Lecture 1
Introduction

Lin ZHANG, PhD
School of Software Engineering
Tongji University
Fall 2019
Contact Information

Room 408L, Jishi Building
Email: cslinzhang@tongji.edu.cn

All material can be found at

http://sse.tongji.edu.cn/linzhang
Materials

• Major materials
  • My slides
• References
  • Some papers
  • 《机器学习》，周志华
Examination

- Homework 30%: 3 times, and each time 10%.
- Paper reading and presentation 20%
  - Read a paper related to machine learning and do a presentation
- Final report and presentation 50%
  - Select a problem related to your research direction, try to solve it with machine learning techniques, write an essay and finally do a presentation
- Being absent >=1/3 lectures, you will fail this course
Arrangement of Lectures (temporarily)

- Basic concepts and model evaluation
- AdaBoost and Cascade structure
- Principle Component Analysis
- Sparse representation based classification
- Linear model
- Neural network and CNN
- Applications of CNN
- Generative Adversarial Networks
- Graphical Convolutional Networks
- Other topics*
A little history about AI

人工智能

1956年，麦卡锡召集哈佛大学、麻省理工学院、IBM公司、贝尔实验室的研究人员召开达特茅斯会议正式提出“人工智能”

2006年达特茅斯会议当事人重聚，左起：摩尔、麦卡锡、明斯基、赛弗里奇、所罗门诺夫

人工智能是指计算机系统具备的能力，该能力可以履行原本只有依靠人类智慧才能完成的复杂任务
人工智能

人工智能的最终目标

探讨智能形成的基本机理，研究利用自动机模拟人的思维过程

人工智能的近期目标

研究如何使计算机去做那些靠人的智力才能做的工作
人工智能的研究范式及历程

**符号主义**：采用知识表达和逻辑符号系统来模拟人类的智能，试图对智能进行宏观研究（Knowledge-driven）

- 1950-1960：二者独立并驾齐驱
- 1960-1970：符号主义：专家系统和知识工程为主流

**联接主义**：始于W.S. McCulloch和皮兹（Pitts）的先驱工作，直到目前的深度学习，是微观意义上的探索（Data driven）

- 1970-1980：符号主义滞步，日本第五代计算机失败，联接主义蓬勃发展
- 1990-2015：联接主义占据主导；同时模糊逻辑取得重大进展

**生物启发的智能**：依赖于生物学、脑科学、生命科学和心理学等学科的发现，将机理变为可计算的模型（Biology mechanism driven）

- 2016后：生物启发的智能：跨模态的信息处理
A little history about AI

人工智能的研究领域划分

人工智能

符号智能
- 图搜索
- 自动推理
- 不确定性推理

计算智能
- 符号学习
- 神经计算
- 进化计算
- 免疫计算
- 蚁群计算

机器学习
- 归纳学习
- 模式识别
- 统计学习
- 深度学习
- 计算机视觉
- 语音识别
- 自然语言处理

机器感知
- 图像识别
A little history about AI

人工智能产业发展加速明显

自然语言处理（NLP）：
微软Skype Translator实现同声传译

计算机视觉（CV）：
格林深瞳的视频监控可智能识别犯罪

计算机视觉（CV）：
Face++的人脸识别云服务

感知、规划和决策：
Google无人驾驶汽车
A little history about AI

人工智能成为世界焦点

人工智能目前已经成为世界各国关注的焦点。2017年7月，中国政府发布了“新一代人工智能发展规划”

✓ 人工智能是开启未来智能世界的秘钥，是未来科技发展的战略制高点；谁掌握人工智能，谁就将成为未来核心技术的掌控者
10 Breakthrough Technologies 2017 (MIT Tech Review)

- Reversing Paralysis
- Self Driving
- Paying with Your Face
- Practical Quantum Computers
- The 360-Degree Selfie
- Hot Solar Cells
- Gene Therapy 2.0
- The Cell Atlas
- Botnets of Things
- Reinforcement Learning

The core is machine learning
傍晚，小街路面上沁出微雨后的湿润，和煦的细风吹来，抬头看看天边的晚霞，嗯，明天又是一个好天气。走到水果摊旁，挑了个根蒂蜷缩、敲起来声音浊响的青绿西瓜，一边满心期待着皮薄肉厚瓢甜的爽落感，一边愉快地想着：这学期狠下了功夫，基础概念弄得清清楚楚，算法作业也是信手拈来，这门课成绩一定差不了！

摘自《机器学习》（周志华著，2016）
What is machine learning?

