Introduction to Machine Learning

Outline

- Big Data and data mining
- A brief introduction to Machine Learning
- Mathematical Preliminary

Big Data

- Widespread use of personal computers and wireless communication leads to “big data”.
- We are both producers and consumers of data.
- Data is not random, it has structure, e.g., customer behavior.
- We need “big theory” to extract that structure from data for

  (a) Understanding the process.

  (b) Making predictions for the future .

Why “Learn” ?

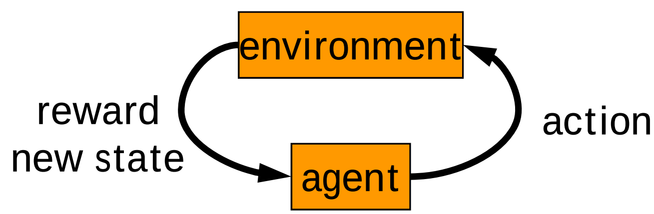
- What is learning?
  - “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.” --Tom Mitchell, CMU
- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- There is no need to “learn” to calculate payroll.
- Learning is used when:
  - Human expertise does not exist (navigating on Mars).
  - Humans are unable to explain their expertise (speech recognition).
  - Solution changes in time (routing on a computer network).
  - Solution needs to be adapted to particular cases (user biometrics).

What is Machine Learning?

- Optimize a performance criterion using example data or past experience.
- Relations to Statistics
  - Stats is more concerned with helping scientists and policymakers draw good conclusions; ML is more concerned with building autonomous agents.
  - Stats puts more emphasis on interpretability and mathematical rigor; ML puts more emphasis on predictive performance, scalability, and autonomy.
- Relations to AI
  - AI does not always imply a learning based system: Symbolic reasoning, Rule based systems, Tree search, etc.
  - Learning based system → learned based on the data → more flexibility, good at solving pattern recognition problems.

Three Types

- Supervised learning
  - Classification, regression. Have labeled examples of the correct behavior.
- Unsupervised learning
  - Clustering. no labeled examples – instead, looking for “interesting” patterns in the data.
- Reinforcement learning
  - More general than supervised/unsupervised learning.
  - Learn from interaction w/ environment to maximize a scalar reward signal.

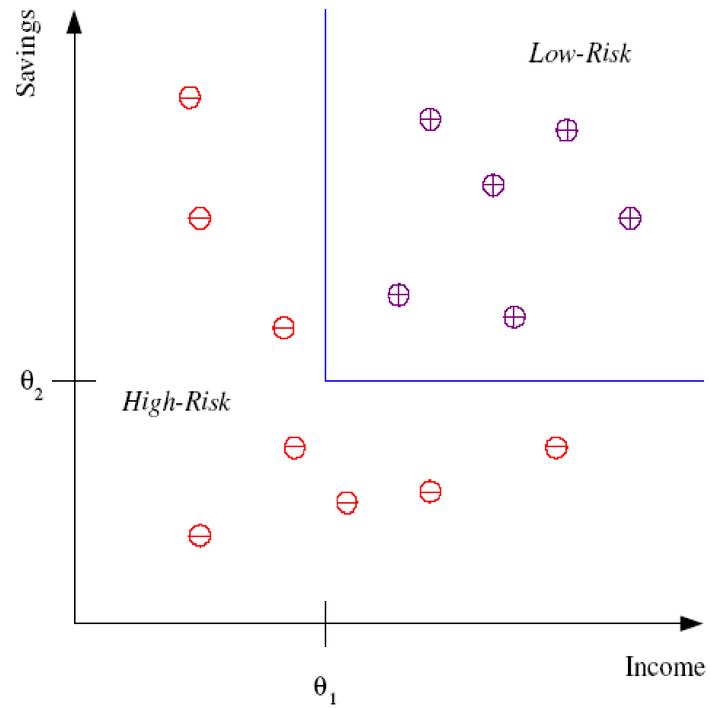


Supervised Learning

- Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. [Wikipedia]
- Prediction of future cases: Use the rule to predict the output for future inputs.
- Knowledge extraction: The rule is easy to understand.
- Model Compression: The rule is simpler than the data it explains.
- Outlier detection: Exceptions that are not covered by the rule, e.g., fraud.

Multiple outputs - Classification

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*



Discriminant: IF *income* > θ_1 AND *savings* > θ_2
THEN low-risk ELSE high-risk

Classification: Applications

- Aka Pattern recognition
- Face recognition: Pose, lighting, occlusion (glasses, beard), make-up, hair style.
- Character recognition: Different handwriting styles.
- Speech recognition: Temporal dependency.
- Medical diagnosis: From symptoms to illnesses.
- Biometrics: Recognition/authentication using physical and/or behavioral characteristics: face, fingerprint, signature, etc.

