CS 383 – Machine Learning

Introduction to Machine Learning

Slides adapted from material created by E. Alpaydin
Prof. Mordohai, Prof. Greenstadt, Pattern Classification (2nd Ed.),
Pattern Recognition and Machine Learning

Objectives

- Understand common machine learning problems
- Understand basic ML terminology

What is Machine Learning?

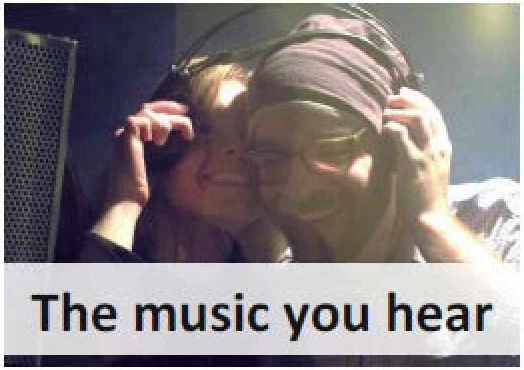
- Definition: "The study of computer algorithms that improve automatically through experience"
- Formally:
  - Improve at task T
  - With respect to performance measure P
  - Based on experience E
- Example: Recognize a Person
  - T: recognize a person
  - P: number of time we recognized a person correctly
  - E: a database of labeled faces

ML vs AI

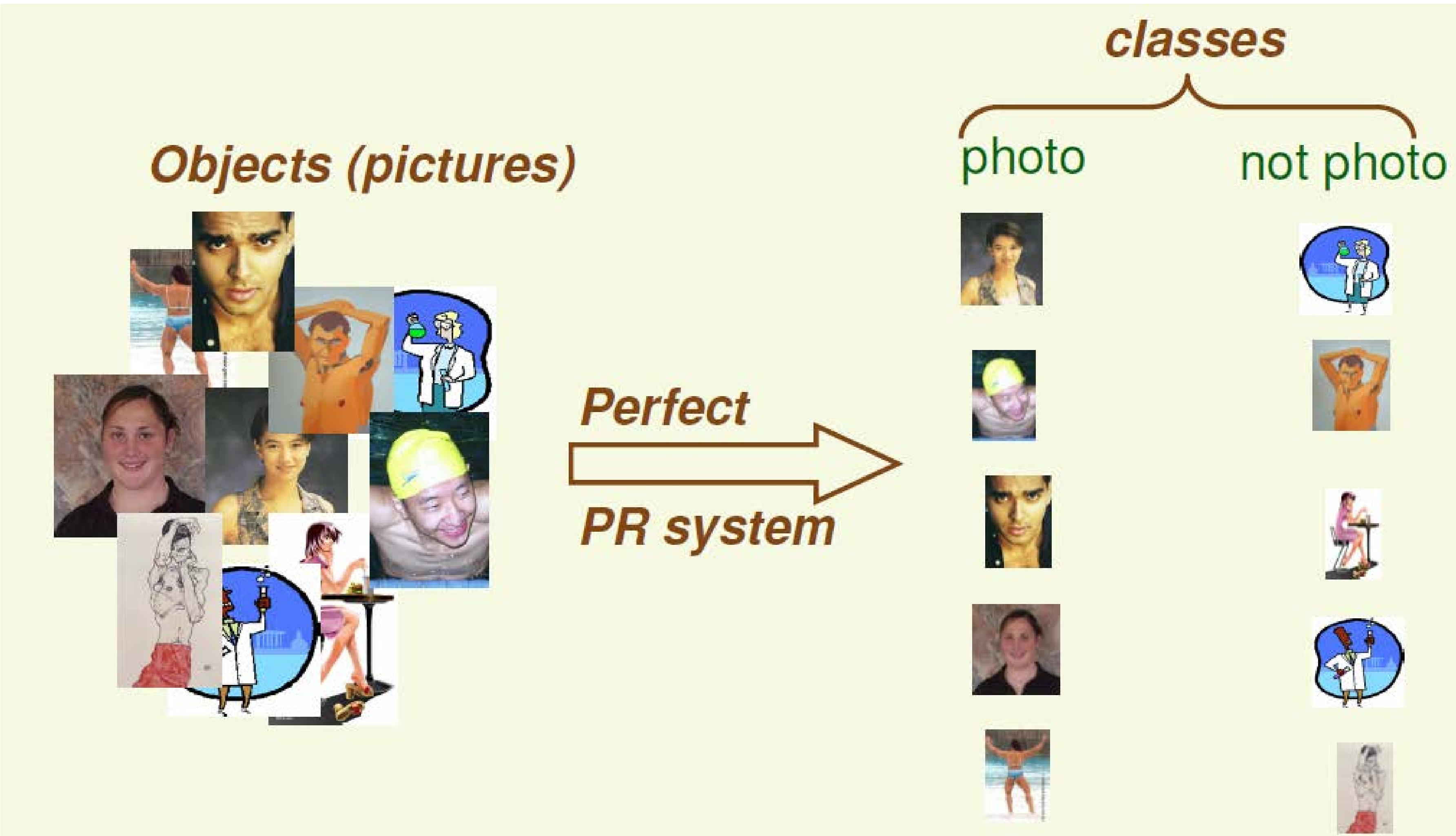
- How is this different than AI?
  - ML can be thought of as a sub-topic within AI
- AI deals with any “intelligent” task performed by a non-human agent
  - Often “path finding algorithms”
- ML specifically deals with making decisions based on acquired data
  - Both past and current

