Basics: Machine Learning

Week 1

Dr. Muhammad Nouman Durrani

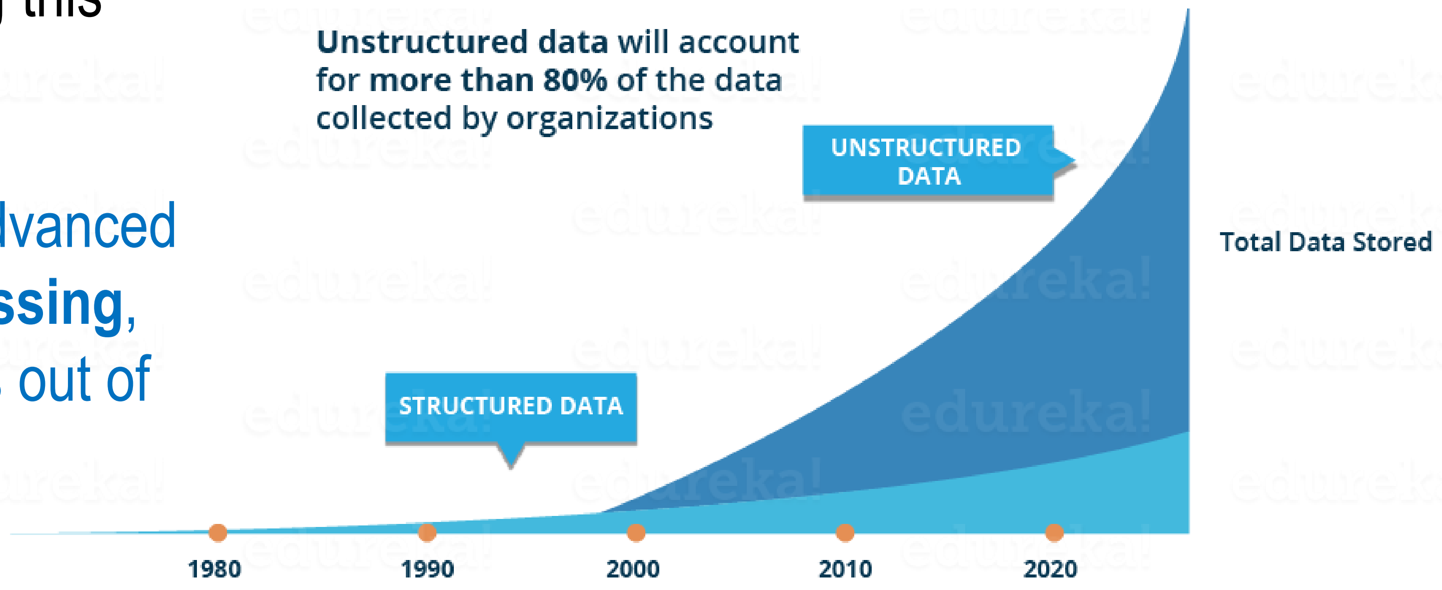
Acknowledgement to all authors whose materials have been used

The Data Processing Problem

- As the world entered the era of big data, the need for its storage also grew.
- It was the main challenge and concern for the enterprise industries until 2010.
- The main focus was on building frameworks and solutions to store data.
- Now when Hadoop and other frameworks have successfully solved the problem of storage, the focus has shifted **to the processing of this data**.

Why We Need algorithms for processing?

- Traditionally, the data was mostly structured and small in size, which could be analyzed using simple traditional tools.
- Today most of the data is unstructured or semi-structured.
- By 2025, more than 85% of the data will be unstructured.
- Simple tools are not capable of processing this huge volume and variety of data
- This is why we need more complex and advanced analytical tools and **algorithms for processing**, analyzing and drawing meaningful insights out of it



Machine Learning

- The term Machine Learning was coined by Arthur Samuel in 1959:

"Machine Learning algorithms enable the computers to learn from data, and even improve themselves, without being explicitly programmed".

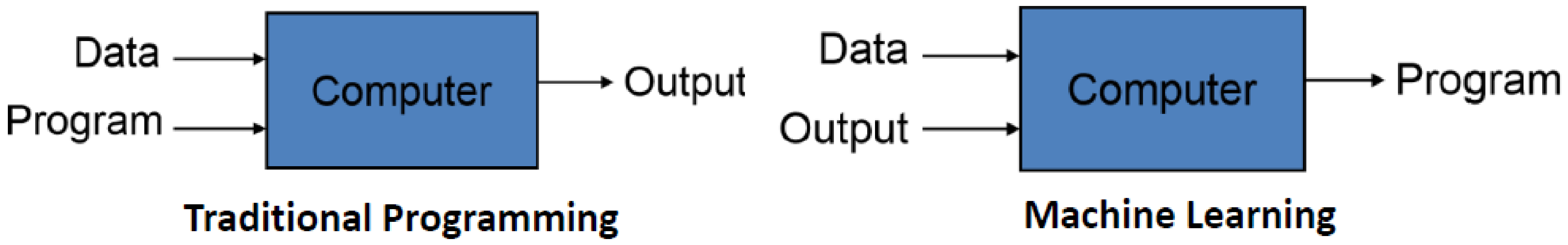
- In 1997, Tom Mitchell gave a mathematical and relational definition that:

"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance (as measured by P) on T improves with experience E".

Machine Learning Overview

What is Machine Learning?

