Introduction to Machine Learning: CS 436/580L

Inductive (Supervised) Learning: Hypothesis Spaces

Instructor: Arti Ramesh

Binghamton University

Administrivia

- Homework 0 is available on myCourses
  - Worth 4/40 points
  - Due Sep 6th, Wednesday, 11:59 pm
  - Late penalty is 10% after the deadline

Probability Review Recap

- Mutual Exclusion
- Independence
- Conditional Independence
- Expectation
- Variance

Probability Review

The weather on a particular day can be sunny, cloudy, or rainy. It can be sunny with probability = 0.3, cloudy with probability = 0.4, and rainy with probability = 0.3. A concert is planned to be held in the city. If the weather is sunny, the concert will be held 100%. If the weather is cloudy or rainy, it will be held with probability 0.8 and 0.5, respectively.

What is the probability that the concert will be held?

Probability Review

Let X denote the sum of two fair dice. What is the expectation of X?

Recap

- Different definitions of machine learning and all are correct!!!
- Slight variations according to type of learning
- Types of Learning
  - Supervised Learning
  - Unsupervised Learning
  - Reinforcement Learning
  - Semi-Supervised Learning

Types of Learning

- **Supervised Learning**
  - **problem**: the learner is required to learn a **function** which maps a vector into one of several classes by looking at several input-output examples of the function.
  - standard formulation of the supervised learning task: **classification**

Types of Learning

- **Unsupervised Learning**
  - models a set of inputs: labeled examples are not available
  - standard formulation of the unsupervised learning task: clustering
- **Semi-supervised Learning**
  - combines both labeled and unlabeled examples to generate an appropriate function or classifier

Types of Learning

- **Reinforcement Learning**
  - the algorithm learns a policy of how to act given an observation of the world
  - Every action has some impact in the environment, the environment provides feedback that guides the learning algorithm

Which type of learning is best?

- Determining the best move to make in a game
- Distinguish between dogs, cats, and horse pictures
- Elevator scheduling
- Agent in field trying to diffuse a bomb
- Speech analysis of telephone conversation (400 hours annotation time for each hour of speech)

ML in a Nutshell

- Tens of thousands of machine learning algorithms
- Hundreds new every year
- Every machine learning algorithm has three components:
  - **Representation**
  - **Evaluation**
  - **Optimization**

Representation

- Decision trees
- Sets of rules / Logic programs
- Instances
- Graphical models (Bayes/Markov nets)
- Neural networks
- Support vector machines
- Model ensembles
- Etc.

Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- Etc.

Optimization

- Combinatorial optimization
  - E.g.: Greedy search
- Convex optimization
  - E.g.: Gradient descent
- Constrained optimization
  - E.g.: Linear programming

Supervised Learning

- **Given**: Training examples ***(x, f(x))***, for some unknown function *f*
- **Find**: an approximation to *f*

Example Applications

- **Credit risk assessments**
  - *x*: properties of customer and proposed purchase
  - *f(x)*: to approve/reject purchase
- **Disease diagnosis**
  - *x*: properties of patient (symptoms, lab tests)
  - *f(x)*: disease diagnosis, recommended therapy
- **Face recognition**
  - *x*: bitmap picture of person's face
  - *f(x)*: Person's name