Why do we care?

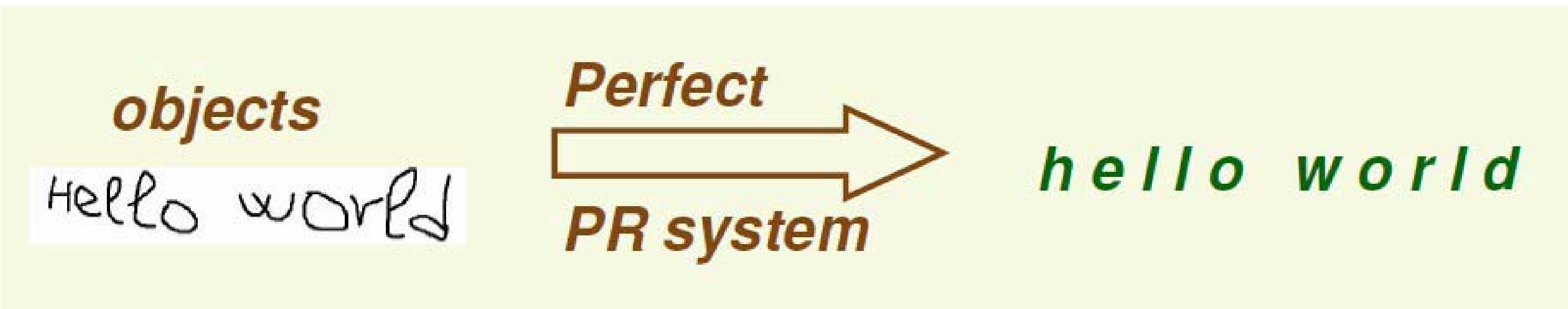
- It's everywhere!!!



