Machine Learning

Lecture 01

Introduction to Machine Learning

Knowledge Discovery

- There are two major fields in knowledge/pattern discovery:
  - Statistics
  - Machine Learning (ML)
- They often employ the same methods and overlap significantly, and they all use techniques to learn from data for better predictions
- Despite the large number of Stat/DM/ML models developed to this day, the number of tasks these models address fall into a few fundamentally different tasks only. What are they?

Some of the tasks in knowledge discovery

- **Classification** (categorization): Predicting a discrete (binary or multi-valued) value
- **Regression** (value estimation): Predicting a continuous, numerical value
- **Clustering** : Grouping items together by their similarity
- **Similarity matching** : Identification of similar items based on data known about them
- **Association mining** (market-basket analysis) : Find associations between entities based on transactions (which items are commonly bought together?)
- **Recommendation engines** (recommender system): a subclass of information filtering system that seeks to predict the rating or preference for a product/service (example: Collaborative filtering)

Getting the terminology right

ARTIFICIAL INTELLIGENCE

An umbrella term used for computers mimicking human intelligence using logic

MACHINE LEARNING

Subset of AI: Statistical techniques for enabling computers to learn from data to perform certain tasks

DEEP LEARNING

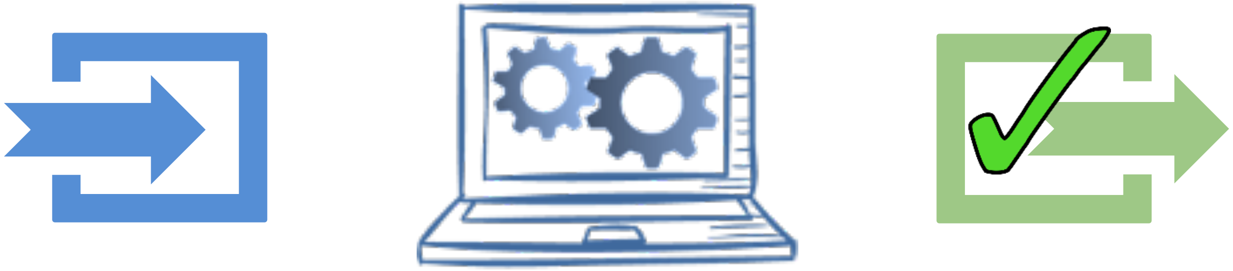
Subset of ML: Set of algorithms that learn from vast amounts of data by using Neural Networks (good at complex tasks like image/speech recognition)

What is Machine Learning?

- ML learns from experience (available data), and use the learned knowledge to make predictions and classifications on new data
- Keep in mind that ML outcomes are mere predictions with an accuracy far from 100% in most cases
- Hard facts about ML:
  - ML is not an exact science, it's an empirical field!
  - ML is not a silver bullet!
  - ML could be a hammer, but not every business problem is a nail. So, prefer simple, heuristic solutions if they work.
- How to succeed with ML?
  - Frame the business problem right
  - Know the limitations/shortcomings of ML

