Machine Learning

Lecture 1: Overview

Feng Li

fli@sdu.edu.cn
https://funglee.github.io

School of Computer Science and Technology
Shandong University
• Instructor: Prof Feng Li
• Office: N3-312-1
• Affiliation: Institute of Intelligent Computing (www.iic.sdu.edu.cn)

• Education:
  • PhD, Nanyang Technological University, Singapore
  • MSc, Shandong University, China
  • BSc, Shandong University, China

• Employment:
  • Sep 2018 – Present, Associate Professor, Shandong University, China
  • Nov 2015 – Aug 2018, Assistant Professor, Shandong University, China
  • Nov 2014 – Nov 2015, Research Fellow, National University of Singapore, Singapore

• Research Interests:
  • Distributed Algorithms and Systems, Applied Optimization, Wireless Networking, Internet of Things
Course Information

• Website: https://funglee.github.io/ml/ml.html

• Grades: Labs (35%) + homeworks (15%) + final exam (50%)

• Teaching Assistants (TAs):
  • Ms Yuan Yuan (yuan930126 AT 163 DOT com)
  • Mr Qi Luo (luoqi4110217 AT hotmail DOT com)
  • Mr Zhiyang Dou (sdudzy AT 163 DOT com)
Suggested Readings

• Hang Li, Statistical Machine Learning (2nd Ed.), The Tsinghua Press, 2019
• Zhihua Zhou, Machine Learning (1st Ed.), The Tsinghua Press, 2019
• Trevor Hastie, The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd Ed.), World Publishing Corporation, 2015
• Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press.
• Ian Goodfellow, Yoshua Bengio, Deep Learning, People’s Posts and Telecommunications Press, 2016 (Online: http://www.deeplearningbook.org/)
• Simon Haykin, McMaster, Neural Networks and Learning Machines (3rd Ed.), China Machine Press, 2009
• Christopher M. Bishop, Pattern Recognition and Machine Learning (1st Ed.), Springer, 2006
Prerequisite Courses

- Linear algebra
- Calculus
- Probability theory
- Statistics
- Information theory
- Convex Optimization
Remarks

• Lectures are important, but not enough
• You should review what have been taught with more hours than the class hours/week
• You should be familiar with all terminologies related with this course
• You should understand the theories behind machine learning techniques
• Practice what you have learned
• Work hard !!!
Definition of Machine 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, Machine Learning]
Examples

Example 1:

- T: Playing checkers
- P: Percentage of games won against an arbitrary opponent
- E: Playing practice games

Example 2:

- T: Recognizing hand-written words
- P: Percentage of words correctly classified
- E: Database of human-labeled images of handwritten words
Examples (Contd.)

- **Example 3:**
  - **T:** Categorize email messages as spam or legitimate.
  - **P:** Percentage of email messages correctly classified.
  - **E:** Database of emails, some with human-given labels

- **Example 4:**
  - **T:** Driving on four-lane highways using vision sensors
  - **P:** Average distance traveled before a human-judged error
  - **E:** A sequence of images and steering commands recorded while observing a human driver.
Steps to Design a Learning System

- Choose the training experience
- Choose exactly what is to be learned, i.e. the target function.
- Choose how to represent the target function.
- Choose a learning algorithm to infer the target function from the experience.
Training Experience

- **Direct experience**: Given sample input and output pairs for a useful target function.
  - Checker boards labeled with the correct move, e.g. extracted from record of expert play

- **Indirect experience**: Given feedback which is not direct I/O pairs for a useful target function.
  - Potentially arbitrary sequences of game moves and their final game results.

- **Credit/Blame Assignment Problem**: How to assign credit blame to individual moves given only indirect feedback?
Sources of Training Data

- Provided random examples outside of the learner's control.
  - Negative examples available or only positive?
- Good training examples selected by a “benevolent” teacher.
  - “Near miss” examples
- Learner can query an oracle about class of an unlabeled example in the environment.
- Learner can construct an arbitrary example and query an oracle for its label.
- Learner can design and run experiments directly in the environment without any human guidance.
Applications of Machine Learning

- **Document Search**
  - Given counts of words in a document, determine what its topic is.
  - Group documents by topic without a pre-specified list of topics.

- **Image/Video Understanding**
  - Given an image/video, determine what objects it contains.
  - Determine what semantics it contains.
  - Determine what actions it contains.
Applications of Machine Learning (Contd.)

- **Cancer Diagnosis**
  - Given data on expression levels of genes, classify the type of tumor
  - Discover categories of tumors having different characteristics.