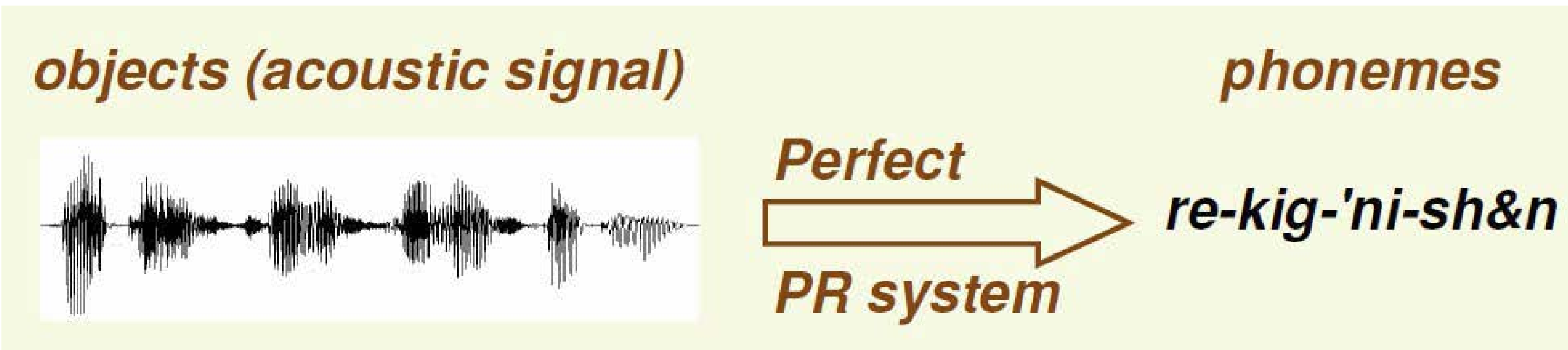
Example: Photograph or Not?



Example: Character Recognition



Example: Speech Understanding

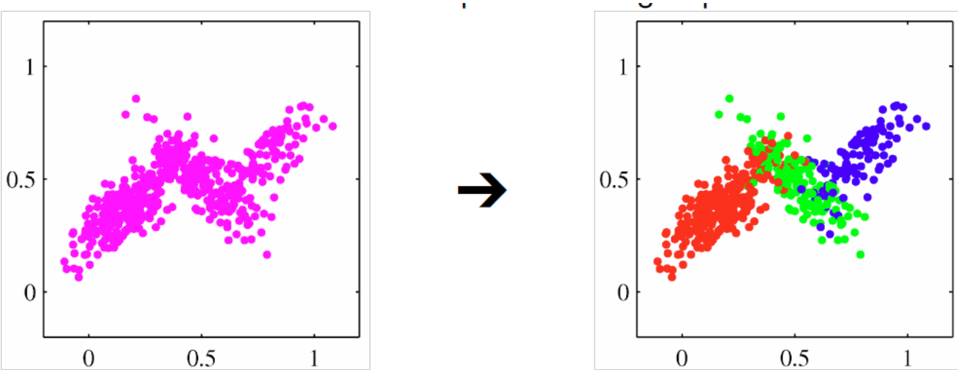


How Do We Do It?

- We need to think about
  1. What needs to be learned?
     - What's our task/goal?
  2. What feedback can we get and in what form?
     - Supervised learning (correct answers for each example)
     - Unsupervised learning (correct answers not given)
  3. What representation should we use (features)?

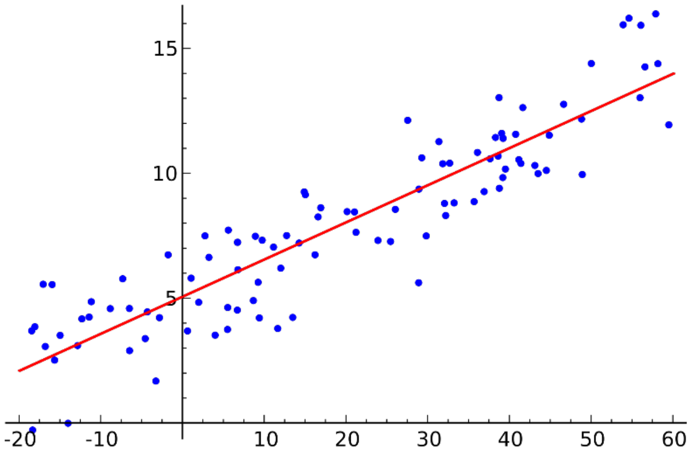
Problems in ML

- There are several typically machine learning problems that we'll tackle in the course
- Clustering
  - Given some data we want to figure out how to group them together.
  - This is an example of *unsupervised learning* where we're trying to discover the groups (instead of being given them)