Appropriate Applications for Supervised Learning

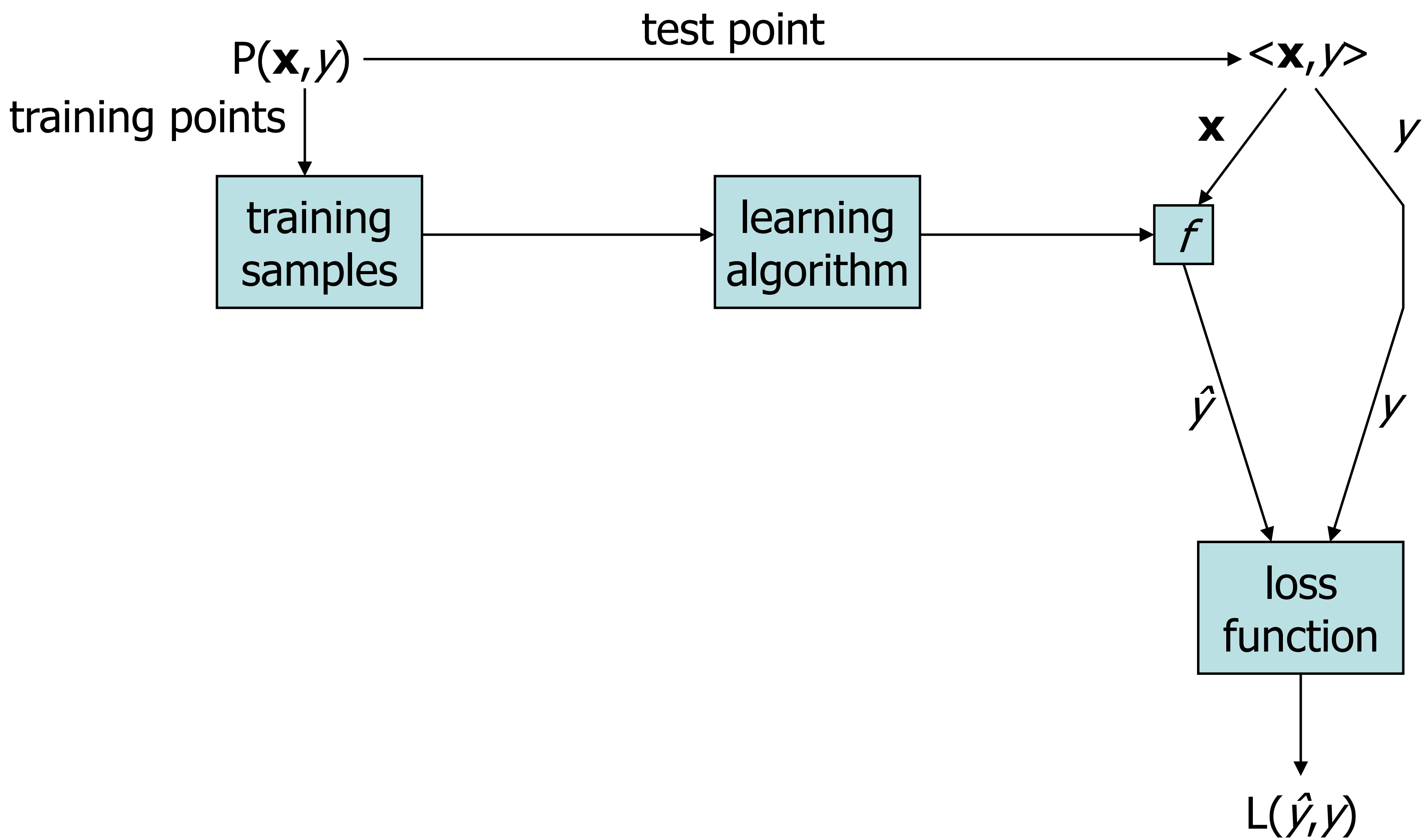
- **Situations where there is no human expert**
  - *x*: bond graph for a new molecule
  - *f(x)*: predicted binding strength to AIDS protease molecule
- **Situations where humans can perform the task but cant describe how to do it**
  - *x*: bitmap picture of handwritten character
  - *f(x)*: ASCII code of character
- **Situations where desired f(x) is changing rapidly**
  - *x*: description of stock prices and trades for last 10 days
  - *f(x)*: recommended stock transactions
- **Situations where each user needs a customized f**
  - *x*: incoming email message
  - *f(x)*: importance score for presenting to user

Example: A dataset for supervised learning

Sepal Length	Sepal Width	Petal Length	Petal Width	Class
5.1	3.5	1.4	0.2	Iris-Sentosa
6.1	3.0	4.6	1.4	Iris-Versicolor
7.2	3.6	6.1	2.5	Iris-Virginica

- Columns are called **input variables**, **features**, or **attributes**
- The type of flower {Iris-Sentosa, Iris-Versicolor, Iris-Virginica} are called **target variables, output variables,** or **labels**
- A row in the table is called a **training example**
- The whole table is called **(training, validation, test or evaluation) data set**
- The problem of predicting the label is called **classification**

Supervised Learning: Formal Definition



A learning problem!

