EECS4404
Introduction to Machine Learning and Pattern Recognition

Lecture 1

Amir Ashouri

York University

September 4, 2019

Contents are largely based on the slides of Prof. Ruth Urner

Organization of the class

Basic Information

- Instructor: Amir Ashouri (aashouri@eecs.yorku.ca)
- Office Hours: Wednesdays (To be announced)
- Website: Currently being setup on wiki pages will be announced

Lectures

Lecture times	Mon, Wed, 14:30 - 16:00
Place	CB 129, CC2 11
First lecture	Today :) September 4
Last lecture	Monday, December 2
Reading week	October 12-18
More info	https://registrar.yorku.ca/enrol/dates/fw19#2

Evaluation

Assignments:

- 3 Assignments
- Roughly due end of September/October/November
- Mix of theoretical and programming questions

Tests:

- In-class midterm (tentatively October 23)
- Final exam (December 5 - 20)

`Presentations (for graduate students)`

- `Short presentation (10-15 minutes) on a ML research paper`
- `Paper to be selected in discussion with instructor`

Evaluation

EECS4404

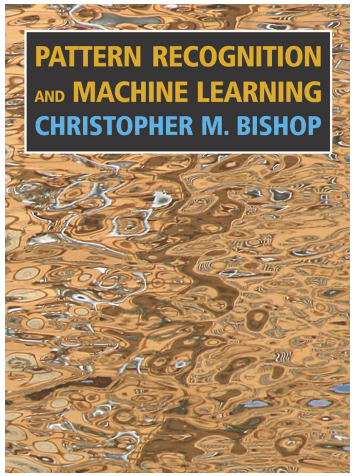
- 30 % Assignments
- 30 % Midterm
- 40 % Final exam

eecs5327

- 25 % Assignments
- 25 % Midterm
- 35 % Final exam
- 15 % Project/Paper presentation

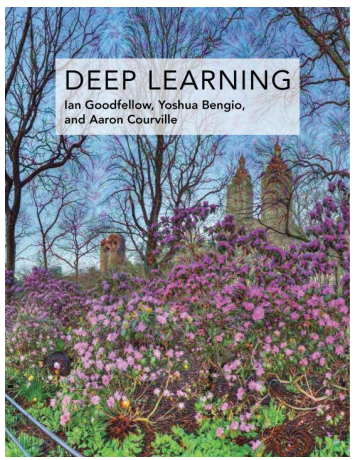
Textbooks

Pattern Recognition and Machine Learning (PRML)
by Christopher M.Bishop. Springer. (2006).



- Available on Amazon.
- Available in bookstore

Deep Learning
by Ian Goodfellow, Yoshua Bengio and Aaron Courville.
www.deeplearningbook.org. (2016)



- Available on Amazon.

- Full content from deeplearning.org website:
https://www.deeplearningbook.org/contents/TOC.html

Material

Lectures will include a mixture of high level motivation and explanations and low-level derivations

Material in readings and lectures will overlap but won't be identical. **You are responsible for both!**

Slides used in class will generally be posted to the course website the day within 24 hours after lecture.

In cases where lectures are delivered on the board, these notes may not be posted. **Don't skip lectures and plan to catch up by looking at the posted notes!**

Programming

Assignments to be in done in **Matlab** and **TensorFlow**:

- We use Python/TensorFlow for our Deep Learning assignment
- We use Matlab for the other assignments

- Available on EECS lab machines and can be installed on your own computers

- If you are not familiar with it, start working your way through a tutorial

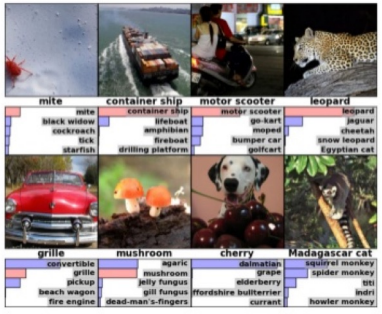
https://www.tensorflow.org/tutorials
https://www.mathworks.com/support/learn-with-matlab-tutorials.html