Face Recognition

Training examples of a person

Test images

Single output - Regression

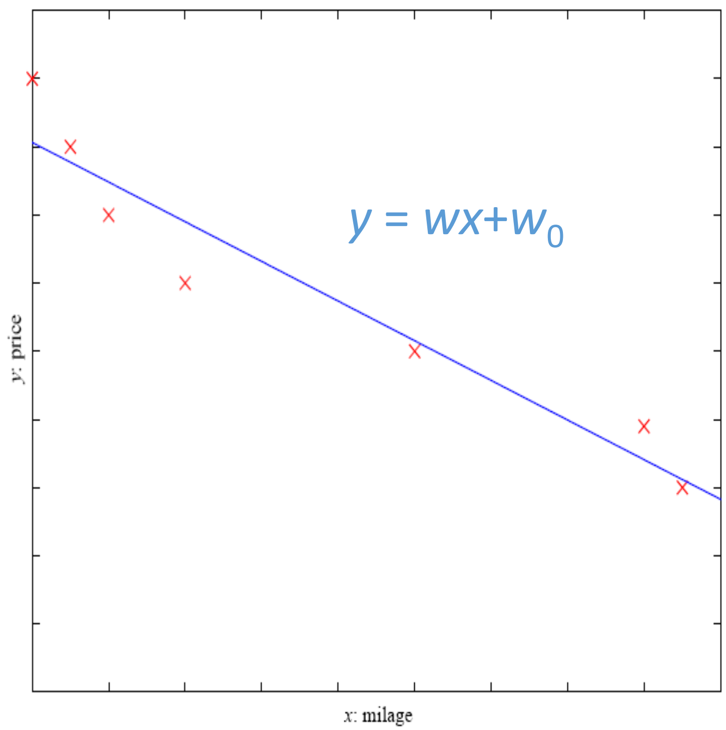
- Example: Price of a used car
- x : car attributes

  y : price

  $$y = g(x \mid \theta)$$

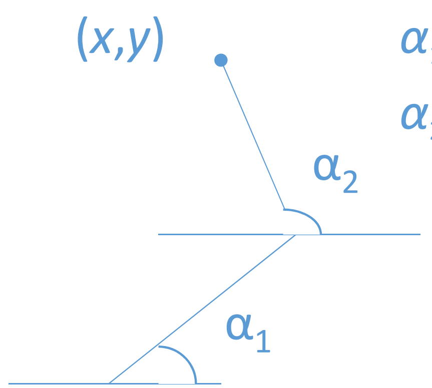
  $g(\ )$ model,

  θ parameters

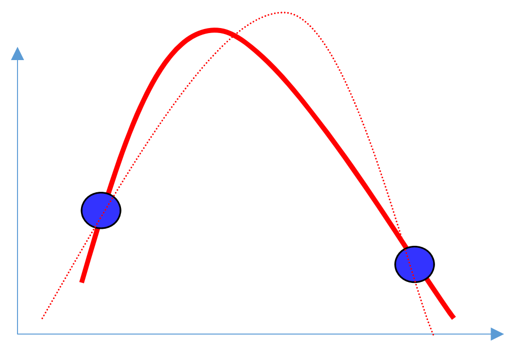


Regression Applications

- Navigating a car: Angle of the steering
- Kinematics of a robot arm



$$\alpha_1 = g_1(x,y)$$
$$\alpha_2 = g_2(x,y)$$



Unsupervised Learning

- Learning “what normally happens”.
- Clustering: Grouping similar instances.
- Example applications
  - Customer segmentation in Customer Relationship Management.
  - Image compression: Color quantization.
  - Bioinformatics: Learning motifs (orders of A, C, G, T).

Reinforcement Learning

- Learning a policy: A sequence of [state, action] transactions.
- No supervised output but delayed reward (i.e. reinforcement).
- Applications
  - Credit assignment problem.
  - Game playing.
  - Robot in a maze.
  - Multiple agents, partial observability, ...

Resources: Dataset

- UCI Machine Learning Repository: https://archive.ics.uci.edu/datasets
- Kaggle Datasets: https://www.kaggle.com/datasets

Resources: Journals

- Journal of Machine Learning Research www.jmlr.org
- Machine Learning
- Neural Computation
- Neural Networks
- IEEE Trans on Neural Networks and Learning Systems
- IEEE Trans on Pattern Analysis and Machine Intelligence
- Journals on Statistics/Data Mining/Signal Processing/Natural Language Processing/Bioinformatics/...

