Course Overview and Introduction
CE-717 : Machine Learning
Sharif University of Technology

M. Soleymani
Fall 2020
Course Info

Instructor: Mahdieh Soleymani
   Email: soleymani@sharif.edu
   Office hours: set appointment through email

Head TA: Fatemeh Seyyedsalehi (fateme.ssalehi@gmail.com)

Website: http://ce.sharif.edu/courses/99-00/1/ce717-1/
   Tentative schedule
   Slides and notes
   Policies and rules

Discussions: On Piazza

HWs: On Quera
Prerequisites:

- Programming skills
  - Python
- Probability and statistics
- Basic linear algebra
  - We'll go over it in the review sections.
Marking Scheme

<table>
<thead>
<tr>
<th>Component</th>
<th>Weightage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm Exam</td>
<td>20%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>25%</td>
</tr>
<tr>
<td>Homeworks (written &amp; programming)</td>
<td>40%</td>
</tr>
<tr>
<td>Mini-exams</td>
<td>15%</td>
</tr>
</tbody>
</table>
Assignments

7 Problem sets

The first one is on prerequisites.
Other sets contain both theoretical and programming assignments
Text Books


Other books:

A Definition of ML

Tom Mitchell (1998):
A computer program is said to learn from experience if its performance improves with experience

Using the observed data to make better decisions
Generalizing from the observed data
ML Definition: Example

Consider an email program that learns how to filter spam according to emails you do or do not mark as spam.

Task: Classifying emails as spam or not spam.
Experience: Watching you label emails as spam or not spam.
Performance: The number (or fraction) of emails correctly classified as spam/not spam.
The essence of machine learning

A pattern exist

We do not know it mathematically

We have data on it
Example: Home Price

Housing price prediction

Price ($) in 1000’s vs. Size in feet²
Example: Home Price

Predicting house price from 3 attributes

<table>
<thead>
<tr>
<th>Size ($m^2$)</th>
<th>Age (year)</th>
<th>Region</th>
<th>Price ($10^6T$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>2</td>
<td>5</td>
<td>500</td>
</tr>
<tr>
<td>80</td>
<td>25</td>
<td>3</td>
<td>250</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Example: Home Price

Input: X
Size
Age
Region

Model

Output: Y
price
Example: Bank loan

Applicant form as the input:

- salary
- age
- gender
- current debt
- ...

Output: approving or denying the request
Experience (E) in ML

Basic premise of learning:

“Using a set of observations to uncover an underlying process”

We have different types of (getting) observations in different types or paradigms of ML methods
Paradigms of ML

**Supervised learning** (regression, classification)
predicting a target variable for which we get to see examples.

**Unsupervised learning**
revealing structure in the observed data

**Reinforcement learning**
partial (indirect) feedback, no explicit guidance
Given rewards for a sequence of moves to learn a policy and utility functions
Data in Supervised Learning

Data are usually considered as vectors in a $d$ dimensional space

Now, we make this assumption for illustrative purpose
We will see it is not necessary

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>...</th>
<th>$x_d$</th>
<th>$y$ (Target)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample n-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample n</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Columns:
*Features/attributes/dimensions*

Rows:
*Data/points/instances/examples/samples*

Y column:
*Target/outcome/response/label*
Supervised Learning: Regression vs. Classification

**Supervised Learning**

**Regression**: predict a *continuous* target variable
E.g., \( y \in [0,1] \)

**Classification**: predict a *discrete* (unordered) target variable
E.g., \( y \in \{1, 2, \ldots, C\} \)
Regression: Example

Housing price prediction

Price ($)
in 1000’s

Size in feet²
Classification: Example

Classification of tumors to Benign/Malignant according to attributes

Malignant +1

Benign 0

Tumor size
Classification: Example

Classification of tumors to Benign/Malignant according to attributes

Tumor size

+1

0

Tumor size

Tumor size
Training data: Example

![Diagram of training data with points and axes labeled $x_1$ and $x_2$.]

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>2.3</td>
<td>0</td>
</tr>
<tr>
<td>3.5</td>
<td>2.6</td>
<td>0</td>
</tr>
<tr>
<td>2.6</td>
<td>3.3</td>
<td>0</td>
</tr>
<tr>
<td>2.7</td>
<td>4.1</td>
<td>0</td>
</tr>
<tr>
<td>1.8</td>
<td>3.9</td>
<td>0</td>
</tr>
<tr>
<td>3.5</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4.2</td>
<td>3.7</td>
<td>1</td>
</tr>
<tr>
<td>4.9</td>
<td>4.5</td>
<td>1</td>
</tr>
<tr>
<td>3.9</td>
<td>4.5</td>
<td>1</td>
</tr>
<tr>
<td>5.8</td>
<td>4.1</td>
<td>1</td>
</tr>
<tr>
<td>6.1</td>
<td>2.6</td>
<td>1</td>
</tr>
</tbody>
</table>
Handwritten Digit Recognition Example

Data: labeled samples

0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9
Handwritten Digit Recognition Example