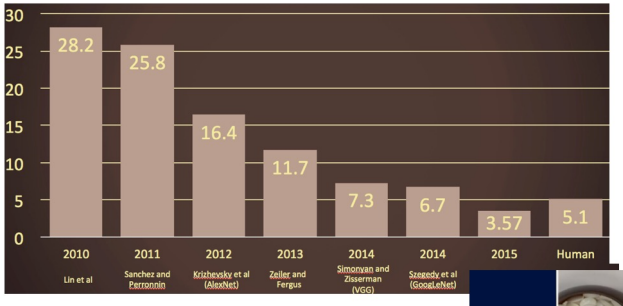
Applications of ML

ImageNet Challenge

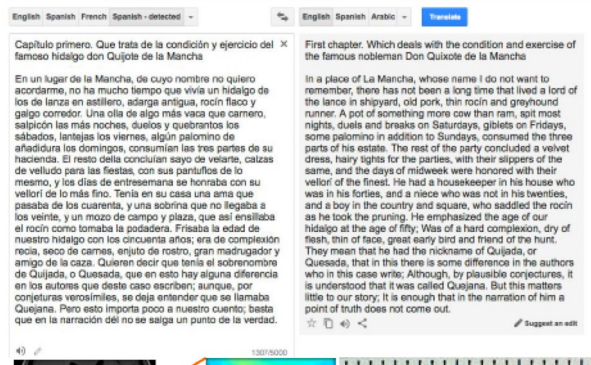
IMAGENET

- 1,000 object classes (categories).
- Images:
  - 1.2 M train
  - 100k test.

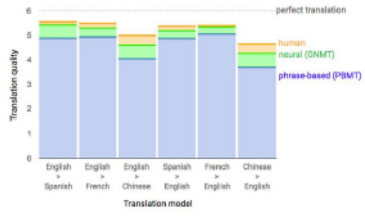




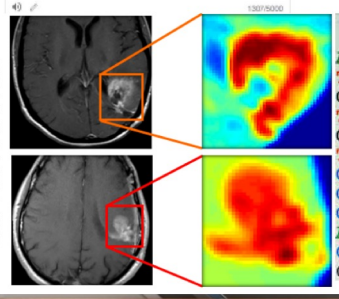
- Natural language processing

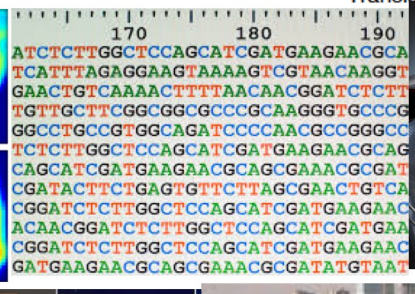


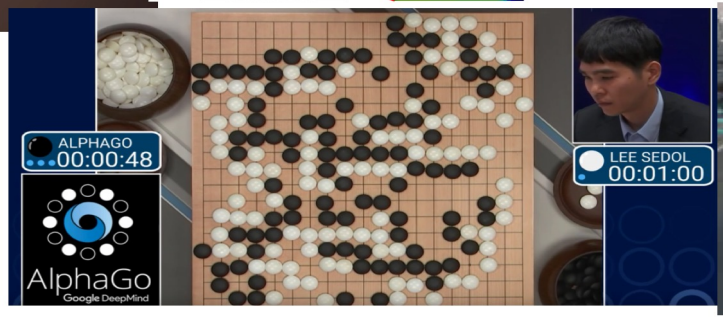
Google's Neural Machine Translation (Wu et al, 2016)