- **Marketing**
  - Given data on age, income, etc., predict how much each customer spends
  - Discover how the spending behaviors of customers are related
  - Fair amount of data on each customer, but messy
  - May have data on a very large number of customer.
Example 1: Handwritten Digit Recognition

• Handcrafted rules will result in large number of rules and exceptions
• Better to have a machine that learns from a large training set
• Handwriting recognition cannot be done without machine learning

0 1 2 3 4
5 6 7 8 9
Example 2: Autonomous Driving-ALVINN

- Drives 70 mph on a public highway (a predecessor of Google car)
- 30 outputs for steering
- 4 hidden units
- 30×32 pixels as input
Example 3: Breast Cancer Diagnosis
Two Questions?

Why is machine learning necessary?

- Learning is a hallmark of intelligence; many would argue that a system that cannot learn is not intelligent.
- Without learning, everything is new; a system that cannot learn is not efficient.

Why is learning possible?

- Because there are regularities in the world.
Categories of Machine Learning

- Supervised learning: learning with a teacher
  - Training examples with labels are given
- Unsupervised learning: learning without a teacher
  - Training examples without labels
- Reinforcement learning: learning by interacting
- Semi-supervised learning: partially supervised learning
- Active learning: actively making queries
Supervised Learning

In the ML literature, a supervised learning problem has the following characteristics:

- We are primarily interested in prediction.
- We are interested in predicting only one thing.
- The possible values of what we want to predict are specified, and we have some training cases for which its value is known.

The thing we want to predict is called the **target** or the **response variable**

Usually, we need labeled training data
Supervised Learning (Contd.)

- For **classification** problem, we want to predict the class of an item.
  - The type of tumor, the topic of a document, the semantics contained in an image, whether a customer will purchase a product.

- For a **regression** problem, we want to predict a numerical quantity.
  - The amount of customer spends, the blood pressure of a patient, etc.

- To make predictions, we have various inputs,
  - Gene expression levels for predicting tumor type, age and income for predicting amount spent, the features of images with known semantics.
Classification and Regression

- Classification: finding decision boundaries
- Regression: fitting a curve/plane to data
Supervised Classification Problem

- Cancer diagnosis (training set)

<table>
<thead>
<tr>
<th>Patient ID</th>
<th># of Tumors</th>
<th>Avg Area</th>
<th>Avg Density</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>20</td>
<td>118</td>
<td>Malignant</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>15</td>
<td>130</td>
<td>Benign</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>10</td>
<td>52</td>
<td>Benign</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>30</td>
<td>100</td>
<td>Malignant</td>
</tr>
</tbody>
</table>

- Use the above training set to learn how to classify patients where diagnosis is not known (test set):

<table>
<thead>
<tr>
<th>Patient ID</th>
<th># of Tumors</th>
<th>Avg Area</th>
<th>Avg Density</th>
<th>Diagnosis</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>4</td>
<td>16</td>
<td>95</td>
<td>?</td>
</tr>
<tr>
<td>102</td>
<td>9</td>
<td>22</td>
<td>125</td>
<td>?</td>
</tr>
<tr>
<td>103</td>
<td>1</td>
<td>14</td>
<td>80</td>
<td>?</td>
</tr>
</tbody>
</table>
How to Make Predictions?

Main methods
• We can train a model by using the training data to estimate parameters of it
• Use these parameters to make predictions for the test data.
• Such approaches save computation when we make predictions for test data.
• That is, estimate parameters once, use them many times.
• E.g. Linear regression $y = \theta_0 + \sum_{j=1}^{n} \theta_j x_j$

Other methods
• Nearest neighbor like methods, which need to store training data
Nearest-Neighbor Methods

- Make predictions for test data based on a subset of training cases, e.g., by approximating the mean, median or mode of \( p(y \mid x) \)

\[
y = \frac{1}{K} \sum_{i \in N_K(x)} y_i
\]

- Big question: How to choose \( K \)?
  - If \( K \) is too small, we may “overfitting”, but if \( K \) is too big, we will average over training cases that aren't relevant to the test case
Comparisons

These two methods are opposite w.r.t. computation.

- NN like methods are memory-based methods. We need to remember all the training data.
- Linear regression, after getting parameters, can forget the training data, and just use the parameters.

They are also opposite w.r.t. to statistical properties.

- NN makes few assumptions about the data, and has a high potential for overfitting.
- Linear regression makes strong assumption about the data, and consequently has a high potential for bias.
Unsupervised Learning

For an unsupervised learning problem, we do not focus on prediction of any particular thing, but rather try to find interesting aspects of the data.