• Gives "computers the ability to learn without being explicitly programmed" (Arthur Samuel in 1959)

Arthur Lee Samuel
(December 5, 1901 – July 29, 1990)

• It explores the study and construction of algorithms that can learn from and make predictions on data

• It is employed in a range of computing tasks where designing and programming explicit algorithms with good performance is difficult or unfeasible

Supervised VS Unsupervised

• Supervised learning
  – It will infer a function from labeled training data
  – The training data consists of a set of training examples
  – Each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal)

• Unsupervised learning
  – Trying to find hidden structure in unlabeled data
  – Since the examples given to the learner are unlabeled, there is no error or reward signal to evaluate a potential solution
  – Such as PCA, K-means (a clustering algorithm)
About sample

• Attribute (feature), attribute value, label, and example

<table>
<thead>
<tr>
<th>Features</th>
<th>Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>色泽，根蒂，敲声</td>
<td>{好瓜，坏瓜}</td>
</tr>
</tbody>
</table>

{青绿，蜷缩，浊响：好瓜}

feature values  label value

one example
Training sample and training set

A training set comprising $m$ training samples,

$$D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$$

where $x_i = (x_{i1}, x_{i2}, \ldots, x_{id}) \in \chi$ is the feature vector of $i$th sample and $y_i \in \zeta$ is its label.

By training, our aim is to find a mapping,

$$f : \chi \mapsto \zeta$$

based on $D$

If $\zeta$ comprises discrete values, such a prediction task is called “classification”; if it comprises real numbers, such a prediction task is called “regression.”
Training, testing, and validation

• Training sample and training set

• Test set
  – A test set is a set of data that is independent of the training data, but that follows the same probability distribution as the training data
  – Used only to assess the performance of a fully specified classifier
Training, testing, and validation

• Training sample and training set
• Test set
• Validation set
  – In order to avoid overfitting, when any classification parameter needs to be adjusted, it is necessary to have a validation set; it is used for model selection
  – The training set is used to train the candidate algorithms, while the validation set is used to compare their performances and decide which one to take
Overfitting, Generalization, and Capacity

• Overfitting
  – It occurs when a statistical model describes random error or noise instead of the underlying relationship
  – It generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations
  – A model that has been overfit will generally have poor predictive performance, as it can exaggerate minor fluctuations in the data
Overfitting, Generalization, and Capacity

• Overfitting

• Generalization
  – Refers to the performance of the learned model on new, previously unseen examples, such as the test set
Overfitting, Generalization, and Capacity

- Overfitting
- Generalization

Example: Linear regression (housing prices)

\[ \rightarrow \theta_0 + \theta_1 x \]

“Underfit” “High bias”

\[ \rightarrow \theta_0 + \theta_1 x + \theta_2 x^2 \]

“Just right”

\[ \rightarrow \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 \]

“Overfit” “High variance”
Overfitting, Generalization, and Capacity

- Overfitting
- Generalization

Example: Logistic regression

$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$  
($g = \text{sigmoid function}$)

\[ g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2) \]

"Underfit"

"Overfit"

http://blog.osdn.net/zouxy09
Overfitting, Generalization, and Capacity

- Overfitting
- Generalization
- Capacity
  - Measures the complexity, expressive power, richness, or flexibility of a classification algorithm
  - Ex, DCNN (deep convolutional neural networks) is powerful since its capacity is very large

\[ y^* = b + \omega x, \quad y^* = b + \omega_1 x_1 + \omega_2 x_2, \quad y^* = b + \sum_{i=1}^{10} \omega_i x_i \]

higher capacity
Overfitting, Generalization, and Capacity

- **Underfitting**
- **Appropriate model complexity**
- **Overfitting**

higher capacity
Performance Evaluation

Given a sample set (training, validation, or test)

\[ D = \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\} \]

