COMP 562: Introduction to Machine Learning

Mahmoud Mostapha

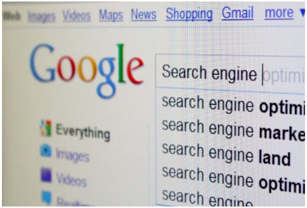
Department of Computer Science

University of North Carolina at Chapel Hill

mahmoudm@cs.unc.edu

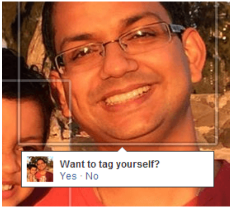
August 22, 2018

Machine Learning in Everyday Life



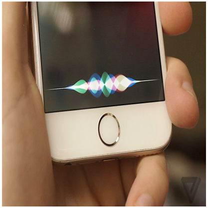
Search Engine Rankings

Email Spam Filtering

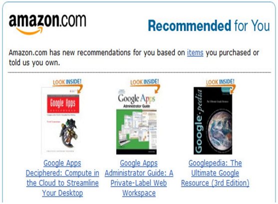


Social Media Services

online Banking Services



Personal Assistants



Product Recommendations

Overview

General Information

- Meet: Mondays and Wednesdays 11:15-12:30, SN011
- Instructor: Mahmoud Mostapha (*email: mahmoudm@cs.unc.edu*)
  - A PhD candidate, member of NIRAL @ UNC Psych and CS Depts
  - Neuroimaging, Medical Image Analysis, and Machine Learning
  - Study brain development and early disease diagnosis (e.g., Autism)
  - For more information see my Website, Linkedin and Google Scholar
- Office: Wednesdays and Fridays 12:30-1:30, SN228
  - For appointments outside office hours, email me in advance
  - Please include [COMP 562] in the email title
- Prerequisites: Basic Probability/Statistics and Linear Algebra
  Some programming (Python/Matlab)
- Goals: Understanding of basic machine learning algorithms
  Apply the algorithms to a real-world problems
- Web: `http://comp562fall18.web.unc.edu/`

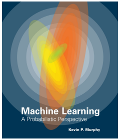
Textbook and Resources

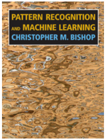
Main Textbook [used for assigned readings]

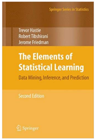
- "Machine Learning: A Probabilistic Perspective," Kevin P. Murphy

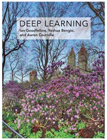
Other Resources

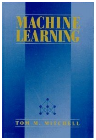
- "Pattern Recognition and Machine Learning," Chris M. Bishop
- "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," T. Hastie, R. Tibshirani, J. Friedman [Download]
- "Deep Learning" I. Goodfellow, Y. Bengio, A. Courville [Download]
- "Machine Learning" Tom M. Mitchell [Download]
- "Machine Learning: The Art and Science of Algorithms that Make Sense of Data" Peter Flach

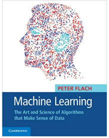












Topics Covered and Class Schedule

Main Topics Covered:

- Linear Models for Classification and Regression (liner regression,...)
- Directed Graphical Models (Bayesian networks, . . .)
- Mixture Models (EM Algorithm, . . .)
- Latent Linear Models (PCA, ICA, . . .)
- Sparse Linear Models (Lasso, Dictionary Learning, . . .)
- Kernel Methods (SVM, KDE, . . .)
- Neural networks and Deep Learning (MLP, CNN, . . .)

Please see class website for updated weekly schedule

Credit and Grades

This course is worth 3 credits. Make sure you registered appropriately.

Please fill in the sign-up sheet posted on course website

Grading:

- Homework: Maximum of 35% (4 assignments each 10%)
  - Groups of two with single submission through Sakai
  - See course website for updated deadlines
- Midterm Exam: 20% (Monday Oct 15)
- Final Exam: 30% (Tuesday Dec 11)
- Final Project: 15% (see project guidelines for full details)

Please participate on Piazza by posting questions and answers

Course Project Guidelines

Objective: apply machine learning algorithms to real-world problems

- In job interviews, often your course projects that end up discussing

Milestones and Deadlines:

- Project Groups (Monday Sep 10)
  - Groups of four students with common interest (e.g., robotics, etc.)
  - May work with a CS faculty as an advisor
- Project Proposal (Wednesday Oct 3 – 10%)
  - 1 page describing the proposed project (template provided)
  - Project Ideas: NIRAL, UNC CS faculty, and other listed resources
- Project Status Report (Monday Nov 12 – 10%)
  - 1-2 page report to track your progress (template provided)
- Project Report and Poster (Wednesday Dec 5 – 80%)
  - 4 page paper in the style of a machine learning conference
  - Project poster for poster session held at Sitterson

Quiz: Which of these are you familiar with?