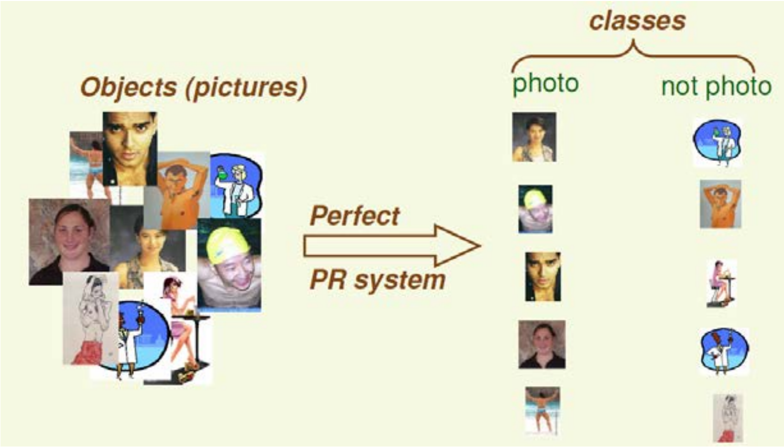
Problems in ML

- Regression
  - Given some data, can we predict an outcome value?
  - Example: We have a car's brand, year, mpgs and want to figure out its worth
  - This is an example of *supervised* learning
    - To build our prediction system, we have data with labels.



Problems in ML

- Classification
  - Given data, can we predict which category something belongs to.
  - Typically involves *learning* some rules.
  - This is an example of *supervised* learning
    - To build our prediction system, we have data with labels.



Administrative Stuff…

Contacts

- Instructor:
  - Matt Burlick, mjburlick@drexel.edu, UC137
  - Office hours
    - Wednesday 3:00pm – 5:00pm
    - Thursdays 3:30pm-5:30pm
    - And by appointment
- Teacher Assistant:
  - Janith Weerasignhe, bnw37@drexel.edu, CLC
    - Thursdays 2:00pm-4:00pm
- Lectures:
  - Section 001 TR 09:30pm-10:50am Rush 014
  - Section 002 TR 11:00am-12:20pm Rush 014

Pre-Requisites

- CS 260 (Data Structures
- CS 380 (Artificial Intelligence)
- The idea is that you should be a proficient programmer such that you can pick up a new language "on the fly" and use it as a tool.
- You should also be comfortable with linear algebra, probability, statistics, and calculus.
  - We'll have a little (ungraded) quiz at the end of today's lecture so you'll see what you're expected to know

Course Resources

- Official Textbook:
  - None
- Recommended Textbooks:
  - Basic: An Introduction to Statistical Learning (free PDF)
  - Medium: Introduction to Machine Learning (Alpaydin)
  - Advanced: Machine Learning (Murphy)
- Blackboard
- Piazza

Course Software

- Programming Environment
  - Your choice
  - I recommend MatLab (programming environment)
    - Obtain for free from http://drexel.edu/irt/computers-software/software/
- Typesetting Environment
  - LaTex or MS Equation Editor
    - First download LatTex itself (warning, it's huge!)
      - https://latex-project.org/ftp.html
    - Then (optionally) get a IDE wrapper for it
      - http://www.xm1math.net/texmaker/download.html#window
      - https://github.com/TeXworks/texworks/releases
    - Or use an online Latex typesetter!
      - www.overleaf.com
- Discussion Forum
  - Piazza – You should have already received an invitation!
  - Use this as your first place to pose questions
    - Hopefully not just I can help
  - But don't post code.

Course Objectives

- Foundations of modern statistical Machine Learning
  - Regression (maximum likelihood)
  - Probabilities, Bayesian modeling and inference
  - Classification (support vector machines, etc..)
  - Mixture models (k-means, Gaussian mixtures)
  - Sequential data (Hidden Markov models)
- Applications of Machine Learning algorithms
- Implementation and use of Machine Learning algorithms

Evaluation

• Homework Sets	55%
• Quizzes	15%
• Final Exam	30%

Homework Sets

- Eight distributed throughout the course
- Reinforce your understanding of the material
- Include theoretical questions as well as implementation of ML algorithms
- Submission is to be made on Blackboard as a single compressed file consisting of:
    - PDF with your solutions to the theory questions
        - Must be typeset with either Microsoft Equation Editor or Tex
    - README text file on how to run your code
    - Source code
- You will lose 1pt for every hour late (round up) on an assignment up to 48hrs (after which you will receive a zero)

Midterm/Final

- Based on theory and mathematics.
- Similar to theory questions that are part of assignments.
- Midterm has significantly less weight so you can get a feel for what to expect prior to the final.