To assess the performance of the learner \( f \), we need to compare the prediction \( f(x) \) and its ground-truth label \( y \)

For regression task, the most common performance measure is MSE (mean squared error),

\[ E(f; D) = \frac{1}{m} \sum_{i=1}^{m} (f(x_i) - y_i)^2 \]
Performance Evaluation (for classification)

• Error rate
  – The ratio of the number of misclassified samples to the total number of samples

\[
E(f; D) = \frac{1}{m} \sum_{i=1}^{m} 1(f(x_i) \neq y_i)
\]

• Accuracy
  – It is derived from the error rate

\[
acc(f; D) = \frac{1}{m} \sum_{i=1}^{m} 1(f(x_i) = y_i) = 1 - E(f; D)
\]
Performance Evaluation (for classification)

- Precision and Recall

<table>
<thead>
<tr>
<th>Ground truth</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>positive</td>
</tr>
<tr>
<td>positive</td>
<td>True Positive (TP)</td>
</tr>
<tr>
<td>negative</td>
<td>False Positive (FP)</td>
</tr>
</tbody>
</table>

\[
\text{precision} = \frac{TP}{TP + FP}
\]

\[
\text{recall} = \frac{TP}{TP + FN}
\]
Performance Evaluation (for classification)

• Precision and Recall
  – Often, there is an inverse relationship between precision and recall, where it is possible to increase one at the cost of reducing the other
  – Usually, PR-curve is not monotonic
Performance Evaluation (for classification)

• Precision-recall should be used together; it is meaningless to use only one of them

• However, in many cases, people want to know explicitly which algorithm is better; we can use $F$-measure

$$F_\beta = \frac{(1 + \beta^2) \times P \times R}{(\beta^2 \times P) + R}$$
Performance Evaluation (for classification)

• To derive a single performance measure

Varying threshold, we can have a series of \((P, R)\) pairs,

\[(P_1, R_1), (P_2, R_2), ..., (P_n, R_n)\]

Then,

\[P_{\text{macro}} = \frac{1}{n} \sum_{i=1}^{n} P_i \quad R_{\text{macro}} = \frac{1}{n} \sum_{i=1}^{n} R_i\]

\[F_{\beta-\text{macro}} = \frac{(1 + \beta^2) \times P_{\text{macro}} \times R_{\text{macro}}}{\left(\beta^2 \times P_{\text{macro}}\right) + R_{\text{macro}}}\]
Class-imbalance Issue

• Problem definition
  – It is the problem in machine learning where the total number of a class of data is far less than the total number of another class of data
  – This problem is extremely common in practice

• Why is it a problem?
  – Most machine learning algorithms work best when the number of instances of each classes are roughly equal
  – When the number of instances of one class far exceeds the other, problems arise
Class-imbalance Issue

• How to deal with this issue?
  – Modify the cost function
  – Under-sampling, throwing out samples from majority classes
  – Oversampling, creating new virtual samples for minority classes
    » Just duplicating the minority classes could lead the classifier to overfitting to a few examples
    » Instead, use some algorithm for oversampling, such as SMOTE (synthetic minority over-sampling technique)[1]

Class-imbalance Issue

- Minority oversampling by SMOTE\textsuperscript{[1]}

\textbf{Add new minority class instances by:}

- For each minority class instance $c$
  - neighbours = Get KNN(5)
  - $n$ = Random pick one from neighbours
  - Create a new minority class $r$ instance using $c$’s feature vector and the feature vector’s difference of $n$ and $c$ multiplied by a random number
    - i.e. $r$.feats = $c$.feats + ($n$.feats - $c$.feats) * rand(0,1)

\textsuperscript{[1]} N.V. Chawla \textit{et al.}, SMOTE: Synthetic Minority Over-sampling Technique, J. Artificial Intelligence Research 16: 321-357, 2002