\mathbf{x}, y|h^*)\right]$$

The expectation here is over the distribution P of $(\mathbf{x}, y)$.

Under the assumption that data D is i.i.d (independent and identically distributed) drawn from P, ϵ_{TE} serves as an unbiased estimator of the generalized loss ϵ. This simply means ϵ_{TE} converges to ϵ with the increase in the data size, that is,

$$\lim_{n\to\infty} \epsilon_{\mathrm{TE}} = \epsilon.$$

Supervised Learning Setup

Generalization: The Train-Test Split

At times, we usually split D into three subsets, that is, the training data is further divided into traaining and validation datasets:

- D_{TR} as the training data, (80%)
- D_{VA} as the validation data, (10%)
- D_{TE} as the test data, (10%)

Q: Idea:

Validation data is used to evaluate the loss for a function h that is determined using the learning on the training data-set. If the loss on validation data is high for a given h, the hypothesis or model needs to be changed.

Supervised Learning Setup

Generalization: The Train-Test Split

More explanation* to better understand the difference between validation and test data:

- **Training set:** A set of examples used for learning, that is to fit the parameters of the hypothesis (model).

- **Validation set:** A set of examples used to tune the hyper-parameters of the hypothesis function, for example to choose the number of hidden units in a neural network OR the order of polynomial approximating the data.

- **Test set:** A set of examples **used** only to assess the performance of a fully-specified model or hypothesis.

Adapted from *Brian Ripley, Pattern Recognition and Neural Networks, 1996

Supervised Learning Setup

Reference:

- CB: sec 1.1
- HTF section 2.1
- KM: sec. 1.1, 1.2