Input: X

Classifier Model

Output: Y

0
1
2
...
9
Components of (Supervised) Learning

Unknown target function: \( f : X \rightarrow Y \)
- Input space: \( X \)
- Output space: \( Y \)

Training data: \((x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\)

Pick a formula \( g : X \rightarrow Y \) that approximates the target function \( f \)
- selected from a set of hypotheses \( \mathcal{H} \)
Components of (Supervised) Learning

We have some example pairs of (input, output) called training samples

\[(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})\]

We want to select a function from the input space to the output space

\[f: \mathcal{X} \rightarrow \mathcal{Y}\]

We choose a set of hypotheses (candidate formulas)

e.g., linear functions

We use a learning algorithm to select a function from hypothesis set that approximates the target function
Components of (Supervised) Learning

\[ f : \mathcal{X} \mapsto \mathcal{Y} \]

\[ (x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N) \]

Learning model

\[ \mathcal{H} \]

(set of candidate formulas)

\[ g \approx f \]
(Supervised) Learning problem

Selecting a **hypothesis space**

Hypothesis space: a set of mappings from feature vector to target

**Learning:** find mapping \( \hat{f} \) (from hypothesis set) based on the training data

Which notion of error should we use? (loss functions)

Optimization of loss function to find mapping \( \hat{f} \)

**Evaluation:** we measure how well \( \hat{f} \) generalizes to unseen examples (generalization)
Solution Components

Learning model composed of:
- Hypothesis set
- Learning algorithm

Perceptron example
Handwritten Digit Recognition Example

Data: labeled samples

0
1
2
3
4
5
6
7
8
9
Example: Input representation

'raw' input $\mathbf{x} = (x_0, x_1, x_2, \cdots, x_{256})$

linear model: $(w_0, w_1, w_2, \cdots, w_{256})$

Features: Extract useful information, e.g.,

intensity and symmetry $\mathbf{x} = (x_0, x_1, x_2)$

linear model: $(w_0, w_1, w_2)$
Example: Illustration of features

\[ \mathbf{x} = (x_0, x_1, x_2) \]

- \( x_1 \): intensity
- \( x_2 \): symmetry

https://work.caltech.edu/telecourse.html
Perceptron classifier

Input $x = [x_1, ..., x_d]$

Classifier:
- If $\sum_{i=1}^{d} w_i x_i > \text{threshold}$ then output 1
- else output $-1$

The linear formula $g \in \mathcal{H}$ can be written:

$$g(x) = \text{sign} \left( \sum_{i=1}^{d} w_i x_i - \text{threshold} \right)$$
Perceptron classifier

Input $x = [x_1, ..., x_d]$

Classifier:

If $\sum_{i=1}^{d} w_i x_i > \text{threshold}$ then output 1
else output $-1$

The linear formula $g \in \mathcal{H}$ can be written:

$$g(x) = \text{sign} \left( \sum_{i=1}^{d} w_i x_i + w_0 \right)$$
Perceptron classifier

Input $x = [x_1, ..., x_d]$

Classifier:

If $\sum_{i=1}^{d} w_i x_i > \text{threshold}$ then output 1
else output $-1$

The linear formula $g \in \mathcal{H}$ can be written:

$$g(x) = \text{sign} \left( \sum_{i=1}^{d} w_i x_i + w_0 \right)$$

If we add a coordinate $x_0 = 1$ to the input:

$$g(x) = \text{sign} \left( \sum_{i=0}^{d} w_i x_i \right)$$

Vector form

$$g(x) = \text{sign}(\mathbf{w}^T \mathbf{x})$$
Perceptron learning algorithm: linearly separable data

Give the training data $(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})$

Misclassified data $(x^{(n)}, y^{(n)})$:
\[
\text{sign}(w^T x^{(n)}) \neq y^{(n)}
\]

Repeat

Pick a misclassified data $(x^{(n)}, y^{(n)})$ from training data and update $w$:
\[
w = w + y^{(n)} x^{(n)}
\]

Until all training data points are correctly classified by $g$
Perceptron learning algorithm: Example of weight update

\( y = +1 \)

\[ w + y \mathbf{x} \]

\( y = -1 \)

\[ w + y \mathbf{x} \]
Example: linear classifier

\[ \mathbf{x} = (x_0, x_1, x_2) \]

- \( x_1 \): intensity
- \( x_2 \): symmetry

https://work.caltech.edu/telecourse.html
(Supervised) Learning problem

Selecting a **hypothesis space**

- Hypothesis space: a set of mappings from feature vector to target

**Learning (estimation):** optimization of a cost function

- Based on the training set $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^n$ and a cost function we find (an estimate) $f \in F$ of the target function

**Evaluation:** we measure how well $\hat{f}$ generalizes to unseen examples
Generalization

We don’t intend to memorize data but want to distinguish the pattern.

A core objective of learning is to generalize from the experience.