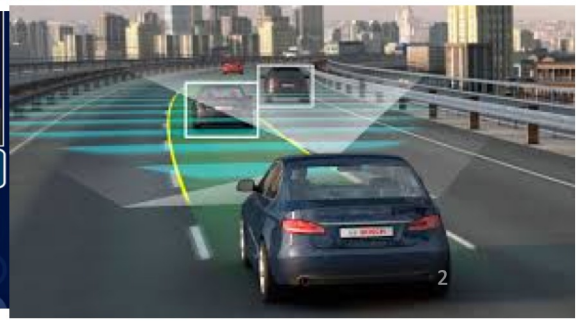


DIAGNOSTICS

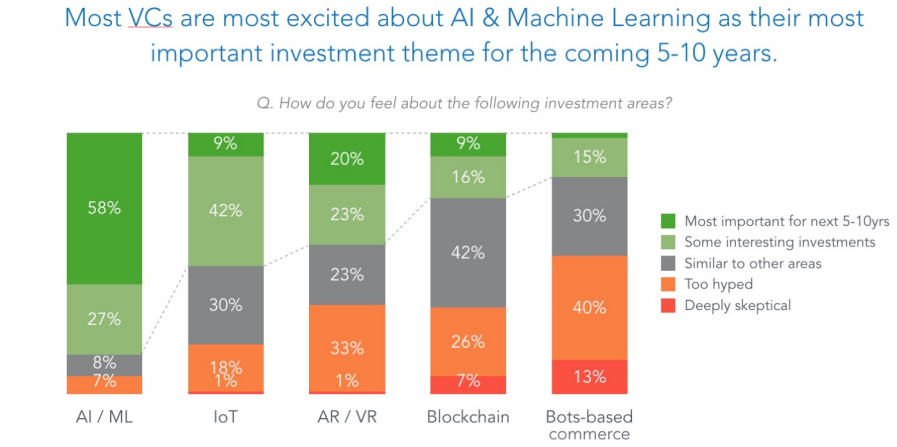


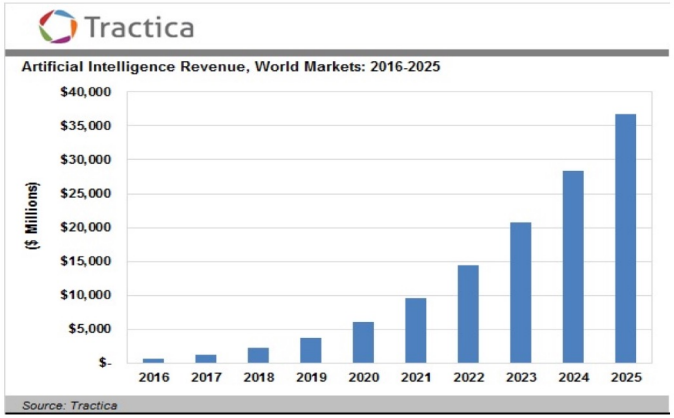




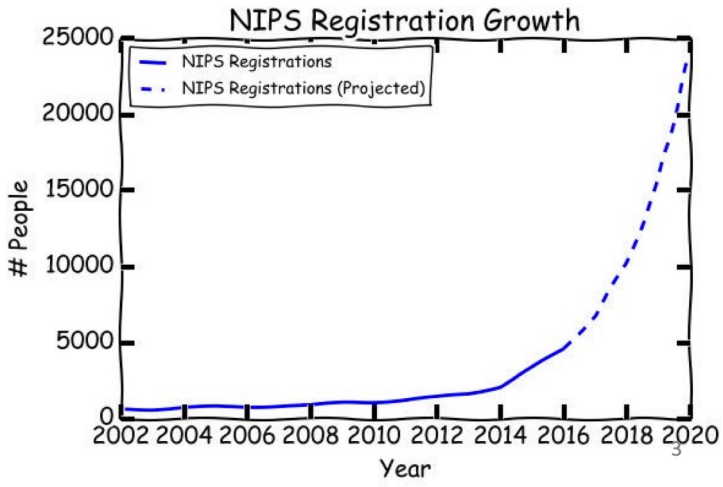


ML Growth


Most VCs are most excited about AI & Machine Learning as their most important investment theme for the coming 5-10 years.
Q. How do you feel about the following investment areas?
58%
27%
8%
7%
9%
42%
30%
18%
1%
20%
23%
23%
33%
1%
9%
16%
42%
26%
7%
15%
30%
40%
13%
AI / ML
IoT
AR / VR
Blockchain
Bots-based commerce
Most important for next 5-10yrs
Some interesting investments
Similar to other areas
Too hyped
Deeply skeptical


Tractica
Artificial Intelligence Revenue, World Markets: 2016-2025
($ Millions)
$40,000
$35,000
$30,000
$25,000
$20,000
$15,000
$10,000
$5,000
$-
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
Source: Tractica

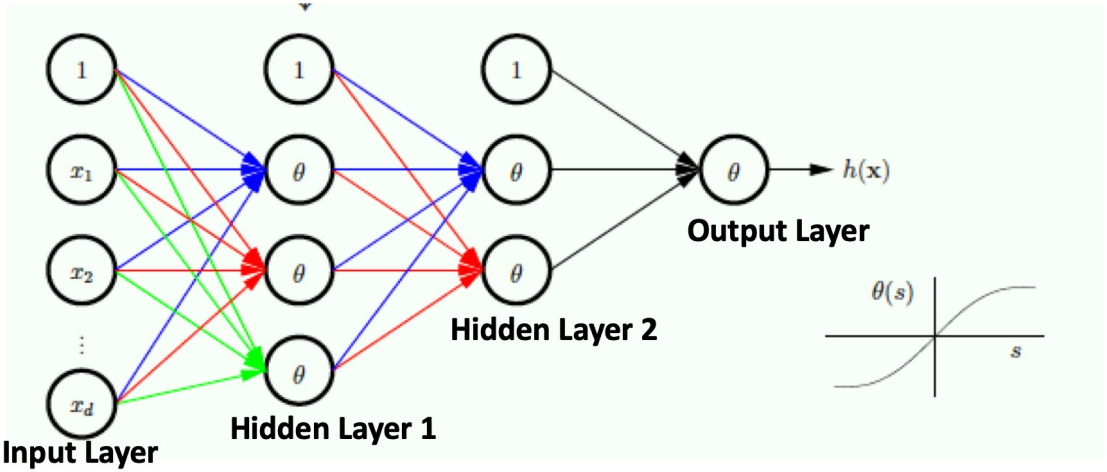

17 Source: Upfront Ventures survey of VCs (N=115), Jan 2017
upfront
Artificial intelligence startups are a global phenomenon
Artificial Intelligence Startup Count by Country
0
875
Companies
Venture Scanner
Data as of April 2017 13