X	0	X
0	X	0
0	X	X

X	0	X
X	X	0
X	0	0

X	X	X
0	X	X
0	0	0

f(x)=1

0	X	0
X	0	X
0	X	X

0	0	X
X	X	0
0	X	X

0	X	X
X	0	0
0	X	X

f(x)=0

0	X	X
0	X	0
X	X	0

f(x)=?

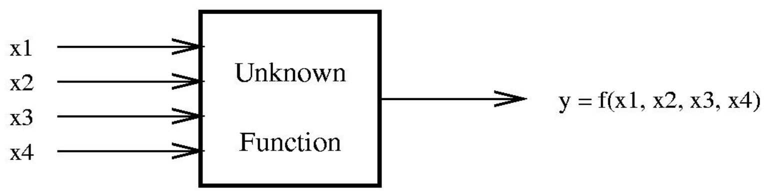
A Learning Problem!

X1	X2	X3	X4	X5	X6	X7	X8	X9	f(x)
X	0	X	0	X	0	0	X	X	1
X	0	X	X	X	0	X	0	0	1
X	X	X	0	X	X	0	0	0	1
0	X	0	X	0	X	0	X	X	0
0	0	X	X	X	0	0	X	X	0
0	X	X	X	0	0	0	X	X	0
0	X	X	0	X	0	X	X	0	?

- x: a 9-dimensional vector
- f(x): a function or a program that takes the vector as input and outputs either a 0 or a 1
- **Task**: given the training examples, find a good approximation to f so that in future if you see an unseen vector "x" you will be able to figure out the value of f(x)

Example of a learning problem

A Learning Problem



A simpler example for analysis!

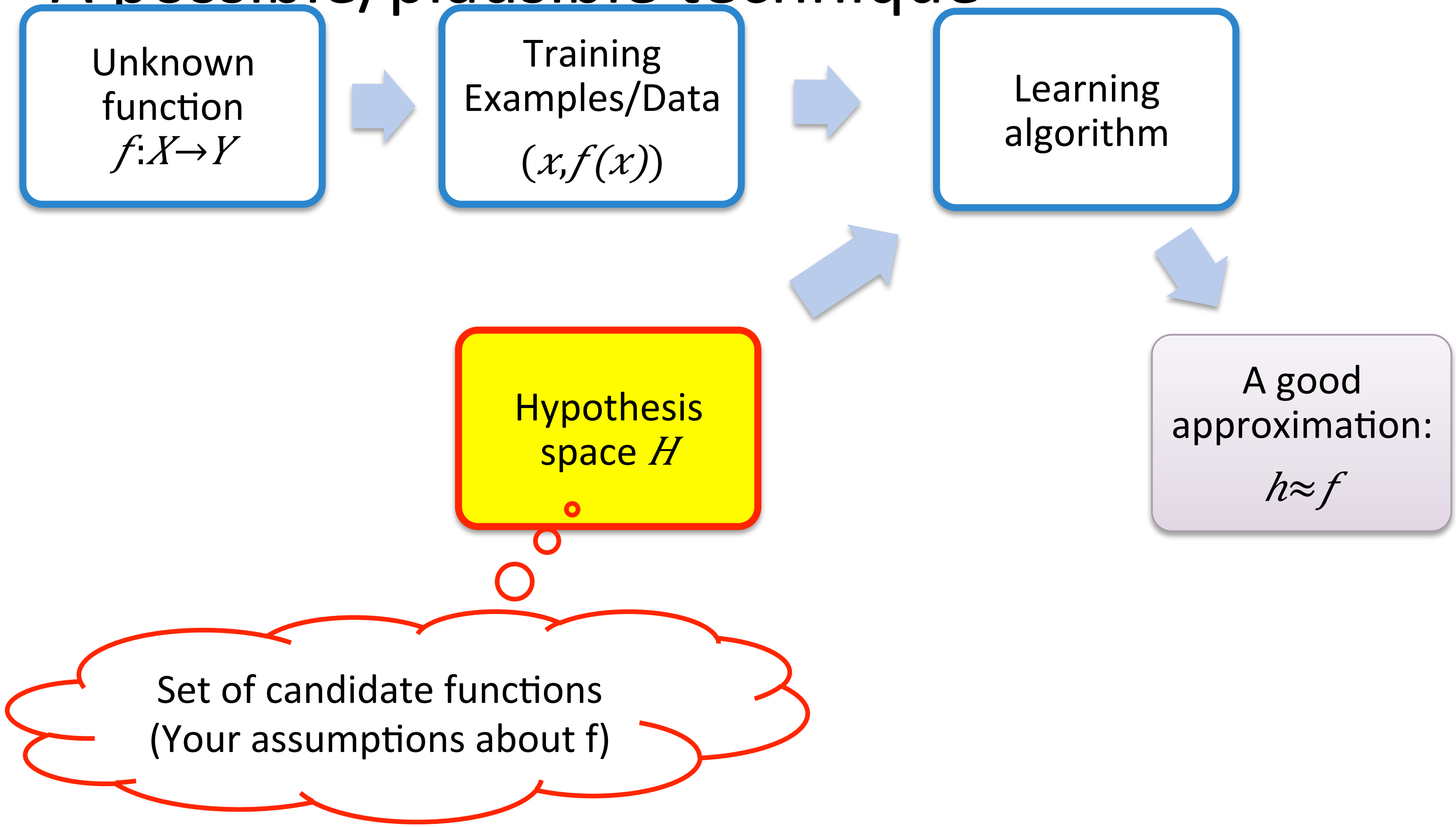
Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