Raise your hand as I read off the concepts

- Matrix Determinant, Trace, and Rank
- Orthogonal and Orthonormal Vectors
- Eigenvalues and Eigenvectors
- The Gradient and the Hessian
- Random Variables, Multivariate Gaussian Distribution
- Chain Rule and Bayes Rule
- Convex function, Gradient Descent Algorithm

Please check the review material posted on the class website

What is Machine Learning?

Definition by Arthur Samuel (1959)

"Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed."

Definition by Tom Mitchell (1998)

"Machine Learning is the study of algorithms that improve their performance (P) at some task (T) with experience (E)"

A well-defined learning task is given by $< P, T, E >$

Quiz: Defining the Learning Task

"Machine Learning is the study of algorithms that improve their performance (P) at some task (T) with experience (E)"

Q: Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is P, T, and E in this setting?[1]

- Classifying emails as spam or not spam
- Watching you label emails as spam or not spam
- The fraction of emails correctly classified as spam/not spam

[1] Slide credit: Andrew Ng

Traditional Programming Approach – Simple Problem

How to we develop software?

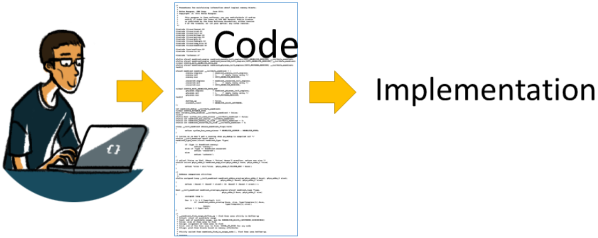
- Specification:
  function f should return a real value that is three times the input
- Implementation:

```
function y = f(x)
    y = 3.0*x;
end
```

- Tests:

```
f(0.0) == 0.0, f(NaN) == NaN, f(1.0) == 3.0,
abs(f(1.000000000001) - 3.000000000003)<1e-12 ...
```

Traditional Programming Approach



Traditional Programming Approach – Another Problem

- Suppose that instead of a specification you get a list of input/output pairs – **the Data**

```
Input (x):  0.2760  0.6797  0.6551  0.1626  ...
Output (y):  0.6147  1.2381  2.3510  0.6959  ...
```

  Generate a program that reproduces behavior captured by these pairs
- Implementation: a lookup table

```
function y = f(x)
    for i=1:length(inputExamples)
        if abs(x-inputExamples(i))<1e-12
            y = outputExamples(i);
            return
        end
    end
    y = NaN;
end
```

- **Q: Is this a good or a bad solution?**

Machine Learning Approach

1. Assume a template for the function (model)

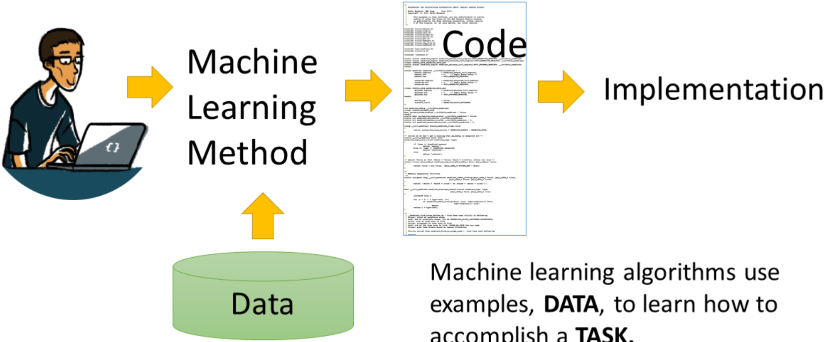
```
function y = f(x,beta)
    y = beta*x;
end
we call beta a parameter
```

2. Specify a cost $C(beta; Data)$ that tells you how well $f(x, beta)$ fits the Data

$$C(beta; Data) = \sum_{(x,y)\in Data} (y - f(x, beta))^2$$

3. Find beta for which cost $C(beta; Data)$ is the smallest, this process is called learning or training

Machine Learning Approach



Machine learning algorithms use examples, **DATA**, to learn how to accomplish a **TASK.**

Machine Learning Approach

- How well does your model perform on new data, data you have not seen during learning? How well your model **"generalize"**?
- If you get a new data instance, for example

```
(No history of cancer, Smoker, Male)
```

Can you assess how good is your throat cancer predictor?