NIPS Registration Growth
NIPS Registrations
NIPS Registrations (Projected)
People
25000
20000
15000
10000
5000
0
2002 2004 2006 2008 2010 2012 2014 2016 2018 2020
Year

NIPS (NeurIPS) Attendance!

NIPS (NeurIPS) 2018 - Montreal

Neural Networks - Architecture



$$h_i^{(1)} = \theta\left(\sum_{j=0}^{d} w_{ji}^{(1)} \cdot x_j\right)$$

Output `i' (Hidden Layer 1)

Weights:Layer 1

Inputs

$$h_i^{(2)} = \theta\left(\sum_{j=0}^{d} w_{ji}^{(2)} \cdot h_j^{(1)}\right)$$

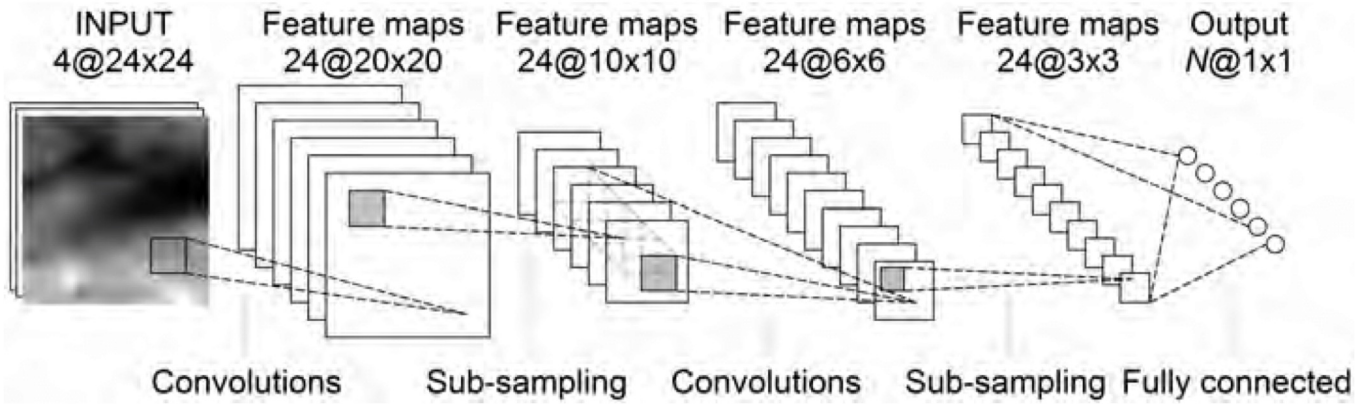
Output `i' (Hidden Layer 2)

Weights: Layer 2

$$y = sign\left(\sum_{j=0}^{d} w_{ji}^{(3)} \cdot h_j^{(2)}\right)$$

Output

Convolutional Neural Networks



- Image statistics are translation invariant (objects and viewpoint translates)
- Expect low-level features to be local (e.g. edge detector)
- Expect high-level features learned to be coarser

Tensor flow Assignments

- **Python based** ML Library released by Google in 2015
- Automatic Training for Neural Networks
- GPU Support (Not Required for Assignments in this courses)

- Installation through Anaconda Environment is Recommended (See Installation Guide on Course Webpage)

- Tons of Resources!
  - Tensorflow.Org Tutorials
  - CS231n Stanford Tutorial (http://cs231n.stanford.edu/)
  - See Course Webpage for a simple tutorial (Updated, Use Chrome Browser)

TensorFlow Example

Tensor flow Example (https://www.tensorflow.org)

```
import tensorflow as tf
x = tf.placeholder(tf.float32, [None, 784])
W = tf.Variable(tf.zeros([784, 10]))
b = tf.Variable(tf.zeros([10]))
y = tf.nn.softmax(tf.matmul(x, W) + b)
```

