Introduction to AI

Lecture #4
What will we cover today?

1. What is AI?
2. Turing
3. Machine learning
   - Linear predictors
   - Classification and regression
   - Statistical Approaches
What is AI?

• “The study of computations that make it possible to perceive, reason and act.”
  – Winston 1992

• Four possible goals of AI:
  – Think like humans
  – Think rationally
  – Act like humans
  – Act rationally
What is AI?

- Four possible goals of AI:
  - Think like humans
    - Cognitive science
  - Think rationally
    - rules, logic, but uncertainty
  - Act like humans
  - Act rationally
    - choose behavior that maximizes goal achievement
Act Like Humans
Act Like Humans

• Can machines think?
• Turing test
  – NLP
  – Knowledge representation
  – Automated reasoning
  – Machine learning
• Turing test 2.0!
  – Computer vision
  – Robotics

• Loebner Prize
On the cover this week: All systems go. At last — a computer program that can beat a champion Go player.

STATE OF THE ART

January 28th, 2016

IBM Watson
Deep Blue
Pac Man!

• Plays Pac Man and always wins
• Learns by trial and error Q-learning, a variant of reinforcement learning
• Good to go after 50 trial runs
Machine Learning

• Learning without being programmed
• Types of learning
  – Supervised
  – Unsupervised
  – Reinforcement
Linear predictors

• Simplest of all machine learning tools
• Covers Classification and regression
Linear predictors

- Simplest of all machine learning tools
- Covers Classification and regression
- For example, spam classification

Amount Won: 5.5 Million united states dollars (USD) PIN;201306
****************************************************
YOUR WINNING INFORMATIONS
BATCH NUMBER:MFI/07/APA-43658
REFERENCE NUMBER: 2008234522
Draw Date: JANUARY, 2016
Amount Won: 5.5 Million UNITED STATES DOLLARS (USD)
CLEARANCE PIN;201306
********************************************************
Please send the below information to
Mrs Karen Peterson
E-mail: collationscreening2014@gmail.com
Application: Spam Classification

- Input: $x =$ email message
- Output: $y \in \{\text{spam, not-spam}\}$

- Objective: obtain a predictor $f$
  \[ x \rightarrow f \rightarrow y \]
- Where $f$ is a linear function:
  - $f(x) = a_1x_1 + a_2x_2 + \ldots + a_nx_n$
  - $a_i$ is a parameter we want to learn
  - $x_i$ is a feature of $x$
Linear predictors: Binary Classification

• Input: $x =$ email message
• Output: $y \in \{\text{spam, not-spam}\}$

• Objective: obtain a predictor $f$

  $$x \rightarrow f \rightarrow y$$
Linear predictors: Regression

• Input: $x = \text{location, year}$
• Output: $y \in \mathbb{R}$ (house price)

• Objective: obtain a predictor $f$
  
  $x \rightarrow f \rightarrow y$
Learning Framework

\[ \mathcal{D}_{\text{train}} \rightarrow \text{Learner} \rightarrow f \rightarrow x \rightarrow y \]
Feature Selection

• Task: Given a string, predict whether it is an email address.
  abc@gmail.com

• Feature selection: Given an input $x$, which properties might be useful for predicting $y$?
Feature Selection

• Task: Given a string, predict whether it is an email address.
  abc@gmail.com

• Feature selection: Given an input $x$, which properties might be useful for predicting $y$?
  – contains_@
Feature Selection

• Task: Given a string, predict whether it is an email address.
  abc@gmail.com

• Feature selection: Given an input $x$, which properties might be useful for predicting $y$?
  – contains_@
  – endswith_.com
  – endswith_.org
  – contains_%
Feature weights

• contains_@: 3
• endswith_.com: 2.4
• endswith_.org: 1.3
• contains_%: -2
• !contains_alpha: -4
Linear combination

• For a new example, sum the weights
  – If the sum is >= 0, output Yes
  – else No

  ▪ String 1: “sam@sccourse.com”

  ▪ String 2: “0.5%”

contains_@: 3
endswith_.com: 2.4
endswith_.org: 1.3
contains_%: -2
!contains_alpha: -4
Let’s add some math!

\[ y = \text{sign}(\mathbf{w} \cdot \varphi(x)) \]
\[ = \text{sign}\left(\sum_{i=1}^{I} w_i \cdot \varphi_i(x)\right) \]

- \( x \): the input
- \( \varphi(x) \): vector of feature functions \( \{\varphi_1(x), \varphi_2(x), \ldots, \varphi_I(x)\} \)
- \( \mathbf{w} \): the weight vector \( \{w_1, w_2, \ldots, w_I\} \)
- \( y \): the prediction, +1 if “yes”, -1 if “no”
  - (\( \text{sign}(y) \) is +1 if \( y \geq 0 \), -1 otherwise)
How to make machines understand natural language?