Course Policies

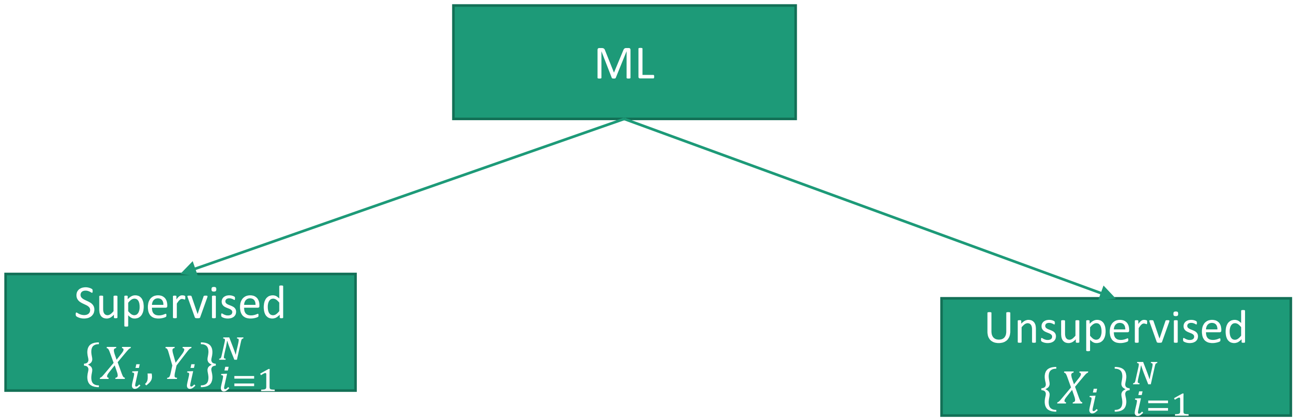
- Assignments are to be done individually unless otherwise noted
- While you are encouraged to use a versioning system like github or bitbucket, please make your work for this course **private.**
- Any dispute about an assignment grade must be made and resolved within 5 days of receiving your grade. After this period your grade cannot be adjusted.

Notation/Mathematics/Matlab

- I have placed on Blackboard a number of resources to help you review the expected and needed math as well as get you started in Matlab:
  - Course Notation – There will be a lot of symbols used in this course. This document tries to give you an overview of them.
  - Similarity and Distance Functions – Often we will need to compute the distance and/or similarity between observations. This document includes several commonly used ones.
  - Matlab Crash Course – For those of you who haven't used Matlab before, here's a crash course I gave a while back.
  - Matlab Functions – Here's a list of most of the Matlab functions I used in developing this course.
  - Math Review – A quick review of the most critical math needed for this course. Including..
    - Calculus
    - Linear Algebra
    - Probability and Statistics