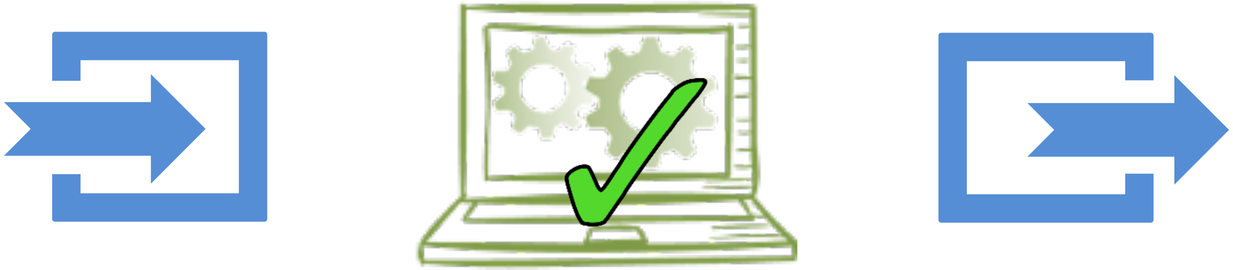
Computer programs vs Machine Learning

COMPUTER PROGRAM



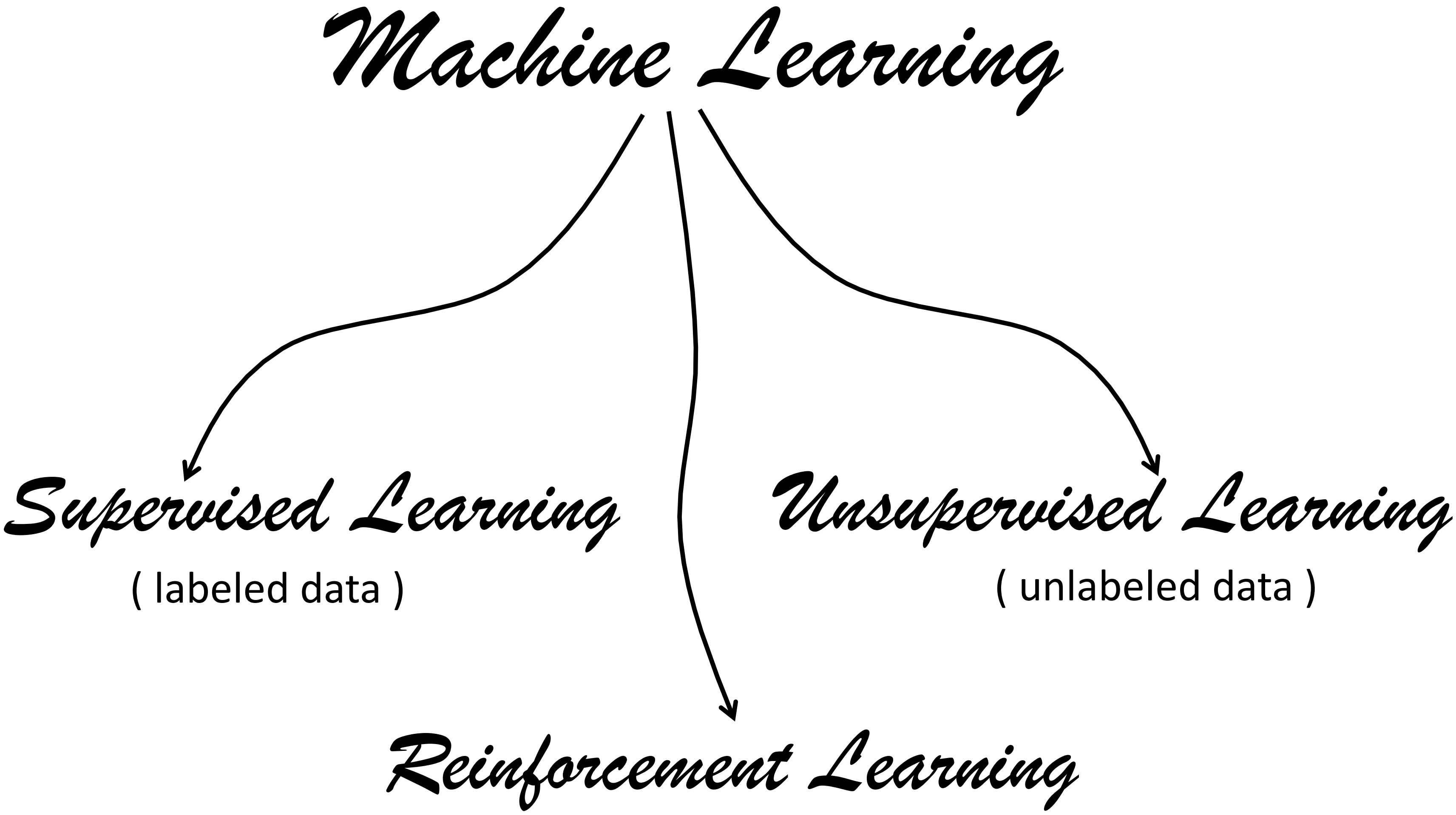
input model output

MACHINE LEARNING



input model output

Types of Machine Learning



Machine Learning

Supervised Learning

(labeled data)

Unsupervised Learning

(unlabeled data)

Reinforcement Learning

Supervised Learning

- **Regression** (predict a continuous numerical value)
- **Classification** (class prediction / class probability)
  - Binary classification (e.g. 0/1, Yes/No, malign/benign)
  - Multi-class classification (e.g. positive/negative/neutral)

Predicts between 2 categories
Two-class (binary) Classification

Is this tweet positive?