Resources: Conferences

- International Conference on Machine Learning (ICML)
- European Conference on Machine Learning (ECML)
- Neural Information Processing Systems (NIPS)
- Uncertainty in Artificial Intelligence (UAI)
- Computational Learning Theory (COLT)
- International Conference on Artificial Neural Networks (ICANN)
- International Conference on AI & Statistics (AISTATS)
- International Conference on Pattern Recognition (ICPR)
- ...

Math Preliminary

- Random Variable
- Expected Value
- Sampling

Random Variable

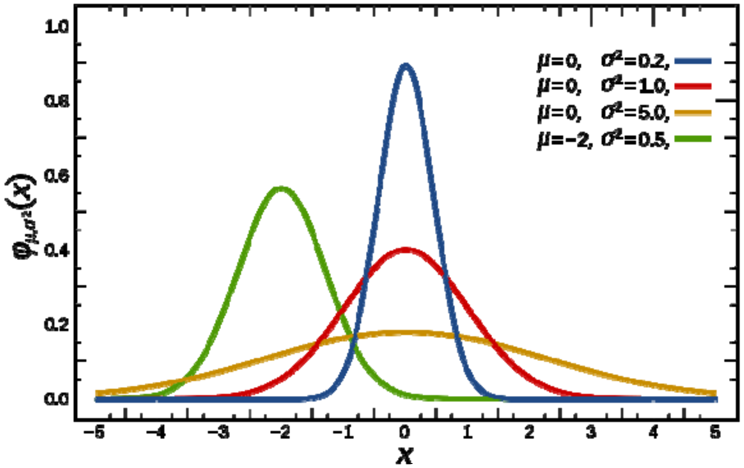
- Random Variable: a variable whose values depend on outcomes of a random event.
- Uppercase letter X for random variable.

Random event	Random Variable	Possible values	Probabilities
	X	0 1	$P(X = 0) = 0.5$ $P(X = 1) = 0.5$

- Lowercase letter x for an observed value. For example, I tossed a coin 4 times and observed: $x_1 = 0$, $x_2 = 1$, $x_3 = 0$, $x_4 = 0$

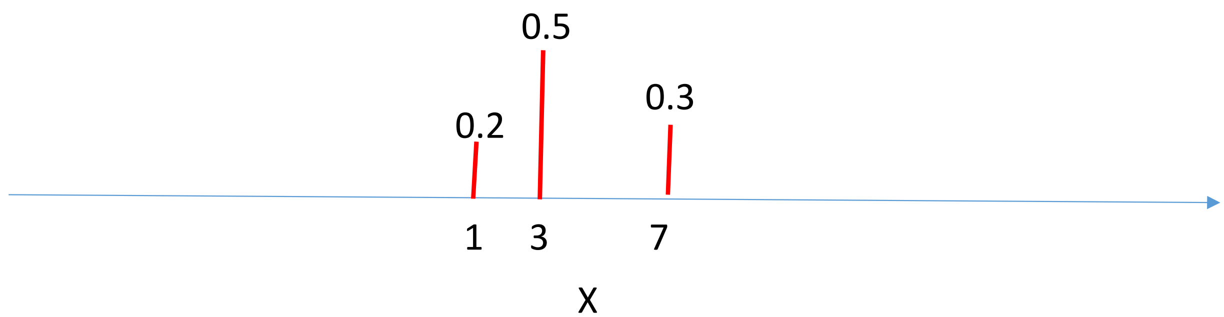
Probability Density Function (PDF)

- PDF provides a relative likelihood that the value of the random variable would equal that sample.
- Example: Gaussian distribution
  - It is a continuous distribution.
  - PDF = $\frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(x-\mu)^2}{2\sigma^2})$, where σ is the standard deviation and μ is the mean.



Probability Density Function (Cont'd)

- Example: Discrete distribution
  - Discrete random variable: $X \in \{1, 3, 7\}$
  - PDF: $p(1) = 0.2$
    $p(3) = 0.5$
    $p(7) = 0.3$



Probability Density Function (Cont'd)

- Random variable X is in the domain $\mathcal{X}$.
- For continuous distribution, $\int_{\mathcal{X}} p(x)dx = 1$.
- For discrete distribution, $\sum_{x \in \mathcal{X}} p(x) = 1$.

Expectation

- Random variable X is in the domain $\mathcal{X}$.
- For continuous distribution, the expectation of f(x) is:

  E[f(X)] = $\int_{\mathcal{X}} p(x) f(x) dx$

- For discrete distribution, the expectation of f(x) is:

  E[f(X)] = $\sum_{x \in \mathcal{X}} p(x) f(x)$.