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

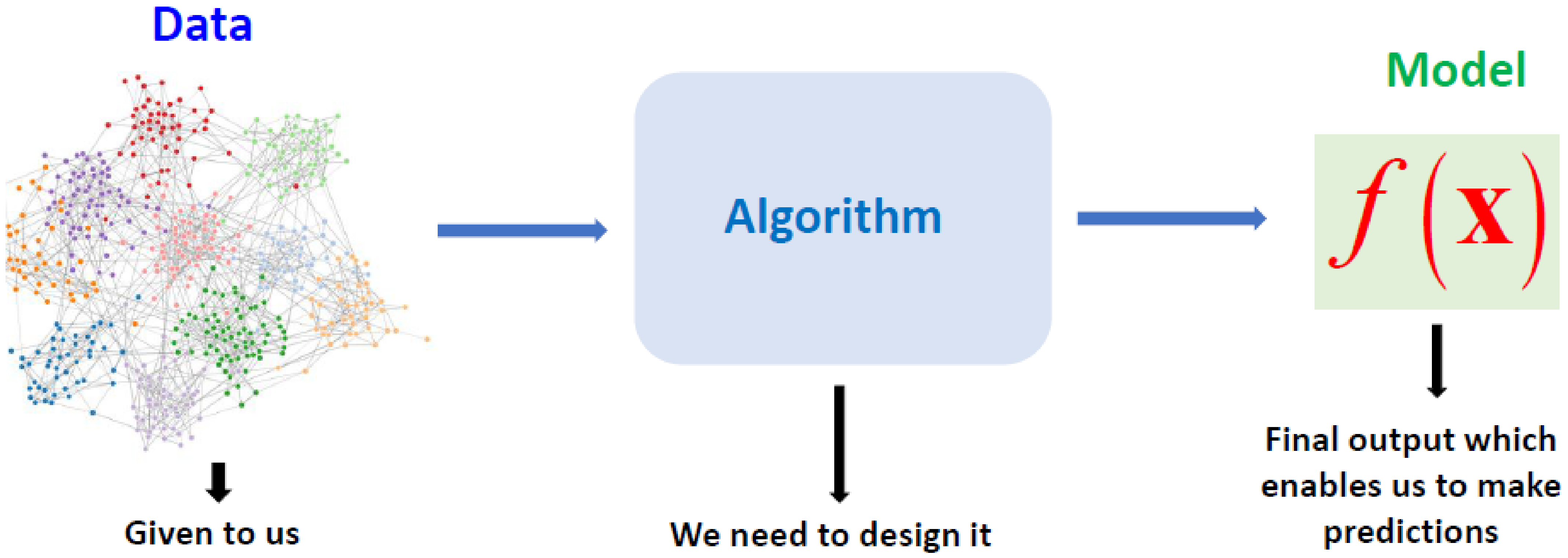


Given examples (training data), make a machine learn system behavior or discover patterns

Machine Learning: Overview

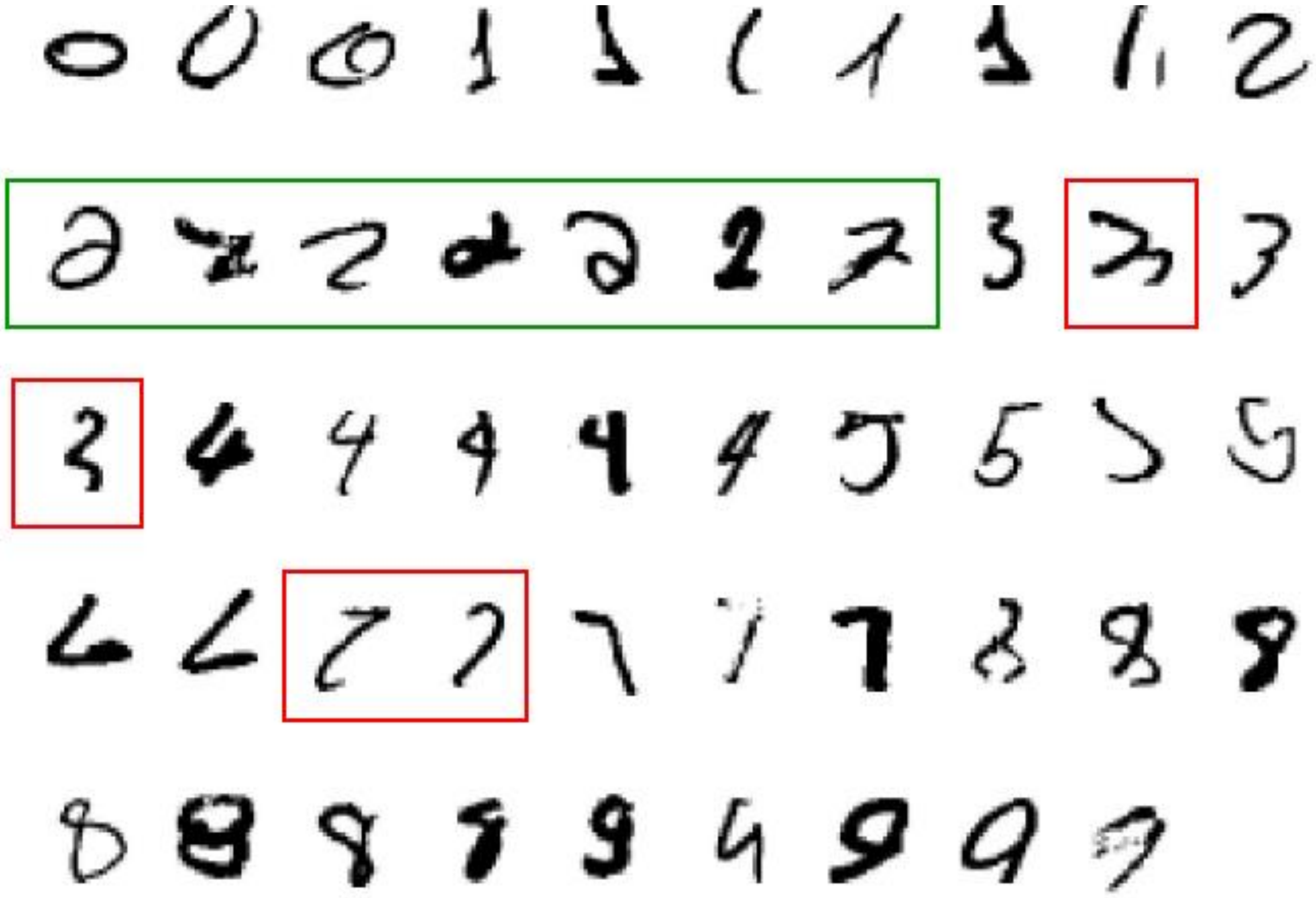
What is Machine Learning?