Will this customer renew her service?

A/B Which of the two ads draws more customers?

Predicts among several categories
Multi-class Classification

What is the mood of this tweet?

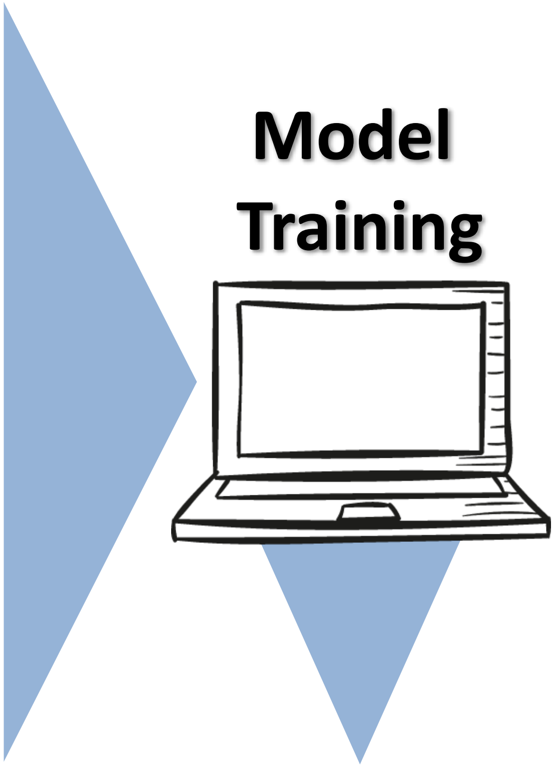
Which service will this customer choose?

A/B/C Which of the three ads draws more customers?

How does classification work?

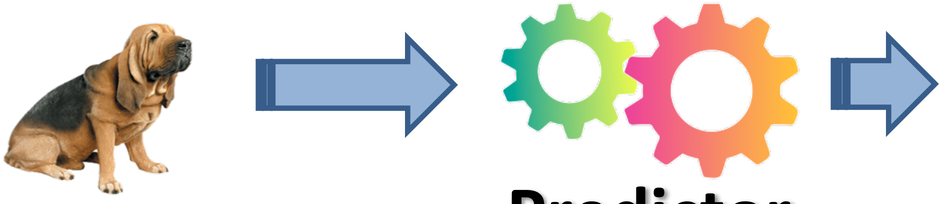
Labeled Training Data

INPUT	CLASS (label)
	CAT
	CAT
	DOG
…	…
	CAT
	DOG



Model Training

New Data



Predictor

DOG

We answer questions like:

- **Regression**
  - Based on the current feature set for this movie, what will its IMDB rating be?
  - How much will this customer use the service?
  - What will be the stock market value of Twitter in one month?
  - How much will it rain next month?
- **Classification**
  - Will it rain tomorrow?
  - Should I approve client’s credit application?
  - Will this consumer respond to our campaign?
  - Will the customer opt for service A given the incentive B?
  - Which services (S1, S2, S3, or none) will this customer be likely to purchase given incentive B?
  - Which segment does this consumer belong to?
  - Which subscribers will terminate their account in 2 months?