Classification problem

Given data or examples, find the function f?

How to find a good approximation to f?

- A possible/plausible technique



Hypothesis Spaces

- **Complete Ignorance.** There are $2^{16} = 65536$ possible boolean functions over four input features. We can't figure out which one is correct until we've seen every possible input-output pair. After 7 examples, we still have 2^9 possibilities.

You are assuming that the unknown function f could be any one of the 2↑16 functions!

x_1	x_2	x_3	x_4	y
0	0	0	0	?
0	0	0	1	?
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	?
1	0	0	0	?
1	0	0	1	1
1	0	1	0	?
1	0	1	1	?
1	1	0	0	0
1	1	0	1	?
1	1	1	0	?
1	1	1	1	?

It turns out that out of the 2↑16 possible functions, 2↑9 classify all points in the training data correctly!

Hypothesis Spaces

- 10,000 features
- Features are binary
- Output is binary

Number of boolean functions?

Hypothesis Spaces

- **Simple Rules.** There are only 16 simple conjunctive rules.

You are assuming that the unknown function f could be any one of the 16 conjunctive rules!

Rule	Counterexample
$\Rightarrow y$	1
$x_1 \Rightarrow y$	3
$x_2 \Rightarrow y$	2
$x_3 \Rightarrow y$	1
$x_4 \Rightarrow y$	7
$x_1 \wedge x_2 \Rightarrow y$	3
$x_1 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \Rightarrow y$	3
$x_2 \wedge x_4 \Rightarrow y$	3
$x_3 \wedge x_4 \Rightarrow y$	4
$x_1 \wedge x_2 \wedge x_3 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3
$x_1 \wedge x_2 \wedge x_3 \wedge x_4 \Rightarrow y$	3

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

Unfortunately, none of them work

No simple rule explains the data. The same is true for simple clauses.

Hypothesis Spaces

- **m-of-n rules**. There are 32 possible rules (includes simple conjunctions and clauses).

At least *m* of the *n* variables must be true

You are assuming that the unknown function f could be any one of the 32 m-of-n rules!

Only one of them, the one marked by "***" works!