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

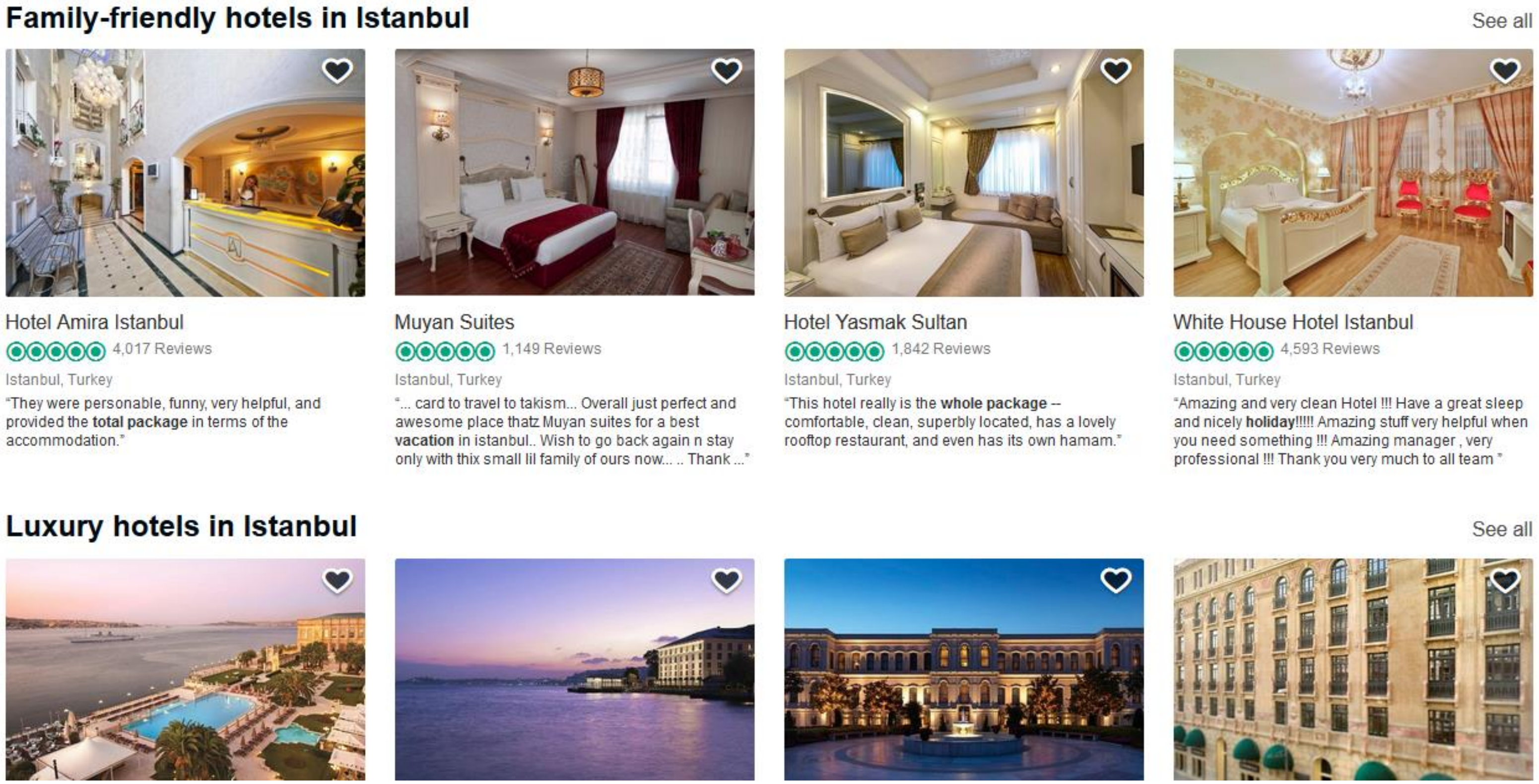


Machine Learning Overview

Classical Example: Recognize hand-written 2!



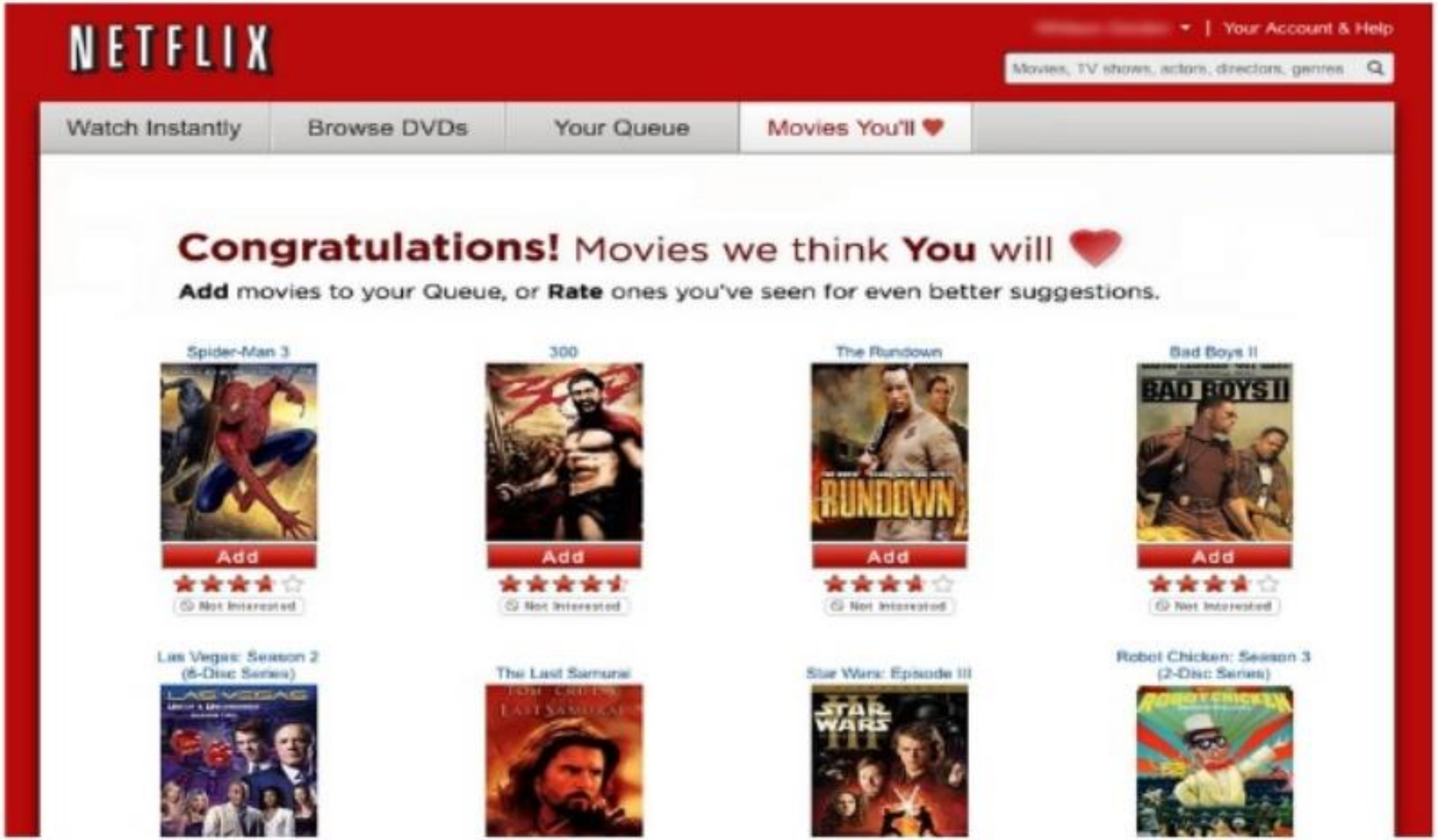
ML Examples



Example 1:

- Suppose you decide to check out trip offers for a vacation
- You browse through the travel agency website and search for a hotel
- When you look at a specific hotel, just below the hotel description there is a section titled "You might also like these hotels".
- This is a common use case of Machine Learning called "Recommendation Engine"
  - Many data points were used to train a model in order to predict what will be the best hotels to show you under that section, based on a lot of information they already know about you

Example: Netflix



ML Examples

Example 3:

- Program to predict traffic patterns at a busy intersection (task T)
- Run it through a machine learning algorithm with data (task T) about past traffic patterns (experience E) and, if it has successfully “learned”, it will then do better in predicting future traffic patterns (performance measure P).

ML Examples

Example 4: Learn to detect SPAM

- T: Distinguish between SPAM and Non-SPAM
- P: % of emails correctly classified
- E: Labeled emails from your friend Abdullah

Machine learning (ML)

- Machine learning studies the design and development of algorithms that learn from the data and improve their performance through experience.
- ML refers to a set of methods and that help computers to learn, optimize and adapt on their own.
- ML has been employed to devise algorithms for diverse applications including:
  - object detection or identification in computer vision,
  - sentiment analysis of speaker or writer,
  - detection of disease and planning of therapy in healthcare,
  - product recommendation in e-commerce,
  - learning strategies for playing games,
  - fraudulent transaction detection or loan application approval in banking sector

Applications of machine learning

- Medical diagnoses: ML is trained to recognize cancerous tissues
  - "Is this cancer?"
- Graph Processing:
  - "Which of these people are good friends with each other?"
- Recommender Systems:
  - "Will person X likes movie Y?" recommending movies to customers
- Financial industry and trading —fraud investigations and loan sanction
- Speech Recognition:
  - "Is this his/her voice?" (voice searches, voice dialing, call routing, and appliance control)

Such problems are excellent targets for Machine Learning, and in fact machine learning has been applied to such problems with great success.