Examples:

- We may find clusters of patients with similar symptoms, which we call diseases.
- We may find clusters of large number of images.
Clustering
Reinforcement Learning

• Learning from interaction (with environment)
• Goal-directed learning
• Learning what to do and its effect
• Trial-and-error search and delayed reward
Reinforcement Learning (Contd.)

- The agent has to **exploit** what it already knows in order to obtain reward, but it also has to **explore** in order to make better action selections in the future.
- Dilemma: neither exploitation nor exploration can be pursued exclusively without failing at the task.
Reinforcement Learning (Contd.)

- Example (Bioreactor)
  - **States**: current temperature and other sensory readings, composition, target chemical
  - **Actions**: how much heating, stirring are required and what ingredients need to be added
  - **Rewards**: moment-by-moment production of desired chemical
Reinforcement Learning (Contd.)

- **Example (Recycling Robot)**
  - **States**: charge level of battery
  - **Actions**: look for cans, wait for can, go recharge
  - **Rewards**: positive for finding cans, negative for running out of battery
**Semi-Supervised Learning**

- As the name suggests, it is in between supervised and unsupervised learning techniques w.r.t the amount of labelled and unlabeled data required for training.

- With the goal of reducing the amount of supervision required compared to supervised learning.

- At the same time, improving the results of unsupervised clustering to the expectations of the user.
Overview of Semi-Supervised Learning

• Constrained Clustering
• Distance Metric Learning
• Manifold based Learning
• Sparsity based Learning (Compressed Sensing).
• Active Learning
Constrained Clustering

- When we have any of the following:
  - Class labels for a subset of the data
  - Domain knowledge about the clusters
  - Information about the “similarity” between objects
  - User preferences

- May be pairwise constraints or a labeled subset
  - Must-link or cannot-link constraints
  - Labels can always be converted to pairwise relations

- Can be clustered by searching for partitionings that respect the constraint

- Recently the trend is toward similarity-based approaches
Constrained Clustering (Contd.)
Active Learning

- Basic idea:
  - Traditional supervised learning algorithms passively accept training data.
  - Instead, query for annotations on informative images from the unlabeled data.
  - Theoretical results show that large reductions in training sizes can be obtained with active learning!

- But how to find images that are the most informative?
Active Learning

- One idea uses uncertainty sampling
- Images on which you are uncertain about classification might be informative!

What is the notion of uncertainty?
- Idea: Train a classifier like SVM on the training set
- For each unlabeled image, output probabilities indicating class membership
- Estimate probabilities can be used to infer uncertainty
- A one-vs-one SVM approach can be used to tackle multiple classes
Challenges for Machine Learning

- Handling complexity
  - Involve many variables, how can we handle this complexity without getting into trouble.

- Optimization and Integration
  - Usually involve finding the best values for some parameters (an optimization problem), or average over many plausible values (an integration problem). How can we do this efficiently when there are many parameters?

- Visualization
  - Understanding what's happening is hard, 2D? 3D?

- All these challenges become greater when there are many variables or parameters, the so-called “curse of dimensionality”.
  - But more variables also provide more information
  - A blessing? A curse?
How to handle complexity

- Properly dealing with complexity is a crucial issue for machine learning

- Limiting complexity is one approach
  - Use a model that is complex enough to represent the essential aspects of the problem, but that is not so complex that overfitting occurs
  - Overfitting happens when we choose parameters of a model that fit the data we have very well, but do poorly on new data (poor generalization ability)
  - Cross-validation, regularization

- Reducing dimensionality is another possibility.
  - It is apparent that things become simpler if can find out how to reduce the large number of variables to a small number.

- Averaging over complexity is the Bayesian approach.
  - Use as complex a model might be needed, but do not choose a single parameter values. Instead, average the predictions found using all the parameter values that fit the data reasonably well, and which are plausible for the problem
Example of Complexity
Example of Complexity (Contd.)

- Graphs of the root-square error, evaluated on the training set and on an independent test set for various degree.

![Graph of root-square error](image-url)
Example of Complexity (Contd.)

- If we make predictions using “the best” parameters of a model, we have to limit the number of parameters to avoid overfitting.
  
  - For this example, the model with $\text{degree}=3$ seems good. We might be able to choose a good value for $M$ using the method of “cross validation”, which looks for the value that does best at prediction one part of the data from the rest of the data.
Reducing Dimensionality

- Suppose dimension of input data is 1000, can we replace these with fewer ones, without loss of information.

- On simple way is to use PCA (Principal Component Analysis)
  - Suppose that all data are in a space, we first find the direction of highest variance of these data points, then the direction of second-highest variance that is orthogonal to the rst one, so on and so forth
  - Replace each training sample by the projections of the inputs on some directions.
  - Might discard useful info., but keep most of it.
Thanks!

Q & A