Supervised ML framework

- Find a prediction function **F** based on a feature representation of the object to get the desired output (by minimizing the prediction error)
- Prediction (target) function: **Y = F(X)**
- Input (feature set)

  X ={color, shape, length, width, weight}

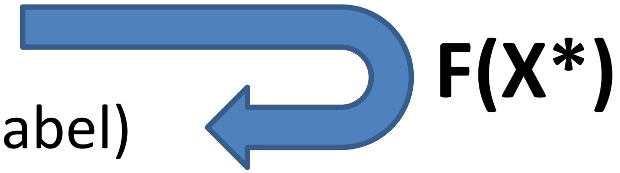
- Output (class labels)

  Y = {lemon, apple, cherry, orange, banana}

- Then apply **F** to new (unseen before) data X* to predict the value of Y* :

  X* = {yellow, ellipsoid, 7, 5, 30}

  Y* = “Lemon” (predicted class label)



Supervised ML framework – cont'd

← FEATURES → TARGET

ID	Age	Bank client	Job prospect	Housing	Credit card limit	Salary	For how long in this job?	Credit Score
	< 25 25-29 30-34 35-41 42-50 > 50	Yes No	Employee Owner	Owner Tenant	< 2000 2001-2750 2751-6000 6001-10000 > 10000	< 1000 1001-3000 3001-5000 5001-7500 7501-10000 > 10000	< 1 1-3 4-6 > 7	⊕ or ⊖
#	36	N	E	Owner	7500	6000	6	⊕
#	22	Y	E	Tenant	1750	2500	1	⊖
#	54	Y	O	Owner	15000	12500	11	⊕
#	...	...	...	...	...	...	...	...

TRAINING DATA

Objective: Finding function " **f** " to predict credit score for a new input

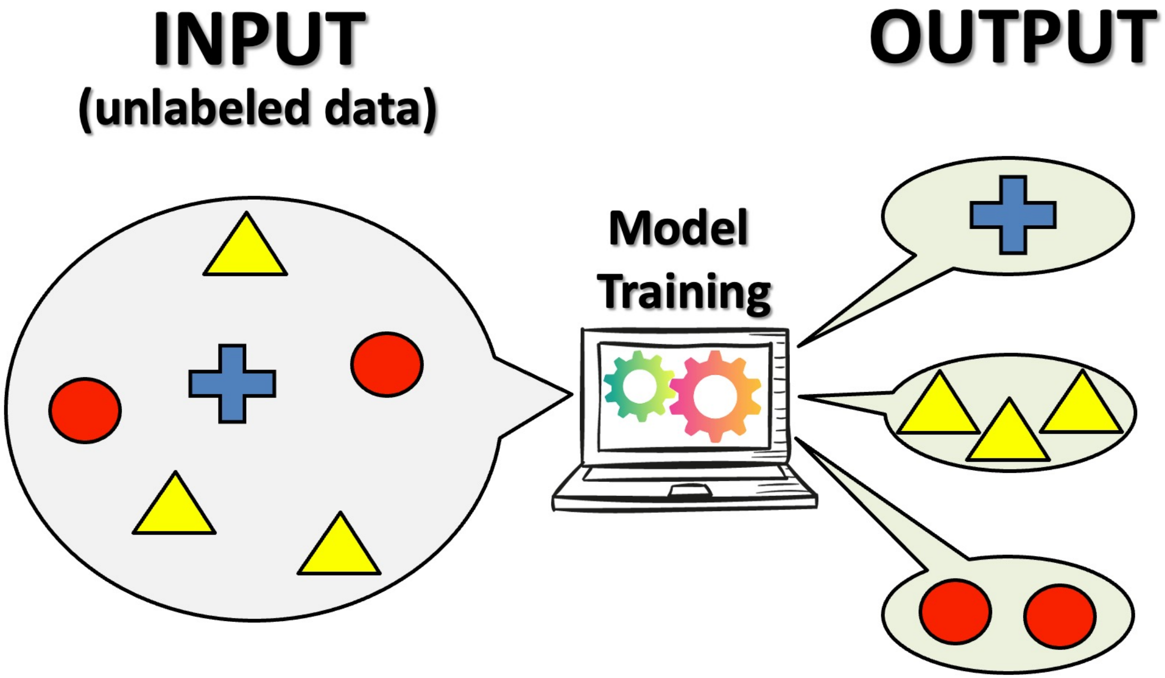
Output:
Y = { ⊕ or ⊖ }

Y = **f** (x)

Input: x = {37, Y, E, Tenant, 10000, 5000, 1-3}

Unsupervised Learning

- Clustering: Identifying patterns without labels



- Association rule mining (Apriori)
- Anomaly detection (also supervised)
- Dimensionality reduction (PCA)
- Others: Page Ranking, Recommendation systems