Data Science – A Definition

Data Science is the science which uses computer science, statistics and machine learning, visualization and human-computer interactions to collect, clean, integrate, analyze, visualize, interact with data to create data products.

Data Science is a blend of various tools, algorithms, and machine learning principles with the goal to discover hidden patterns from the raw data.

ML 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

Types of Data we have

```xml
<note>
  <date>2017-11-08</date>
  <hour>08:30</hour>
  <to>Raj</to>
  <from>Ravi</from>
  <body>Meeting at 8am.</body>
</note>
```

- Relational Data (Tables/Transaction/Legacy Data)
- Text Data (Web)
- Semi-structured Data (XML)
- Graph Data
- Social Network, Semantic Web (RDF), …
- Streaming Data

You can afford to scan the data once

From where the data comes from?

- This data is generated from different sources like:
  - text files, multimedia forms, sensors, and instruments

- Lots of data is being collected and warehoused
  - Web data, e-commerce
  - Financial transactions logs, bank/credit transactions
  - Online trading and purchasing
  - Social Network

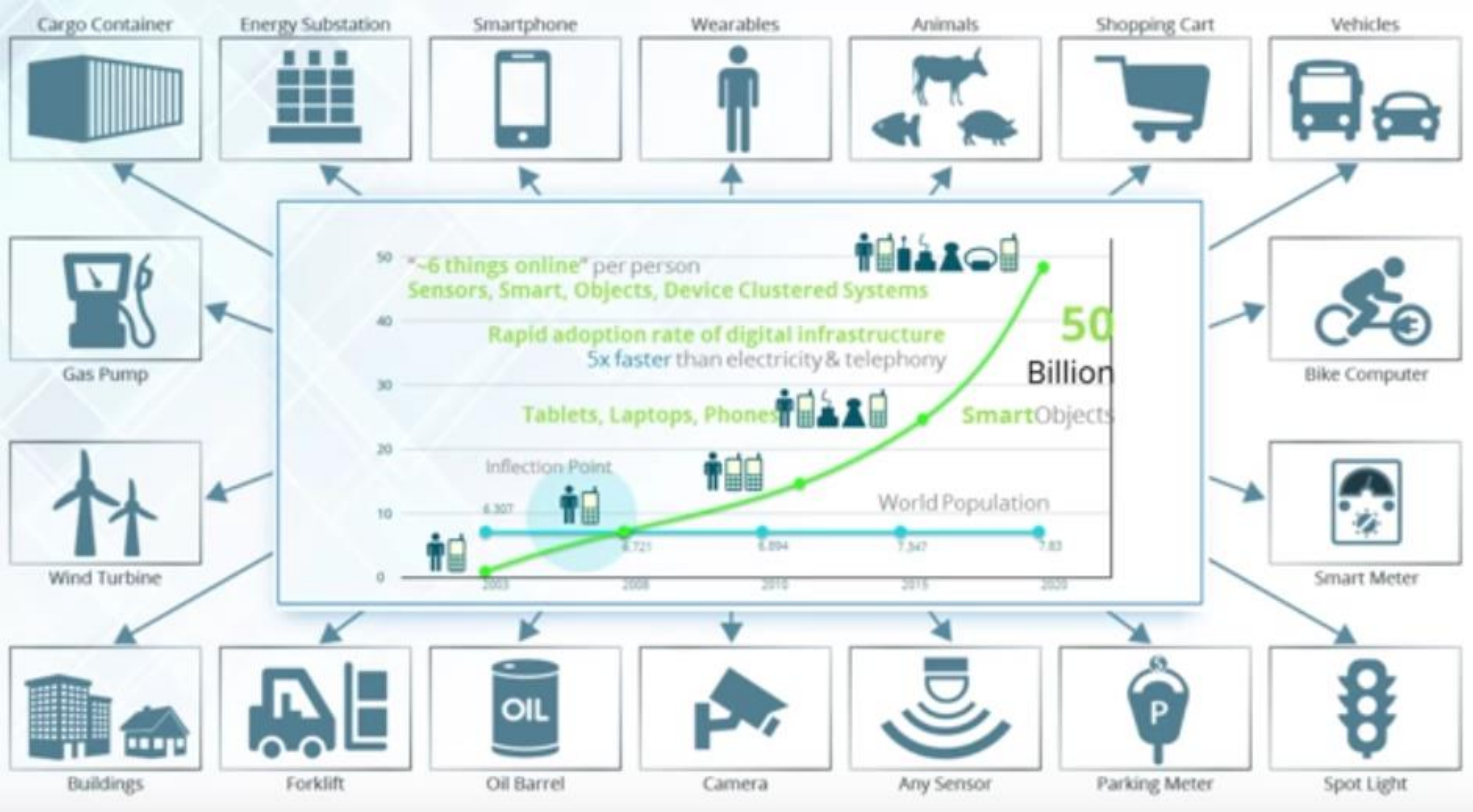
Contributors: Social Networks

Data Generated Every Minute!

Peta-bytes are in norm

- Google processes 24 PB a day (2009)
- AT&T transfers about 30 PB a day through its networks
- Microsoft migrated 150 PB of user data from Hotmail to Outlook (2013)
- Facebook stores about 357 PB of user uploaded images (2013)
- eBay has 6.5 PB of user data + 50 TB/day (2009)

Contributors: IoTs - 50 Billion Connected Devices by 2020



Contributors: Surveillance guys

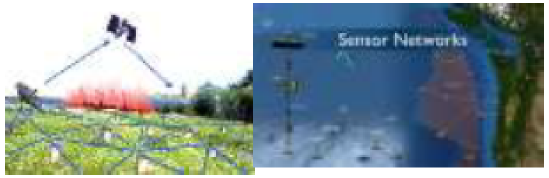
1 VGA resolution color camera produces 800 GB/hour

Contributors: Scientific Instruments

Social media and networks
(all of us are generating data)

Scientific instruments
(collecting all sorts of data)

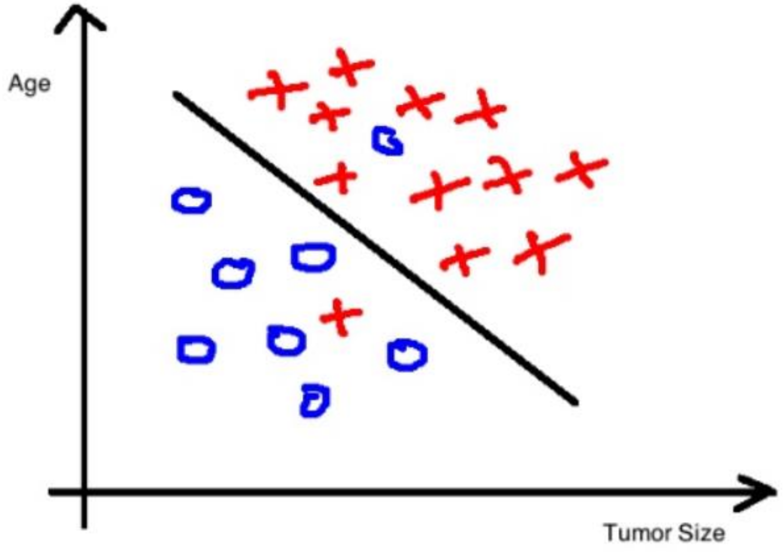
Mobile devices
(tracking all objects all the time)