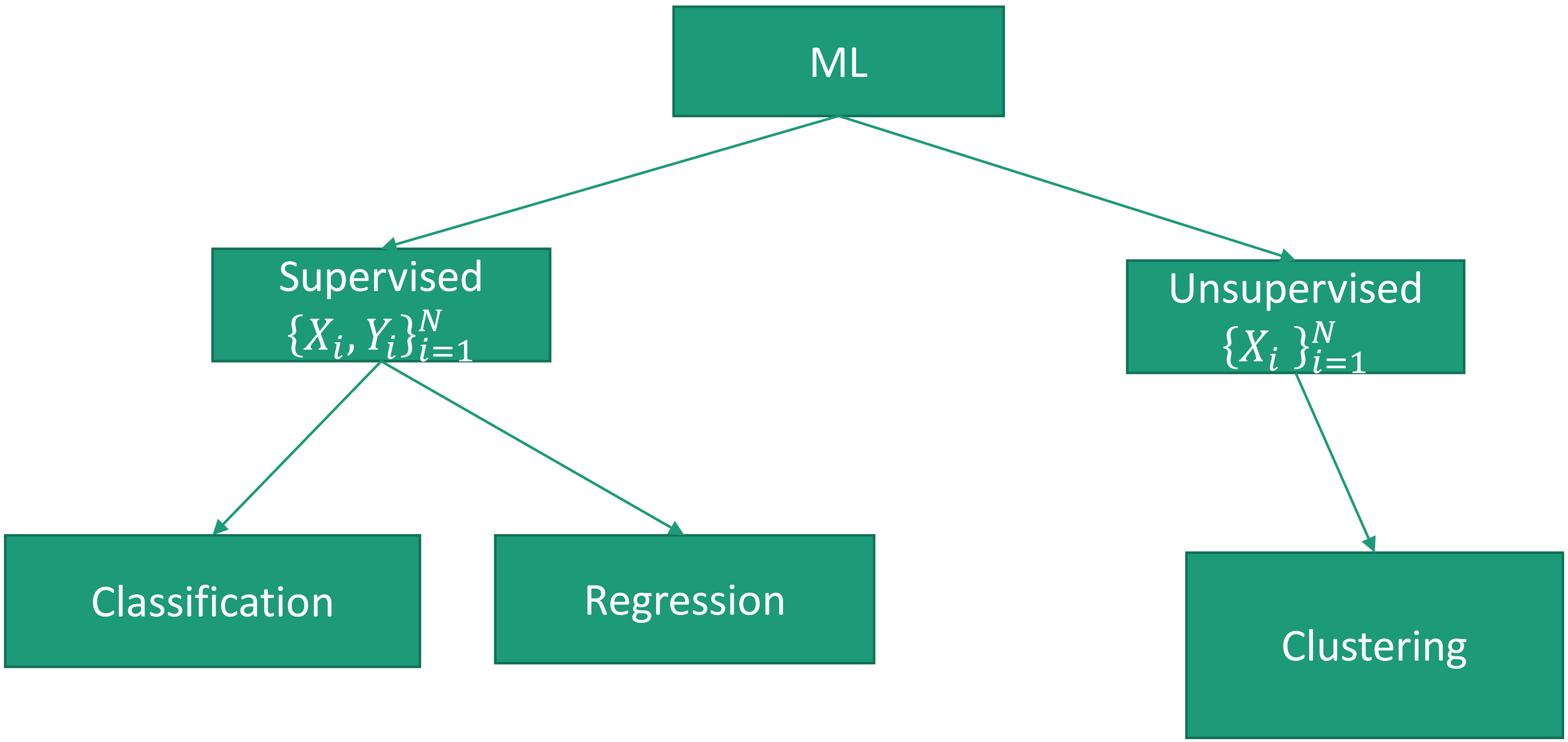
Fundamental ML Concepts

ML Overview

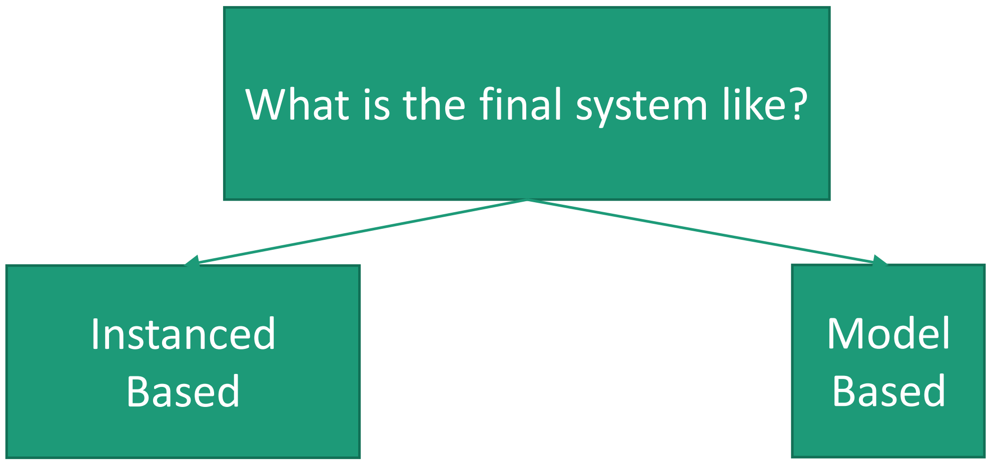


- We can basically break machine learning tasks into two categories
  1. Supervised Learning
  2. Unsupervised Learning
- *Supervised learning*
  - Data X_i and correct answer (label) Y_i given for each example $i \in \{1, \dots, N\}$
- *Unsupervised learning*
  - Only data given for each example, X_i

Types of Problems

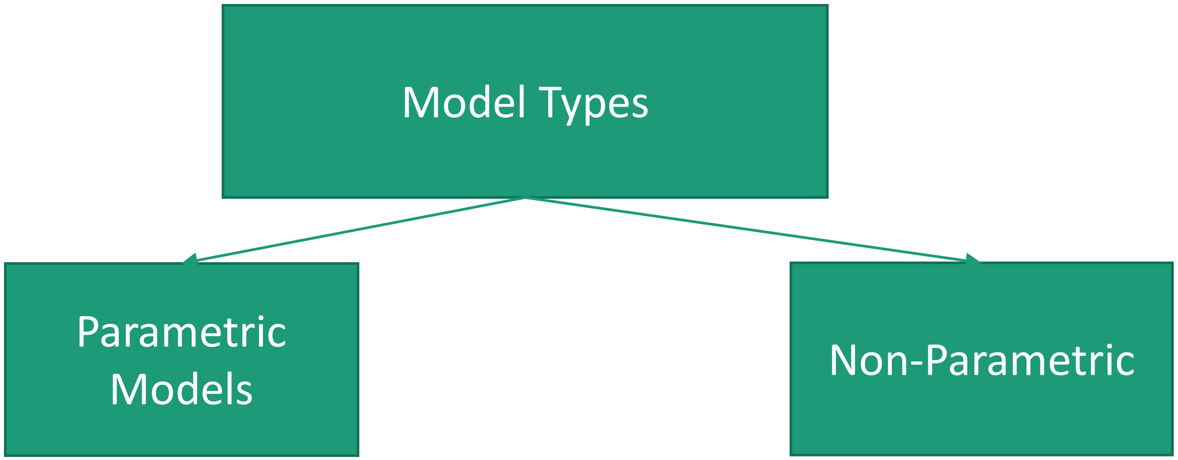


Types of Algorithms



- *Instance Based Algorithms*
  After training, performing the desired task (classification, regression, clustering, etc..) is based on training data samples themselves
- *Model Based Algorithms*
  Training involves learning a model such that performing the desired task is done by running the data through model (which does not directly reference training data samples)

Types of Algorithms



- Algorithms can further be defined as *parametric* vs non-parametric
- *Parametric Algorithms*
  - The model we're learning is based on blending information via parameters. These parameters must be learned during the training process.
- Non-Parametric Algorithms
  - These algorithms/models are not defined by determining blending parameters.
  - However there still may be use-specified parameters