	Counterexample			
variables	1-of	2-of	3-of	4-of
$\{x_1\}$	3	–	–	–
$\{x_2\}$	2	–	–	–
$\{x_3\}$	1	–	–	–
$\{x_4\}$	7	–	–	–
$\{x_1, x_2\}$	3	3	–	–
$\{x_1, x_3\}$	4	3	–	–
$\{x_1, x_4\}$	6	3	–	–
$\{x_2, x_3\}$	2	3	–	–
$\{x_2, x_4\}$	2	3	–	–
$\{x_3, x_4\}$	4	4	–	–
$\{x_1, x_2, x_3\}$	1	3	3	–
$\{x_1, x_2, x_4\}$	2	3	3	–
$\{x_1, x_3, x_4\}$	1	***	3	–
$\{x_2, x_3, x_4\}$	1	5	3	–
$\{x_1, x_2, x_3, x_4\}$	1	5	3	3

Example	x_1	x_2	x_3	x_4	y
1	0	0	1	0	0
2	0	1	0	0	0
3	0	0	1	1	1
4	1	0	0	1	1
5	0	1	1	0	0
6	1	1	0	0	0
7	0	1	0	1	0

Two Views of Learning

- Learning is the removal of uncertainty
- Learning requires guessing a good, small hypothesis case
- We could be wrong!
  - Our prior knowledge might be wrong!
  - Our guess for hypothesis class could be wrong!
    - The smaller the hypothesis class, more likely we are wrong!

Example: $x_4 \wedge Oneof\{x_1, x_3\} \Rightarrow y$ is also consistent with the training data.

Example: $x_4 \wedge \neg x_2 \Rightarrow y$ is also consistent with the training data.

If either of these is the unknown function, then we will make errors when we are given new x values.

Strategies for Machine Learning

- **Strategy 1:** Develop languages for expressing prior knowledge: rule grammars and stochastic models
- **Strategy 2:** Develop flexible hypothesis spaces: Nested collections of hypotheses – decision trees, rules, neural networks, …
- In either case:
  - **Develop algorithms for finding a hypothesis that fits the data!**

Terminology

- Training Example: An example of form ***(x, f(x))***
- Target function (target concept): The true function f
- Hypothesis: A proposed function h believed to be similar to ***f***
- Concept: A boolean function. Examples for which ***f(x) = 1*** are called positive examples or positive instances of the concept. Examples for which ***f(x) = 0*** are called negative examples or negative instances of the concept.
- Classifier: A discrete-valued function. The possible values of ***f*** are called class labels $f \in \{1, 2, \ldots K\}$
- Hypothesis Space: The space of learning algorithms that can be output by a learning algorithm

Key Issues in Machine Learning

- **What are good hypothesis spaces?**
  - Which spaces are useful in practical applications and why?
- **What algorithms can work in these spaces?**
  - Are there general design principles for machine learning algorithms?
- **How can we optimize accuracy on future data points?**
  - This is sometimes called the problem of overfitting
- **How can we have confidence in the results?**
  - How much training data is required to find accurate hypothesis
- **Are some learning problems computationally intractable?**

  (the computational question!)
- **How can we formulate application problems as machine learning problems?**

  (the engineering question!)

Steps in Supervised Learning

1. Determine the representation for ***"x,f(x)"*** and determine what ***"x"*** to use **(Feature Engineering)**
2. Gather a training set (not all data is kosher) **(Data Cleaning)**
3. Select a suitable evaluation method
4. Find a suitable learning algorithm among a plethora of available choices
   – Issues discussed on the previous slide