Sensor technology and networks
(measuring all kinds of data)

- The progress and innovation is no longer hindered by the ability to collect data
- But, by the ability to manage, analyze, summarize, visualize, and discover knowledge from the collected data in a timely manner and in a scalable fashion

What is a Model



- A **model** is a mathematical formula with a number of parameters that need to be learned from the data
  - Fitting a model to the data is a process known as model training

- Example: Consider a one feature/variable linear regression, where the goal is to fit a line (described by the equation y = ax + b) to a set of distributed data points.

- Once the model training is completed we get a model equation y = 2x + 5.
  - Then for a set of inputs [1, 0, 7, 2, …] we would get a set of outputs [7, 5, 19, 9, …].

Machine Learning: Overview

Algorithms vs Model

- The linear regression algorithm produces a model, that is, a vector of values of the coefficients of the model (e.g. y = 2x + 5).
- The decision tree algorithm produces a model comprised of a tree of if-then statements with specific values.
- A neural network along with backpropagation + gradient descent: produces a model comprised of a trained (weights assigned) neural network.

Supervised Learning

- A supervised learning algorithm takes a known set of input dataset and its known responses to the data (output) to learn the model.
- A learning algorithm then trains a model to generate a prediction for the response to new data or the test dataset.
- Supervised learning uses classification algorithms and regression techniques to develop predictive models.
- The algorithms include linear regression, logistic regression, neural networks, decision tree, Support Vector Machine (SVM), random forest, naive Bayes, and k-nearest neighbor.

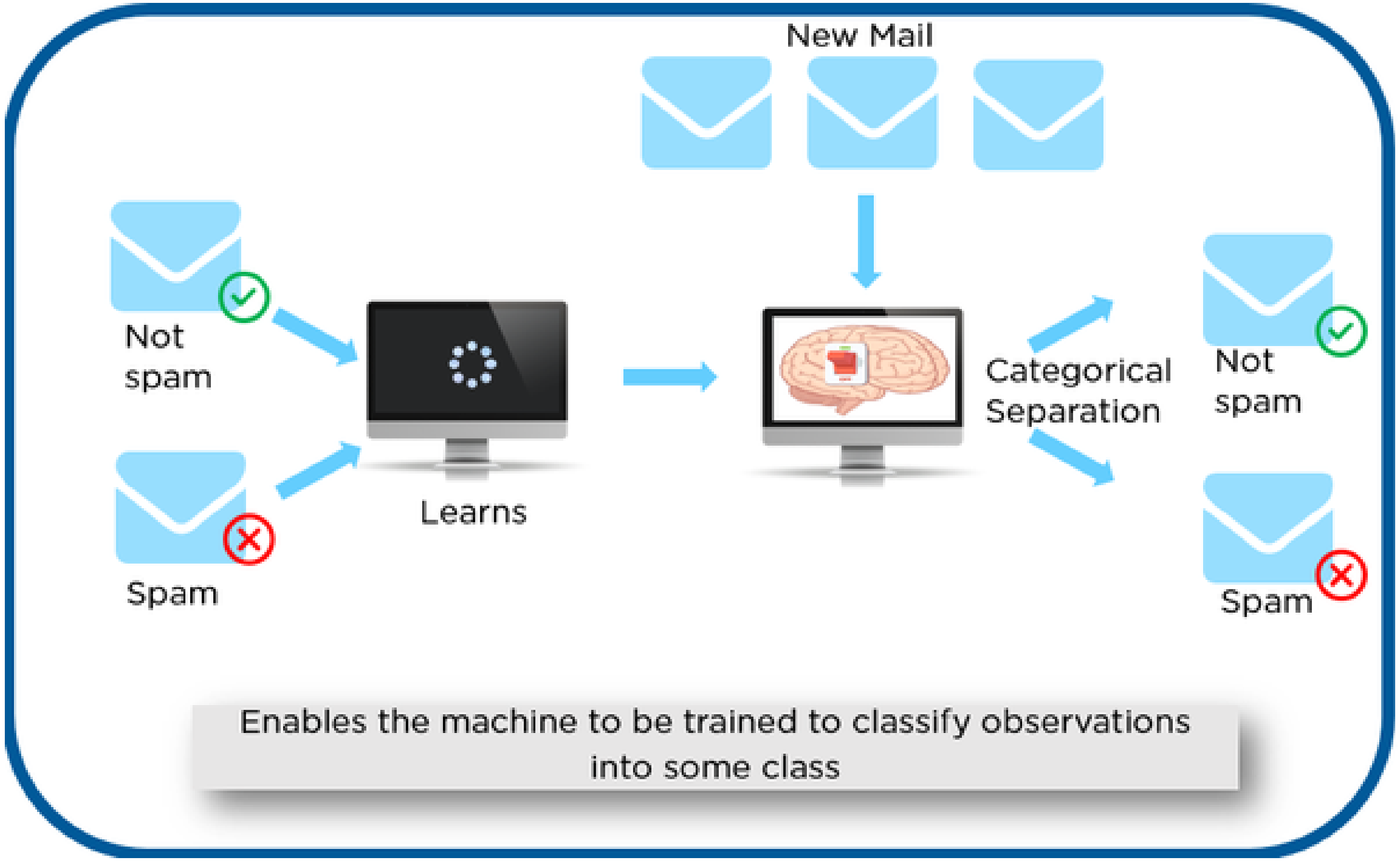
Supervised Learning

For example:

Spam filtering where large number of email messages are labelled as either:

- spam
- non-spam

- New email message will then be classified as spam or non-spam



Supervised Learning: learn from examples

Target function

Feature Matrix

Patient Age	Tumor Size	Clump Thickness	…	Malignant?
55	5	3		TRUE
70	4	7		TRUE
85	4	6		FALSE
35	2	1		FALSE
…	…	…	…	FALSE
				TRUE

Labeled dataset

Cancer model
F(k1, k2, k3, k4)

Feature Vector

f(V1, V2, V3, …) = ?

Patient age	**Tumor size**	**Clump**	**…**
72	3	3	
66	4	4	

Malignant
?
?