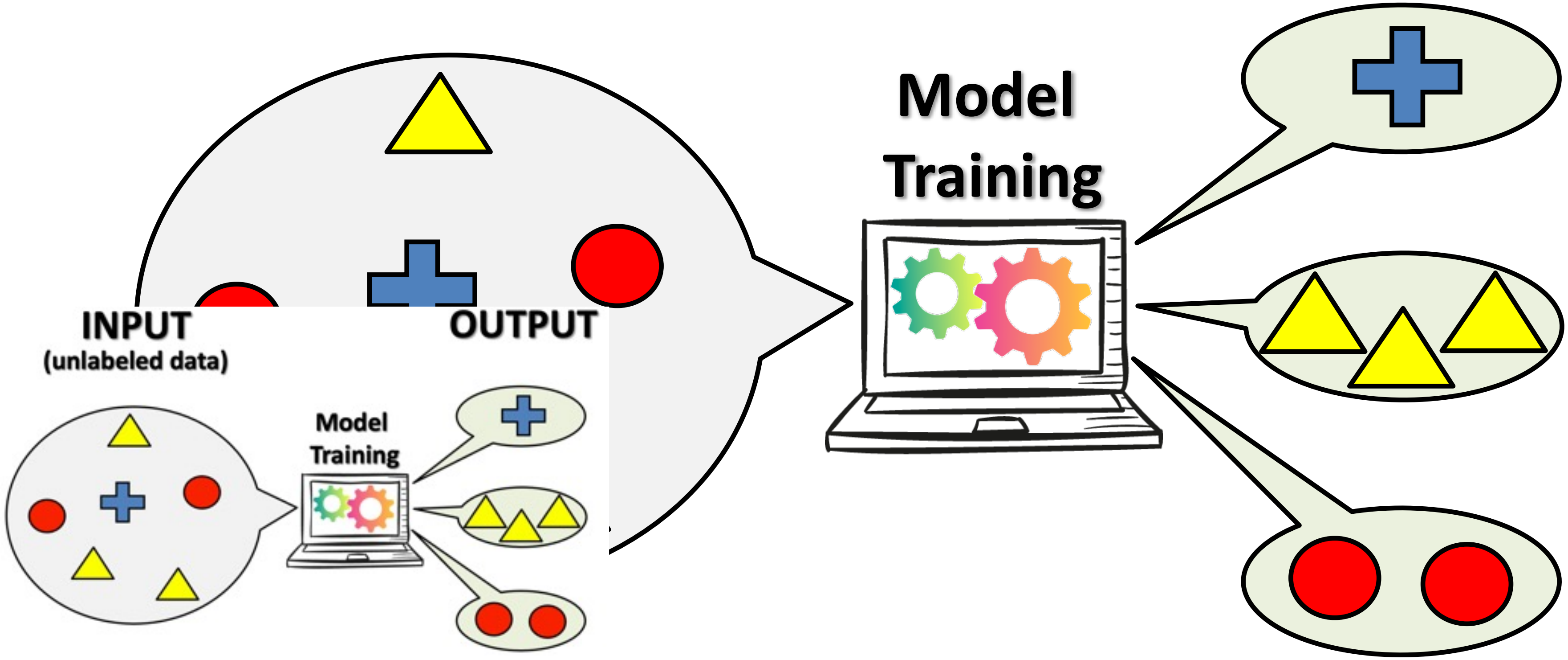
Clustering (unsupervised)

- Identifying patterns without labels (similarity-based)