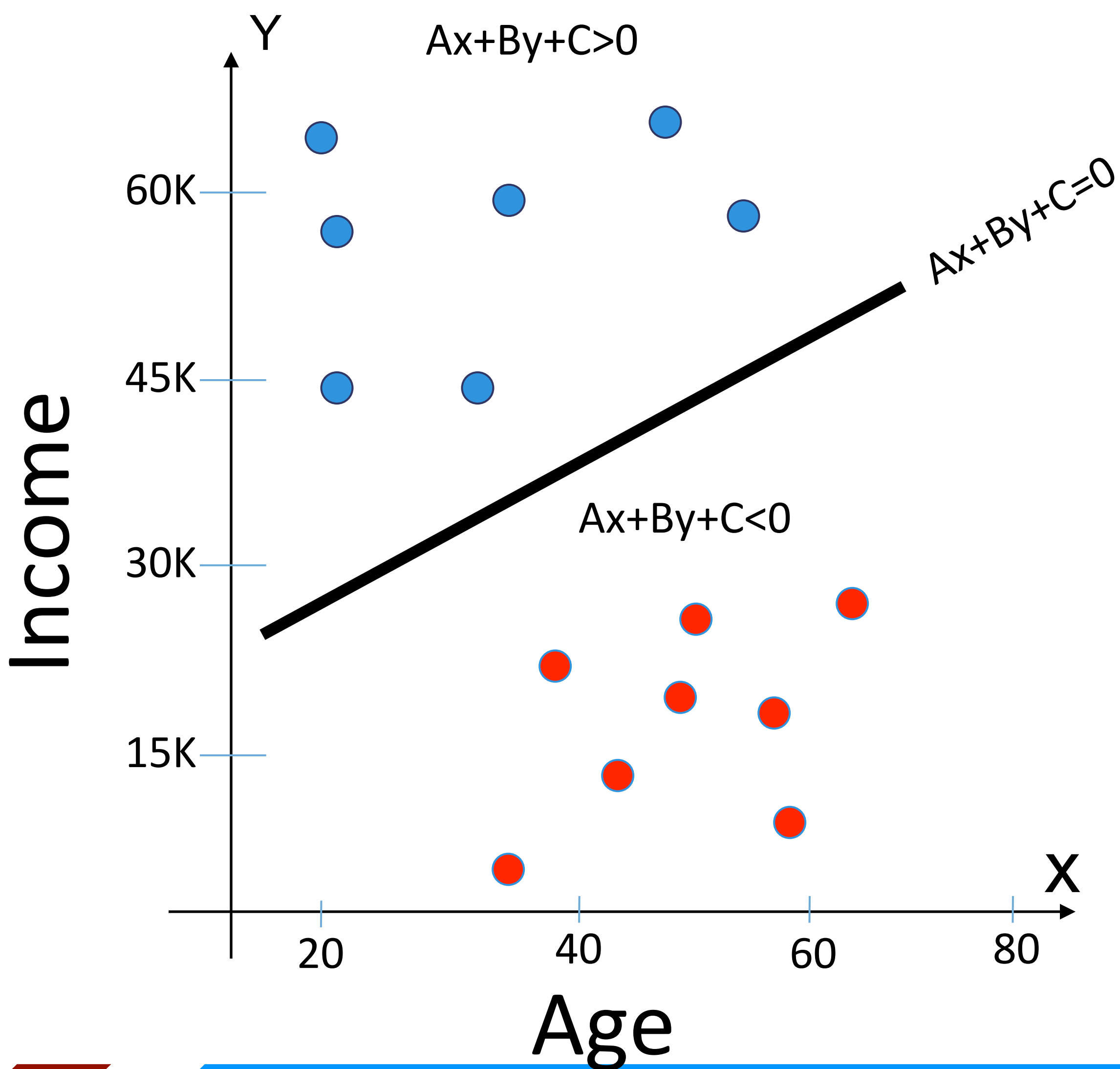
Feature Engineering is the Key

- Most effort in ML projects is constructing features
- Black art: Intuition, creativity required
  - Understand properties of the task at hand
  - How the features interact with or limit the algorithm you are using.
- ML is an iterative process
  - Try different types of features, experiment with each and then decide which feature set/algorithm combination to use

A sample machine learning Algorithm

- 2-way classification problem
  - +ve and –ve classes
- Representation: Lines (Ax+By=C)
  - Specifically
    - if Ax+By+C >0 then classify "+ve"
    - Else classify as "-ve"
- Evaluation: Number of mis-classified examples
- Optimization: An algorithm that searches for the three parameters: A, B and C.