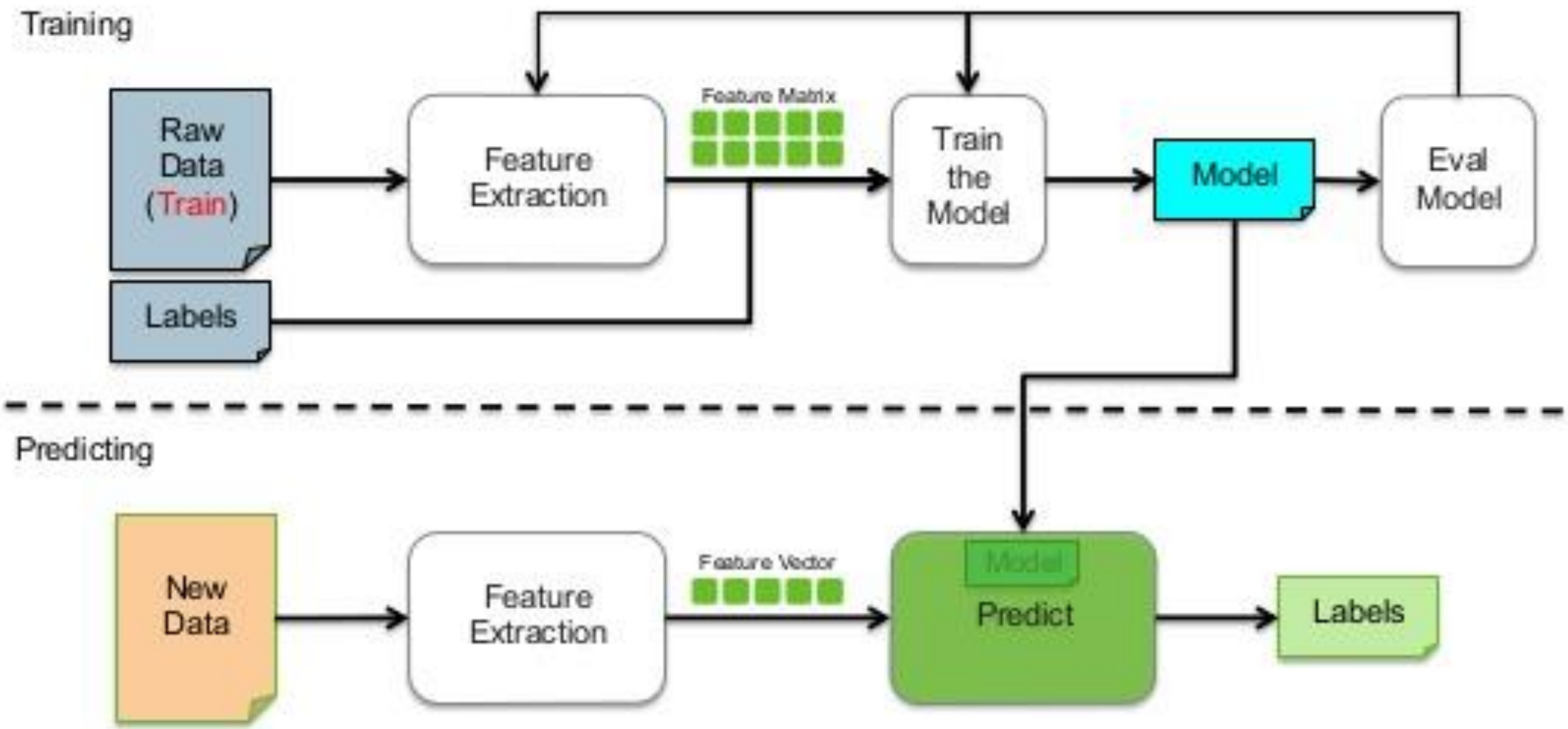
Test data

Supervised Learning

Training Images

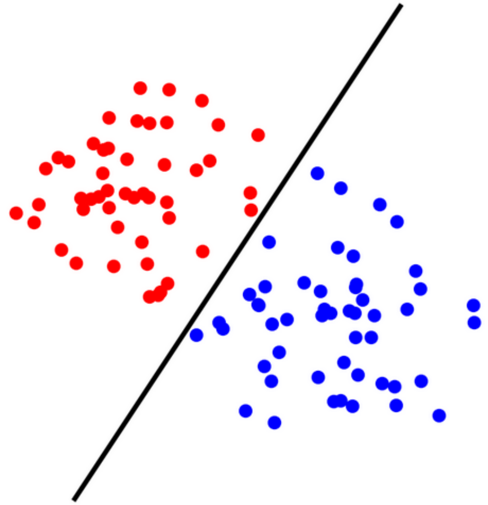
Test Image

Supervised Learning Workflow



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: To check whether an email is genuine or spam
  - Object or face detection in a photo
  - We will be back on Campus on Aug 22
  - 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***

Classification

Classification learns from existing categorizations and then assigns unclassified items to the best category.

- Classification models classify input data into categories and predict discrete responses
- Classification is recommended if the data can be categorized, tagged, or separated into specific groups or classes

- **Classification Examples:**
  - To recognize letters and numbers in Handwriting
  - To detect whether a tumor is benign or cancerous
- Classification Algorithms:
  - k-nn, Decision Trees, Random Forest, SVM, Neural Network…