Initialize Computational Graph

```
cross_entropy = tf.reduce_mean(-tf.reduce_sum(y_ * tf.log(y), reduction_indices=[1]))

train_step = tf.train.GradientDescentOptimizer(0.5).minimize(cross_entropy)
```

Loss Function and Optimizer

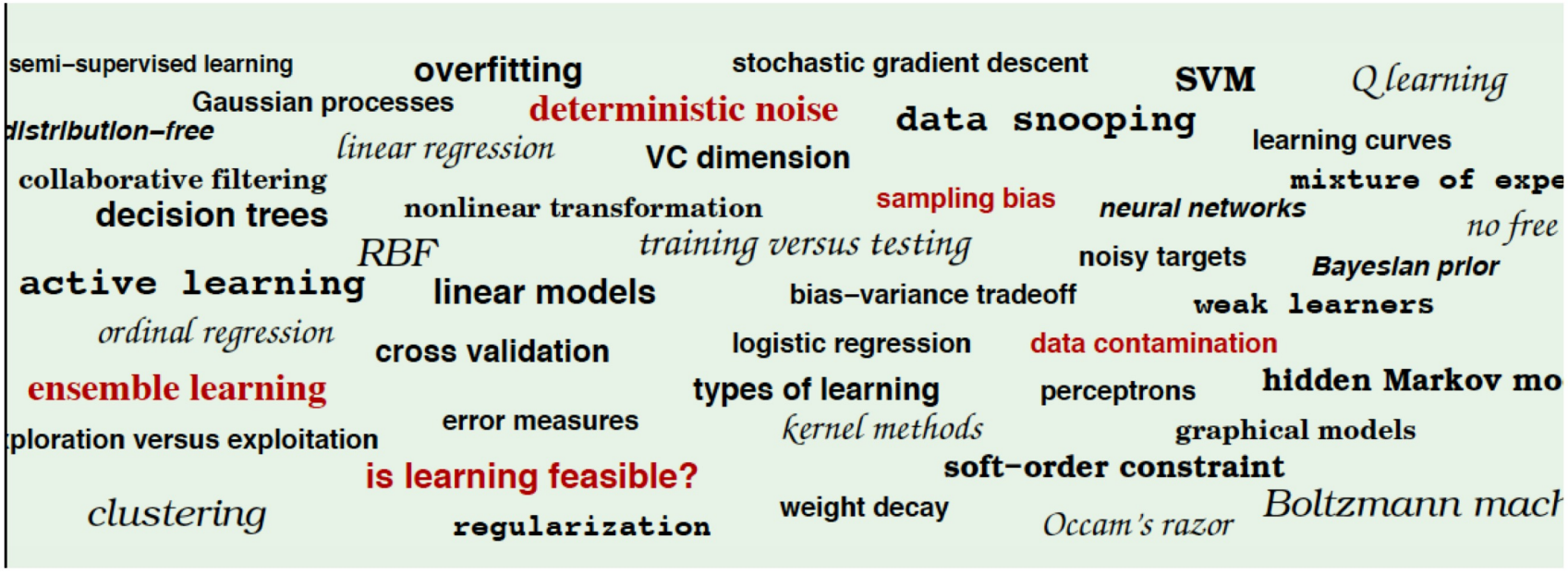
```
sess = tf.InteractiveSession()

tf.global_variables_initializer().run()

for _ in range(1000):
  batch_xs, batch_ys = mnist.train.next_batch(100)
  sess.run(train_step, feed_dict={x: batch_xs, y_: batch_ys})I
```

Training Routing

Bag of ML Jargons



Machine Learning examples

Machine Learning - examples

Self driving cars

Community detection

Fraud detection

Species preservation

Recommender systems

Computational Biology

Logistics

Consumer behavior analysis

Face recognition

Medical diagnosis

Speech recognition

Computer vision

Stock market prediction

Spam filters

Automated translation

Character recognition

Machine Learning

Machine Learning – Why do we need it?

Machine Learning – Why do we need it?

Some tasks are too complex to be implemented directly:

- Self driving cars
- Speech recognition
- Complex rules for classification tasks on high dimensional data
  - Fraud detection
  - Document classification

Machine Learning – Why do we need it?

Some tasks are too complex to be implemented directly:

- Self driving cars
- Speech recognition
- Complex rules for classification tasks on high dimensional data
  - Fraud detection
  - Document classification

→ Learn a program based on data!

What is machine learning?

What is machine learning?

First explanation:

- Development of algorithms which allow a computer to "learn" specific tasks from training examples.
- Learning means that the computer should not just memorize the seen examples, but predict well on previously unseen instances
- Ideally, the computer should use the examples to extract a general "rule" how the specific task has to be performed correctly.