1. Divide data into training set and testing set
2. Use training set as input to learning procedure to produce a predictor
3. Use testing set to evaluate performance of the resulting predictor

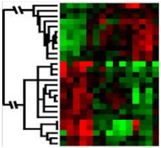
Common mistake to let test data bleed into training data

When Do We Use Machine Learning?

Machine learning is used when

- Human expertise does not exist (navigating on Mars)
- Humans cant explain their expertise (speech recognition)
- Models must be customized (personalized medicine)
- Models are based on huge amounts of data (big data - genomics)
- ......



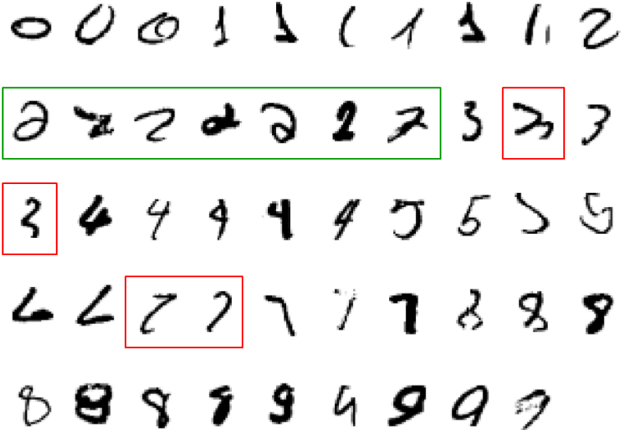



Q: Is machine learning always useful?

- No, for example there is no need to "learn" to calculate payroll

When Do We Use Machine Learning?

Classic example of a task that requires ML:
It is hard to say what makes a 2?[2]



[2] Slide credit: Geoffrey Hinton

Types of Machine Learning

- **Supervised (Predictive) Learning**
  - **Given: training data + desired outputs (labels)**
- **Unsupervised (Descriptive) Learning**
  - **Given: training data (without desired outputs)**
- Semi-supervised learning
  - Given: training data + a few desired outputs
- Reinforcement Learning
  - Rewards from sequence of actions

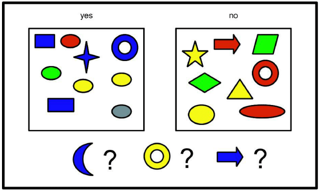
Types of Machine Learning

Learning Type

Output Type	*Supervised Learning*	*Unsupervised Learning*
Discrete	classification or categorization	clustering
Continuous	regression	dimensionality reduction

Supervised Learning: Classification

- Given a training set $D = (\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), ..., (\mathbf{x_N}, y_N)$, where $y \in \{1, \ldots, C\}$, with C being the number of classes



Some labeled training examples of colored shapes, along with 3 unlabeled test cases.

D features (attributes)

N cases

Color	Shape	Size (cm)	Label
Blue	Square	10	1
Red	Ellipse	2.4	1
Red	Ellipse	20.7	0

Representing the training data as an N x D design matrix. Row i represents the feature vector x_i. The last column is the label, $y_i \in \{0, 1\}$ ($C = 2$; binary classification).

- Estimate function f given D, then predict using $\hat{y} = \hat{f}(\mathbf{x})$

Q: What label should we assign to the yellow cirlce in the test data?