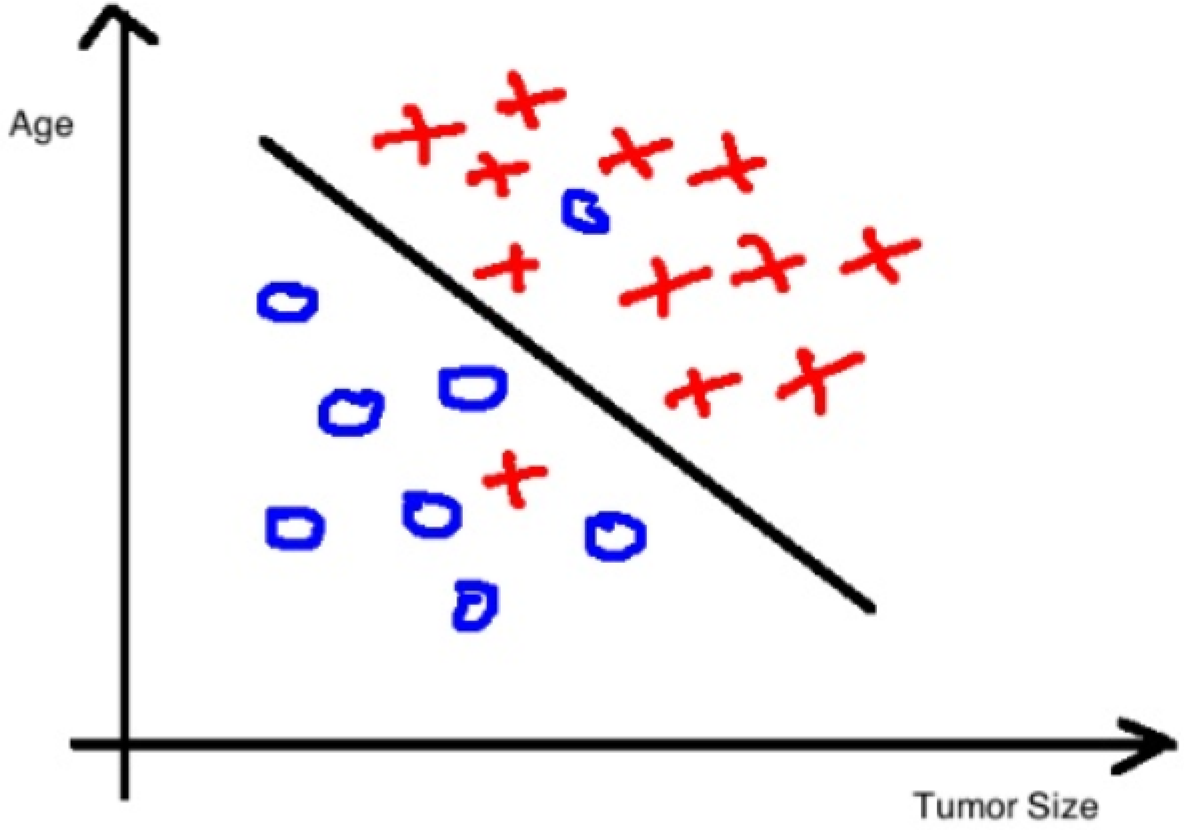
Classification Algorithms

- Classification algorithms attempt to estimate the mapping function (f) from the input variables (x) to discrete or categorical output variables (y).
  - In this case, y is a category that the mapping function predicts.

- **For example**, when provided with a dataset about houses, a classification algorithm predict whether the prices for the houses “sell more or less than the recommended retail price”

- **For example**, in a banking application, the customer who applies for a loan may be classified as a safe and risky according to his/her age and salary. The constructed model can be used to classify new data

Classification: predicting a category



Some techniques:

- Naïve Bayes
- Decision Tree
- Logistic Regression
- SGD
- Support Vector Machines
- Neural Network
- Ensembles

Basics: Regression Algorithms

Regression techniques predict continuous responses

Regression techniques predict a continuous-valued attribute associated with an object

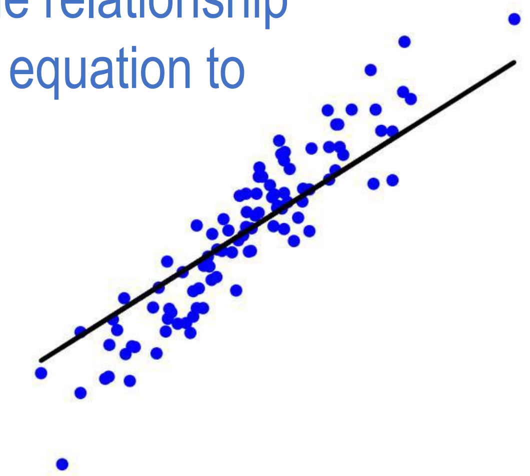
- Regression algorithms attempt to estimate the mapping function (f) from the input variables (x) to numerical or continuous output variables (y).
  - In this case, y is a real value, which can be an integer or a floating point value.
  - Therefore, regression prediction problems are usually quantities or sizes.
- For example, when provided with a dataset about houses, and you are asked to predict their prices, that is a regression task because price will be a continuous output.
- Regression algorithms include linear regression, Ensembles, Support Vector Regression (SVR), and regression trees.

Supervised Learning

Regression

Regression: Quantitative Prediction on a continuous scale

A linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data



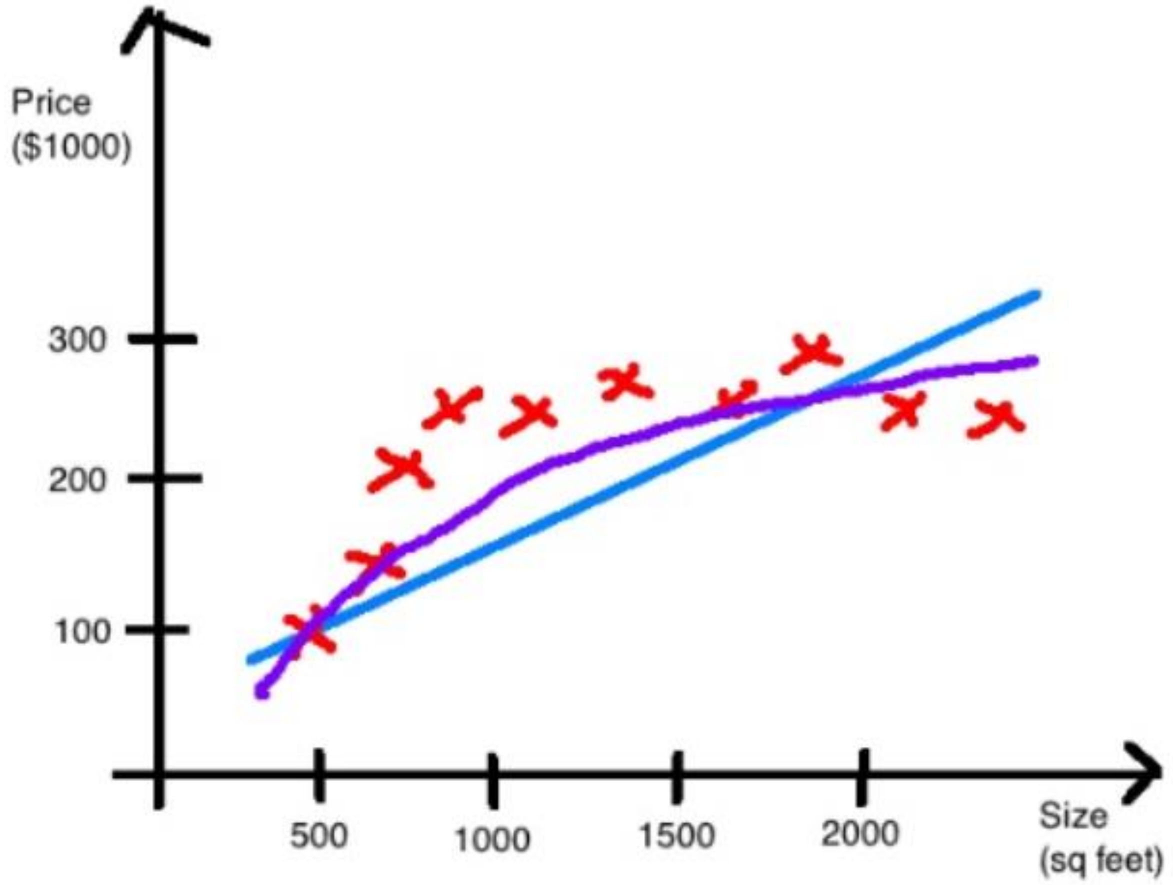
Examples: Prediction of

- 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 Pfizer Covid vaccine
- Average temperature/Rainfall during the monsoon
- Cumulative score in ML course
- Probability of a 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

Regression: predict a continuous value



Some techniques:

- Linear Regression / GLM
- Decision Trees
- Support vector regression
- SGD
- Ensembles

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 trained 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