- For example, if our random variable were the number obtained by rolling a fair 3-sided die, the expected value would be (1 * 1/3) + (2 * 1/3) + (3 * 1/3) = 2.

Expectation (cont'd)

- Neutral Expected Value Games
  - You flip the fair coin. Every time you get heads, you lose $1, and every time you get tails, you gain $1.
  - The expected value is $(-1 * 1/2) + (1 * 1/2) = 0$.
  - Since the coin is fair and the loss amount equals the gain amount, you are expected to neither gain nor lose money over time. In such a game, while there is no reason to play, there is also no reason not to play.
  - These types of games are therefore ideal for simple recreation, such as with rock-paper-scissors, in which randomly choosing a move is the optimal strategy with an expected gain of 0.

Expectation (cont'd)

- Positive Expected Value Games
  - You flip the fair coin. Every time you get heads, you lose $1, and every time you get tails, you gain $2.
  - The expected value is $(-1 * 1/2) + (2 * 1/2) = 1/2$.
  - Since heads and tails are equally-likely, the larger gain for tails outweighs the loss for heads. In such a game you are expected to gain money over time, so you should play this type of game.
  - These type of scenarios appear in many real-life decisions, such as investing in the stock market (the markets are in a general uptrend over time), studying for an exam (the few hours of lost time are outweighed by a higher GPA), or preparing for an interview (a few weeks of lost time are outweighed by the benefits from having a better job).

Expectation (cont'd)

- Negative Expected Value Games
  - You flip the fair coin. Every time you get heads, you lose $1, and every time you get tails, you gain $1. Additionally, there is a $0.01 fee for every flip regardless of the outcome.
  - The expected value is (-1.01 * 1/2) + (.99 * 1/2) = -0.01.
  - Despite the coin itself being fair and the loss amount equaling the gain amount, the constant fee causes the game to be a negative-valued game. In such a game, you are expected to lose money over time, so you should not play this type of game.
  - This is common in many gambling platforms, in which the house provides an initially-neutral game, but then charges a fee that ruins the neutrality of the game (hence the saying that "the house always wins").

Perfect Rationality vs. Human Rationality

- Famous example of Allais (1953)
  - A: [0.8, $4k;   0.2, $0]
  - B: [1.0, $3k;   0.0, $0]

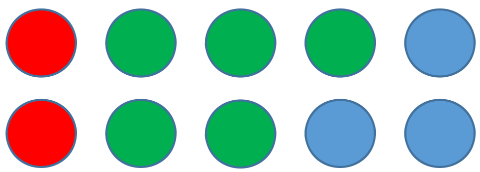
  - C: [0.2, $4k;   0.8, $0]
  - D: [0.25, $3k;   0.75, $0]

- Most people prefer B > A, C > D

- But E(A) > E(B).

Random Sampling

- There are 10 balls in a bin: 2 are red, 5 are green, and 3 are blue.
- Randomly sample a ball.
- What will be the outcome?



- What is X and what are its possible values?

Random Sampling (cont'd)

- Sample red ball with probability 0.2, green ball with probability 0.5, and blue ball with probability 0.3.
- Random sample a ball. Observe its color. Put it back to the bin. Repeat the process for 100 times.
- What will be the outcome?

```python
from numpy.random import choice

samples = choice(['R', 'G', 'B'], size=100, p=[0.2, 0.5, 0.3])
print(samples)
```

```
['R' 'G' 'R' 'R' 'R' 'R' 'B' 'B' 'B' 'G' 'G' 'B' 'G' 'B' 'B' 'G' 'B' 'G'
 'B' 'B' 'G' 'B' 'G' 'B' 'B' 'G' 'B' 'B' 'G' 'B' 'G' 'G' 'G' 'G' 'G' 'B'
 'B' 'B' 'B' 'B' 'B' 'G' 'G' 'B' 'R' 'R' 'B' 'R' 'B' 'G' 'R' 'G' 'R' 'G'
 'R' 'R' 'B' 'G' 'G' 'G' 'B' 'R' 'G' 'B' 'G' 'R' 'G' 'G' 'G' 'B' 'B' 'R'
 'G' 'G' 'B' 'B' 'R' 'B' 'B' 'B' 'R' 'B' 'G' 'B' 'R' 'B' 'R' 'G' 'B' 'R'
 'B' 'B' 'G' 'G' 'G' 'R' 'R' 'B' 'R' 'G']
```