Machine Learning - Probabilistic View

The need for probabilistic predictions

- To handle ambiguous cases, it is desirable to return a probability
- Denote the probability distribution given $\mathbf{x}$ and and D by $P(y|\mathbf{x}, D)$
- For example, for the yellow circle in last example:

$$P(y = 0|\mathbf{x}, D) = 0.6 \text{ and } P(y = 1|\mathbf{x}, D) = 0.4$$

- Given a probabilistic output, we can always compute our best guess as to the "true label" using:

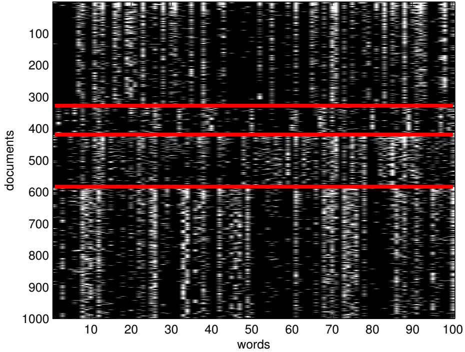
$$\hat{y} = \hat{f}(x) = \underset{c \in C}{\operatorname{argmax}} P(y = c|\mathbf{x}, D)$$

- This is also known as a MAP (maximum a posteriori) estimate
- Continuous predictions will enable feedback on how to adjust them
- This will enable use of optimization and to quantify uncertainty

Supervised learning: Classification Examples

Email spam filtering

- The goal is to classify email message into spam $y = 1$ or $y = 0$
- Bag of words representation for variable-length documents

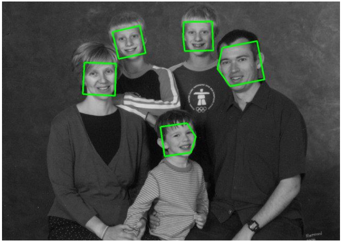


Each row is a document represented as a bag-of-words bit vector. There are subsets of words whose presence or absence is indicative of the class.

Supervised learning: Classification Examples

Face detection and recognition

- Find objects (faces) within an image is called object (face) detection
  - Sliding window detector is an example for solving this problem
- Then, face recognition can be used to estimate the person identity
  - Features used are likely to be different than in face detection problem

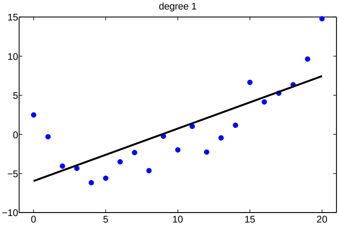


Face detector detected five faces at different poses.

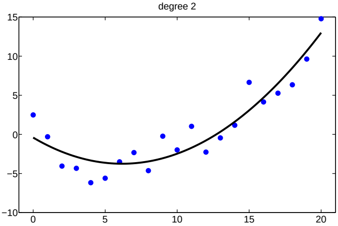
Face database used to train a face recognition classifier.

Supervised Learning: Regression

- Regression like classification except the response variable is continuous
- For a single input $x_i \in \mathbb{R}$, and a single response $y_i \in \mathbb{R}$



Linear regression on some 1d data



Same data with polynomial regression

Some examples of real-world regression problems

- Predict tomorrow's stock market price given current market conditions
- Predict the age of a viewer watching a given video on YouTube
- Predict the location of a robot arm given control signals

Quiz: Classification Vs. Regression

You're running a company, and you want to develop learning algorithms to address each of two problems[3]

Problem 1: You have a large inventory of identical items. You want to predict how many of these items will sell over the next 3 months

Problem 2: Youd like software to examine individual customer accounts, and for each account decide if it has been hacked/compromised

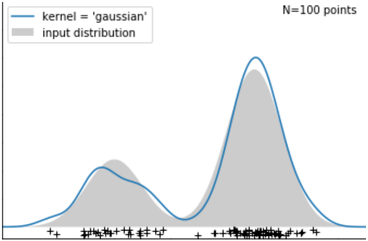
Q: Should we treat these as classification or as regression problems?

1. Both as classification problems
2. Problem 1 as classification problem, problem 2 as regression problem
3. Problem 1 as regression problem, problem 2 as classification problem
4. Both as regression problems

[3] Slide credit: Andrew Ng

Unsupervised learning: Clustering

- Given data $\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N}$ (without Labels), and goal to estimate
$$P(\mathbf{x_i}|\theta) \text{ instead of } P(y_i|\mathbf{x_i}, \theta) \text{ in supervised Learning}$$
- In clustering, estimate $P(K|D)$, where K denote is number of clusters