Types of Algorithms

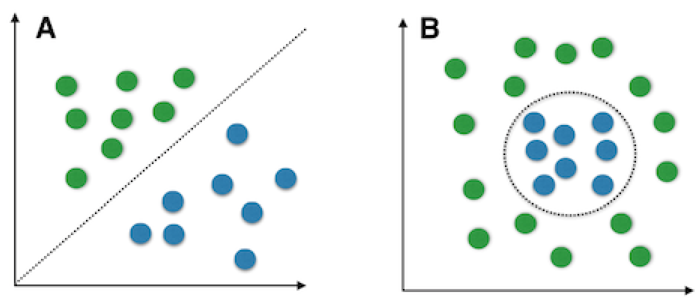
Division of Hypothesis Space

Linear

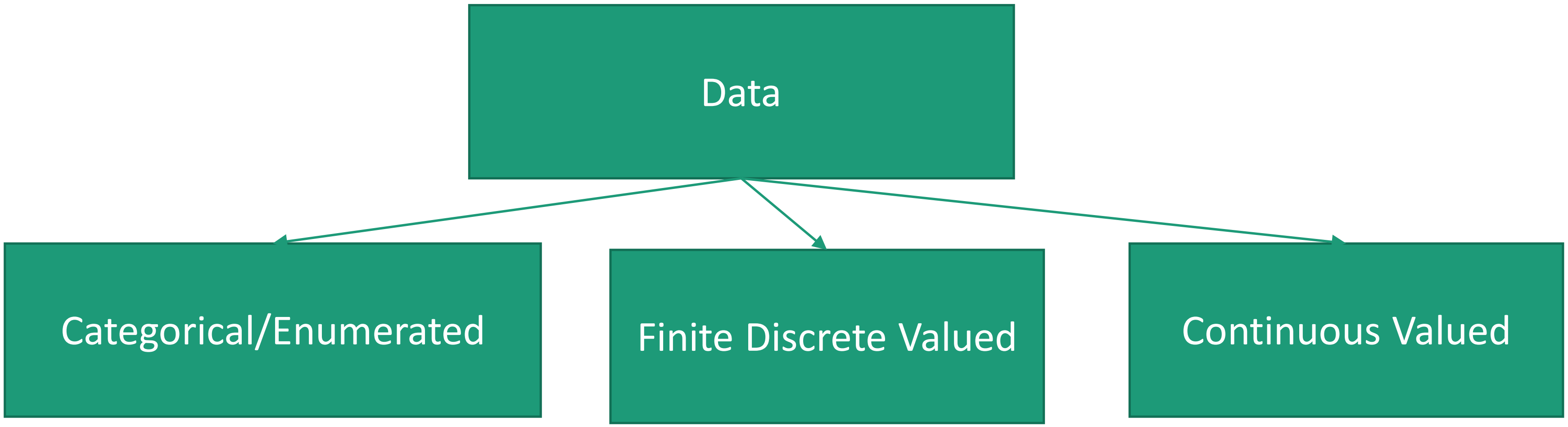
Non-Linear

- A large part of machine learning is attempting to divide the hypothesis space
  - Separate it into clusters
  - Separate it into classes
- These “dividers” can be either
  - Linear
  - Non-Linear

Linear vs. nonlinear problems



Types of Data



- Our data can also be categorical or quantitative
- *Categorical*
  - Examples: Car Model, School
- *Finite Discrete Valued*
  - Ordering still matters, but there's only so many values out there
- *Continuous Valued*
  - Examples: Blood Pressure, Height

No Free Lunch Theorem

- Unfortunately there's no single machine learning algorithm to rule them all ☹
- Typically we must consider the previous information in deciding which set of algorithms to train and test
    - Then we select the best.

No Free Lunch Theorem

- Questions we should ask ourselves are:
  - Do we have labels?
  - Do we want to do clustering?
  - Do we know the number of clusters?
  - Do we want to do regression?
  - Do we want to do classification?
  - Is understanding/interpreting the learned system important?
  - What are our memory and/or processing time limitations?
  - Do we want instance-based learning?
  - Do we have categorical or quantitative data?
  - Do I have enough data?
  - Do I have enough features? Too many?
  - Is linear ok or do I need non-linear?
  - And more…

CS383 ML Algorithms

- Here’s a list of algorithms we’ll look at in the class and what types they are

1. Feature Reduction (Feature Selection, Feature Projection)
2. Clustering
   a. Expectation-Maximization (k-means, k-medoids, GMM)
   b. Agglomerative Clustering
3. Linear Regression
4. Classification
   a. Probabilistic Decisions (Inference, Bayesian Learning, Decision Trees)
   b. Nearest Neighbors
   c. Support Vector Machines
   d. Logistic Regression
   e. Multi-layer Perceptions (artificial neural networks, deep learning)
   f. Hidden Markov Models (HMMs)