Two possible solutions:

• **Symbolic approach**
  – Encode all the required information into computer
  – Knowledge about meaning, similarities, dependencies, etc.

• **Statistical approach**
  – Infer language properties from language samples
  – Learn to understand language by observing how people do it
A very simple example

• Task: determine placement of articles (a, the, or none) in English text:

• Natural Language Processing (NLP) is the computerized approach to analyzing text that is based on both a set of theories and a set of technologies. And, being a very active area of research and development, there is not a single agreed-upon definition that would satisfy everyone.
A Symbolic Approach

• Write explicit rules for article placement:
  1. Type of noun (countable, uncountable)
  2. Reference (specific, generic)
  3. Information value (given, new)
  4. Number (singular, plural)

• Then add more rules to handle exceptions
  – “The” is used with newspaper titles (The Times),

• And exceptions to exceptions...
  – No article used in names of magazines (Time)
A naïve statistical approach

• Collect a large collection of texts relevant to your domain (e.g., newspaper text)

• For each noun seen during training, compute its probability to take a certain determiner

\[ p(\text{determiner}|\text{noun}) = \frac{\text{freq}(\text{noun, determiner})}{\text{freq}(\text{noun})} \]

• Given a new noun, select a determiner with the highest likelihood as estimated on the training corpus
Does it work?

- **Training:**
  - A corpus of Wall Street Journal (WSJ) news
  - Approx. 25 million words

- **Testing:**
  - Set aside section of approx. 1 million words
  - Prediction accuracy: 71.5%

- **Not great, but surprisingly high for such a simple method**

- **Advantages:**
  - Many nouns always appear with the same determiner
    - “the FBI”, “the defendant”, . . . -- easily learned

- **Disadvantages:**
  - Would it hold up to other types of text?
  - What about unseen words?
A better statistical approach

• Learn generalized rules using features of nouns
  – \( F = \{ \text{noun, number, occurrence, countable, ...} \} \)

<table>
<thead>
<tr>
<th>Noun</th>
<th>plural?</th>
<th>first appearance</th>
<th>determiner</th>
</tr>
</thead>
<tbody>
<tr>
<td>defendant</td>
<td>no</td>
<td>yes</td>
<td>the</td>
</tr>
<tr>
<td>cars</td>
<td>yes</td>
<td>no</td>
<td>null</td>
</tr>
<tr>
<td>FBI</td>
<td>no</td>
<td>no</td>
<td>the</td>
</tr>
<tr>
<td>concert</td>
<td>no</td>
<td>yes</td>
<td>a</td>
</tr>
</tbody>
</table>

Learn classification function \( D: F \rightarrow \{ a, \text{the}, \text{null} \} \)
Where else statistical methods?

- Speech recognition
  - Transforming signals to words
- Parsing
  - Transforming word strings to trees
- Machine Translation
  - Strings in language X to strings in language Y
- Natural Language Generation
  - Internal representations to strings
Learning to parse

• Training: parallel corpus of sentences and parse trees
  – Penn Treebank = 50,000 sentences with associated trees
  – Usual set-up: 40,000 training sentences, 2400 to test
Learning to translate

• Training: parallel text in 2 languages
  – UN, Canadian Parliament, EU, Hong Kong
  – Human translations – literature, news, science

Он благополучно избегнул встречи с своей хозяйкой на лестнице.

He had successfully avoided meeting his landlady on the staircase.

• Align: pretend target language an elaborate “code” – code breaking techniques
Supervised vs. unsupervised

• Supervised learning = training data required
  – Significant burden to construct training corpus
  – Parsing, MT use supervised methods
• Unsupervised learning = no training data
  – Learn by discovering relationships in data
  – Word segmentation (Chinese), semantic classes
• Semi-supervised learning = bootstrapping
  – Learn initial rules from annotated data
  – Discover more rules by out-propagation
What did we cover today?

1. What is AI?
2. Turing
3. Machine learning
   – Linear predictors
   – Classification and regression
   – Statistical Approaches
Assigned Reading: Paper 3

Online Human-Bot Interactions: Detection, Estimation, and Characterization
– Onur Varol, Emilio Ferrara, Clayton A. Davis, Filippo Menczer, and Alessandro Flammini
– Arxiv March 2017

• Responses due on Feb 14th, 2019 11:59 pm