Toy Example



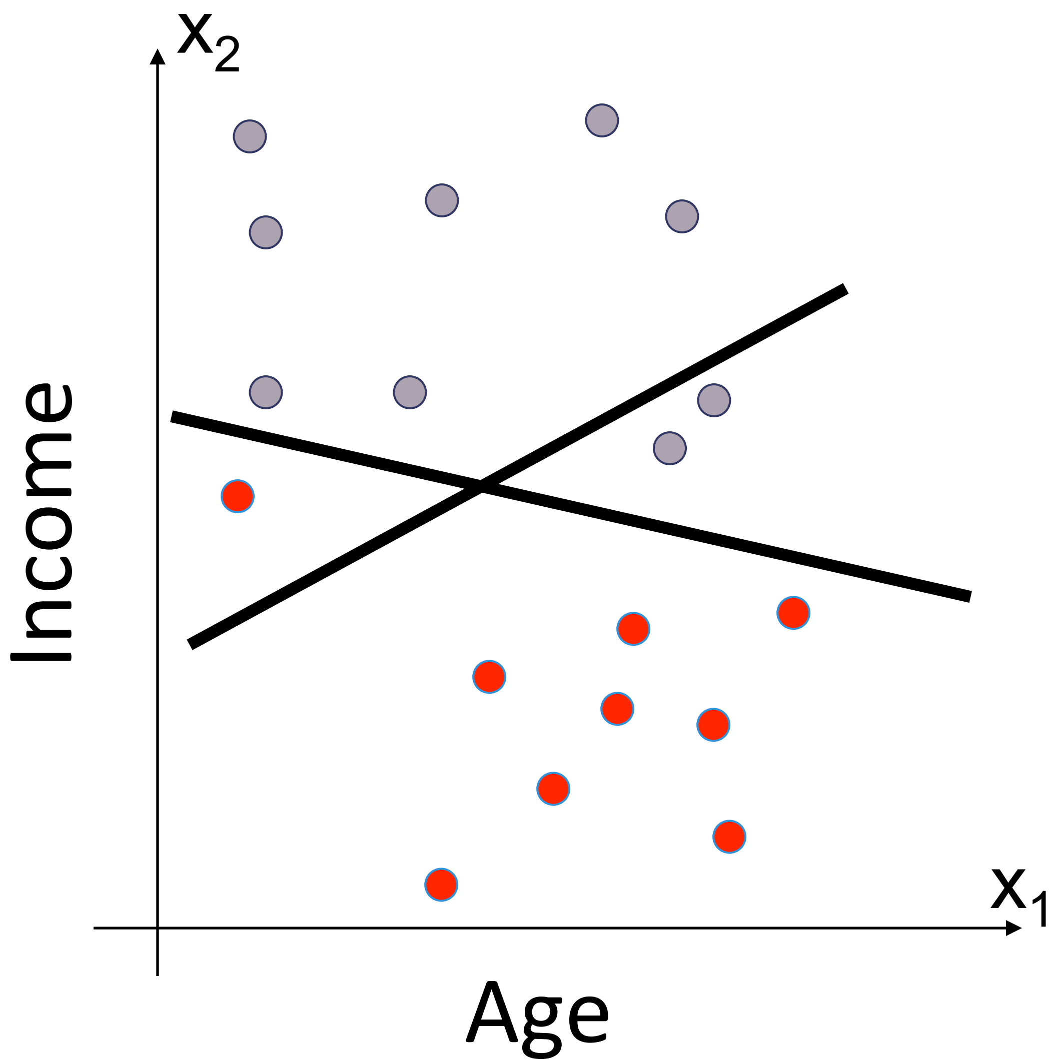
Blue circles: Good credit (low risk)
Red circles: Bad credit (high risk)

Problem: Fit a line that separates the two such that the error is minimized.

How do machine learners solve this problem?

- Try different lines until you find one that separates the data into two
- A more plausible alternative
  - Begin with a random line
  - Repeat until no errors
  - For each point
    - If the current line says +ve and point is –ve then decrease A, B and C
    - If the current line says –ve and the point is +ve then increase A, B, and C

Toy Example: More data



Blue circles: Good credit (low risk)
Red circles: Bad credit (high risk)

Problem: Fit a line that separates the two such that the error is minimized.

Learning = Representation + Evaluation + Optimization

- Combinations of just three elements

Representation	Evaluation	Optimization
Instances	Accuracy	Greedy search
Hyperplanes	Precision/Recall	Branch & bound
Decision trees	Squared error	Gradient descent
Sets of rules	Likelihood	Quasi-Newton
Neural networks	Posterior prob.	Linear progr.
Graphical models	Margin	Quadratic progr.
Etc.	Etc.	Etc.