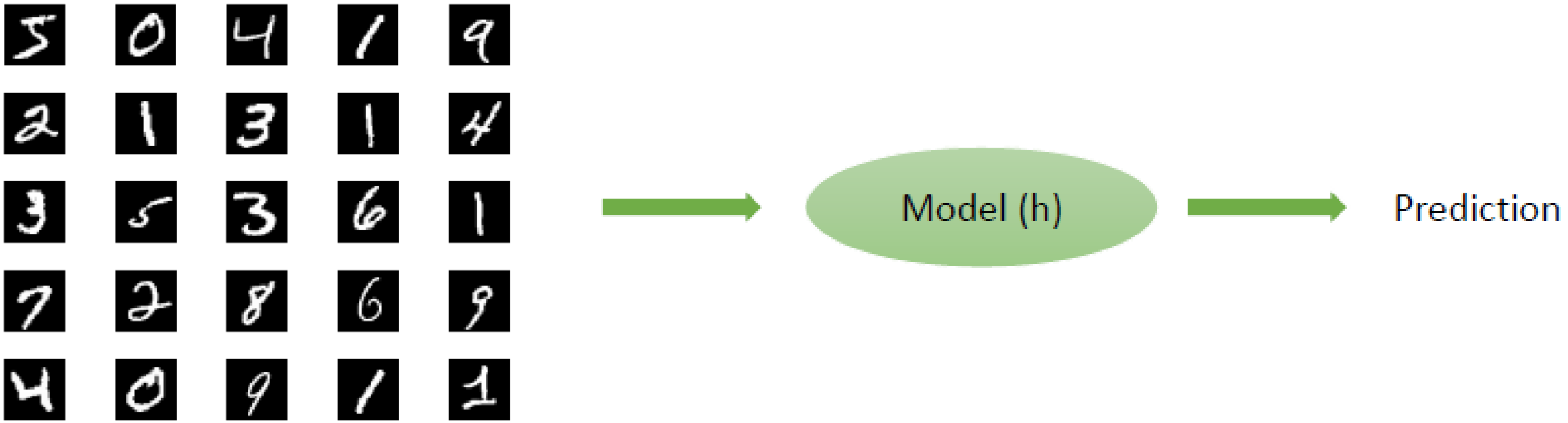
Example

Data of 200 Patients:

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

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 $\boxed{\text{(unknown)}}$ 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:

- **Normalize the loss by the total number of training samples, n, so that the output can be interpreted as the average loss per sample.**
- **Loss function counts the number of mistakes made by hypothesis function 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**)

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

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:

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

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

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:

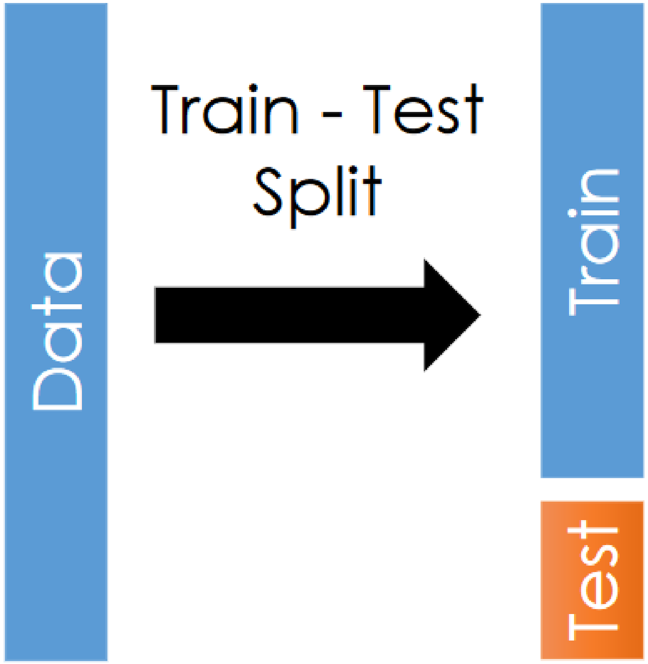
- *0% loss error on the training data (Model is fit to every data point in D).*
- *Large error for some input not in D*
- *First glimpse of **overfitting.***

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

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

Evaluation

Loss on the testing data is given by

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

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

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

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).

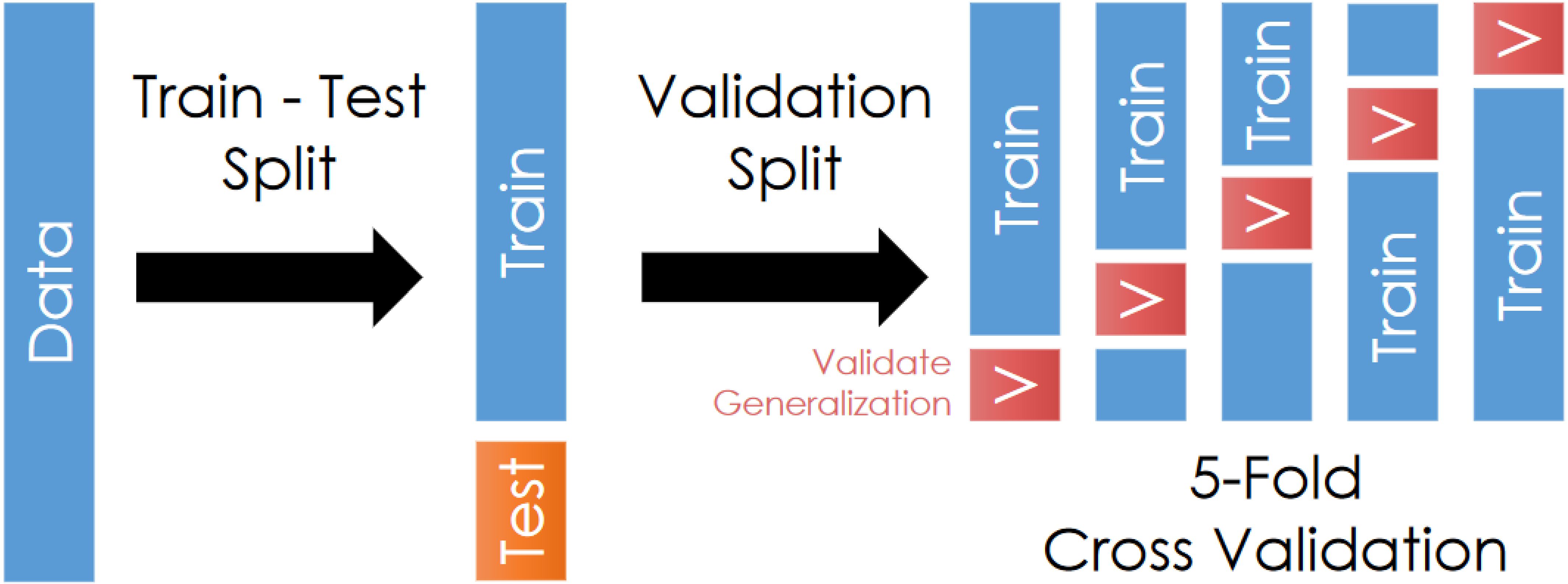
- **Validation set:** A set of examples used to tune the hyper-parameters of the hypothesis function, for example to choose the number of hidden units in a neural network OR the order of polynomial approximating the data.

- **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

Generalization: The Train-Test Split (Example)



Cross validation simulates multiple train-test splits on the training data