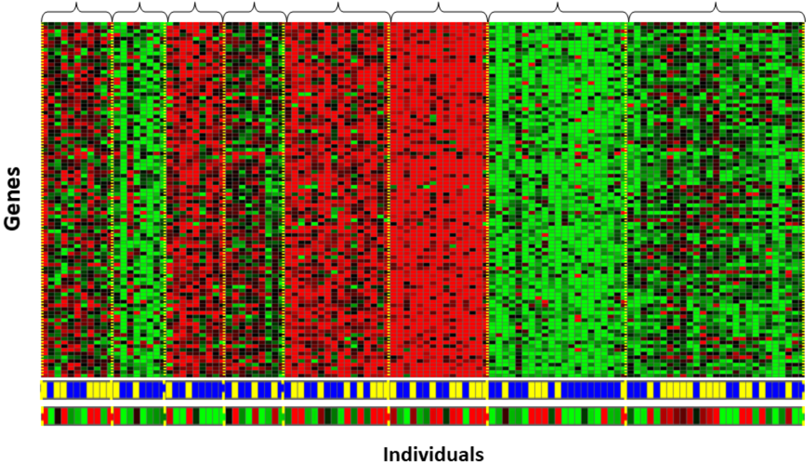


Density estimation using a Gaussian mixture.

Q: What is K here? How would you perform group assignments?

Unsupervised learning: Clustering Examples

Genomics application: group individuals by genetic similarity[4]

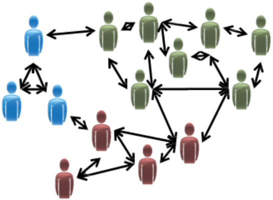


[4] Slide credit: Daphne Koller

Unsupervised learning: Clustering Examples

Additional real-world clustering application[5]

Organize computing clusters



Social network analysis

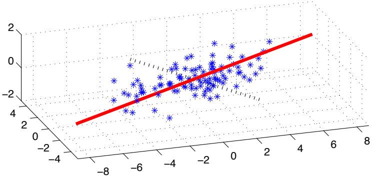
Market segmentation

Astronomical data analysis

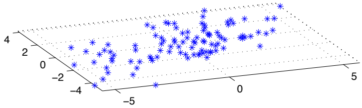
[5] Slide credit: Andrew Ng

Unsupervised learning: Dimensionality Reduction

- Data may appear high dimensional, there may only be a small number of degrees of variability, corresponding to **latent factors**
- When feature reduction is performed by selection a subset of the original features, this is called **feature selection**
- In **feature extraction**, dimensionality is reduced by projecting data to a lower dimensional subspace that captures the "essence" of the data
  - Principal components analysis (PCA) is a commonly used approach



A set of points that live on a 2d linear subspace embedded in 3d.



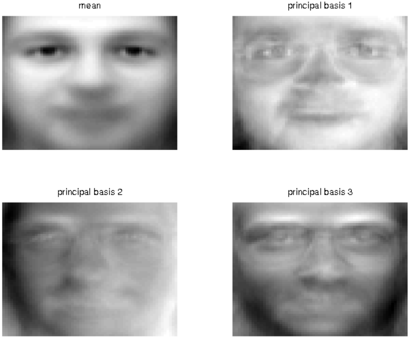
2D representation of the data.

Unsupervised learning: Dimensionality Reduction Example

EignFaces: Modeling the appearance of face images

- Only few underlying latent factors can describe most of the variability
  - e.g., lighting, pose, identity, etc

25 randomly chosen pixel images from the face database.



The mean and the first three principal component basis vectors (eigenfaces).

Quiz: Unsupervised Learning

Q: Of the following examples, which would you address using an unsupervised learning algorithm? (Choose all that apply.)[6]

1. Given email labeled as spam/not spam, learn a spam filter
2. Given a set of news articles found on the web, group them into set of articles about the same story
3. Given a database of customer data, automatically discover market segments and group customers into different market segments
4. Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not

[6] Slide credit: Andrew Ng

Reinforcement Learning

- Given a sequence of states and actions with rewards, output a policy[7]
  - Policy is a mapping from states → actions telling you what to do

Example: Infinite Mario AI Reinforcement Learning (Youtube)

[7]Slide credit: Eric Eaton

Inverse Reinforcement Learning

- Given some agent's policy or a history of behavior and we try to find a reward function that explains the given behavior

Example: Stanford Autonomous Helicopter (Project) (Youtube)

Today

Recap:

- COMP 562 is not easy, check out the review material
- Machine Learning as a different way of developing programs
- Supervised learning discovers a mapping from features to labels
- Unsupervised learning finds compact representations of the data
- Reinforcement learning learns actions that maximizes reward