Generalization: ability of a learning algorithm to perform accurately on new, unseen examples after having experienced.
Paradigms of ML

**Supervised learning** (regression, classification)
predicting a target variable for which we get to see examples.

**Unsupervised learning**
revealing structure in the observed data

**Reinforcement learning**
partial (indirect) feedback, no explicit guidance
Given rewards for a sequence of moves to learn a policy and utility functions
Supervised Learning vs. Unsupervised Learning

**Supervised learning**

Given: Training set

labeled set of $N$ input-output pairs $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$

Goal: learning a mapping from $x$ to $y$

**Unsupervised learning**

Given: Training set

$\{x^{(i)}\}_{i=1}^{N}$

Goal: find groups or structures in the data

Discover the intrinsic structure in the data
Supervised Learning: Samples

Classification
Unsupervised Learning: Samples

Wants to use data to improve their knowledge on a task
Sample Data in Unsupervised Learning

Unsupervised Learning:

Columns:
*Features/attributes/dimensions*

Rows:
*Data/points/instances/examples/samples*

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>...</th>
<th>$x_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample n-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sample n</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Unsupervised learning

**Clustering**: partitioning of data into groups of similar data points.

**Dimensionality reduction**: data representation using a smaller number of dimensions while preserving (perhaps approximately) some properties of the data.

**Density estimation**
Some clustering purposes

**Preprocessing stage** to index, compress, or summarize the data

As a tool to **understand the hidden structure** in data or to **group** them
- To gain knowledge (insight into the structure of the data) or
- To group the data when no label is available
Clustering: Example Applications

Clustering docs based on their similarities
Grouping new stories in the Google news site

Market segmentation: group customers into different market segments given a database of customer data.

Community detection in social networks
Clustering of docs

Google news
Dimensionality reduction: Example

How to map the high dimensional data into a lower dimensional space in which the distance is more meaningful.

[Ali Ghodsi, 2006]
Paradigms of ML

**Supervised learning** (regression, classification)
- predicting a target variable for which we get to see examples.

**Unsupervised learning**
- revealing structure in the observed data

**Reinforcement learning**
- partial (indirect) feedback, no explicit guidance
- Given rewards for a sequence of moves to learn a policy and utility functions
Reinforcement

Provides only an indication as to whether an action is correct or not

Data in supervised learning:

(input, correct output)

Data in Reinforcement Learning:

(input, some output, a reward for this output)
Reinforcement Learning

Typically, we need to get a sequence of decisions

Usually, need to decide under uncertainty

Learn a policy that specifies the action for each state
Paradigms of ML

**Supervised learning** (regression, classification)
- predicting a target variable for which we get to see examples.

**Unsupervised learning**
- revealing structure in the observed data

**Reinforcement learning**
- Reasoning under uncertainty
- partial (indirect) feedback, no explicit guidance
- Given rewards for a sequence of moves to learn a policy and utility functions

Other paradigms: semi-supervised learning, weakly supervised, active learning, etc.
Three axes of ML

Data

Task (i.e. what is the type of knowledge that we seek from data)

Algorithm
Three axes of ML

Data
- Fully observed
- Partially observed
- Actively collecting data

Task (i.e. what is the type of knowledge that we seek from data)
- Prediction (i.e. classification or regression)
- Control
- Description

Algorithm
- Parametric models
- Nonparametric models
Parametric models

We consider a parametric boundary (e.g., hyper-plane, hyperbola, ...) and learn its parameters form data.

The set of parameters does not grow with increasing the data.
Nonparametric models

We must store data and for each prediction, we need to process training data.

More data means a more complex model.

Models that grow with the data.
Nonparametric models

**k-NN classifier**

Label for $x$ predicted by majority voting among its k-NN.

Find k nearest training data to the new input and predict its label from the labels of its k nearest neighbors.

The number of points to search scales with the training data.
ML in Computer Science

Why ML applications are growing?

- Improved machine learning algorithms
- Availability of data (Increased data capture, networking, etc)
- Software too complex to write by hand
  - Demand for complex systems (on high-dimensional, multi-modal, or heterogeneous data)
  - Demand for self-customization to user or environment
Relation to other fields

Statistics: the goal is the understanding of the data at hand

Artificial Intelligence: the goal is to build an intelligent agent

Data Mining: the goal is to extract patterns from large-scale data

Data Science: the science encompassing collection, analysis, and interpretation of data

The goal of machine learning is the underlying mechanisms and algorithms that allow improving our knowledge with more data

http://www.cs.cmu.edu/~pradeepr/701
Some Learning Application Areas

- Computer Vision (Photo tagging, face recognition, …)
- Natural language processing (e.g., machine translation)
- Robotics
- Speech recognition
- Autonomous vehicles
- Social network analysis
- Web search engines
- Medical outcomes analysis
- Market prediction (e.g., stock/house prices)
- Computational biology (e.g., annotation of biological sequences)
- Self-customizing programs (recommender systems)
Topics of this course

Regression & generalization
Classification
  Linear classifier
  Probabilistic classifiers
  SVM & kernel
  Decision tree
Neural Networks
Learning Theory
Non-parametric methods
Ensemble learning
Dimensionality reduction
Clustering
Reinforcement Learning