INPUT

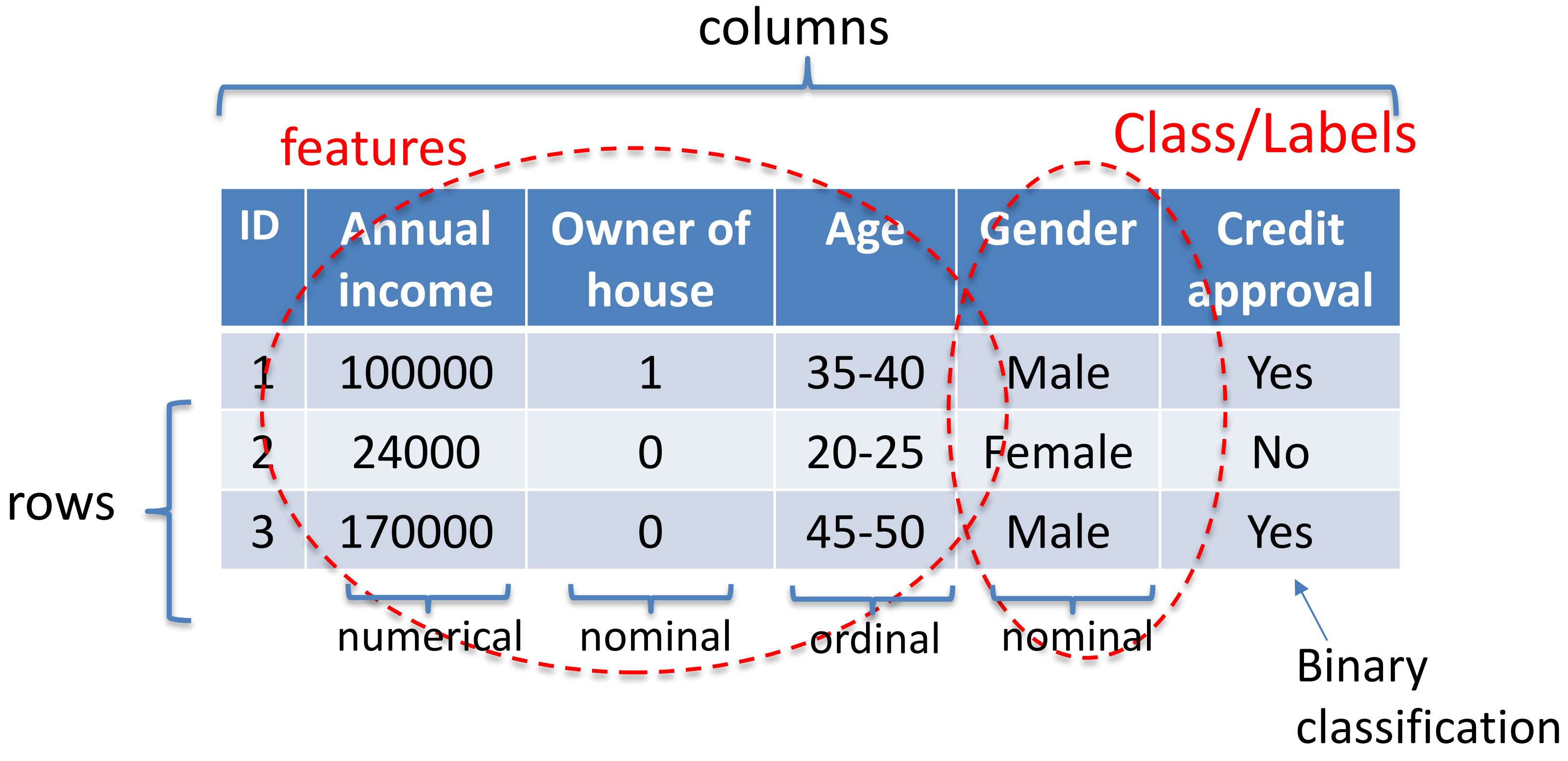
(unlabeled data)

OUTPUT



Model Training

ML terminology



ID	Annual income	Owner of house	Age	Gender	Credit approval
1	100000	1	35-40	Male	Yes
2	24000	0	20-25	Female	No
3	170000	0	45-50	Male	Yes

- **Rows**: observation, sample, example, instance, record
- **Columns**: feature, attribute, predictor, dimension, (independent) variable, input, regressor, covariate
- **Class**: response, target, class, label, outcome, dependent variable (what we're trying to predict)

Types of classifiers

- Linear vs Nonlinear Classifiers
- Generative vs Discriminative Classifiers
- Parametric vs Non-parametric Classifiers
- Lazy vs Eager Learners

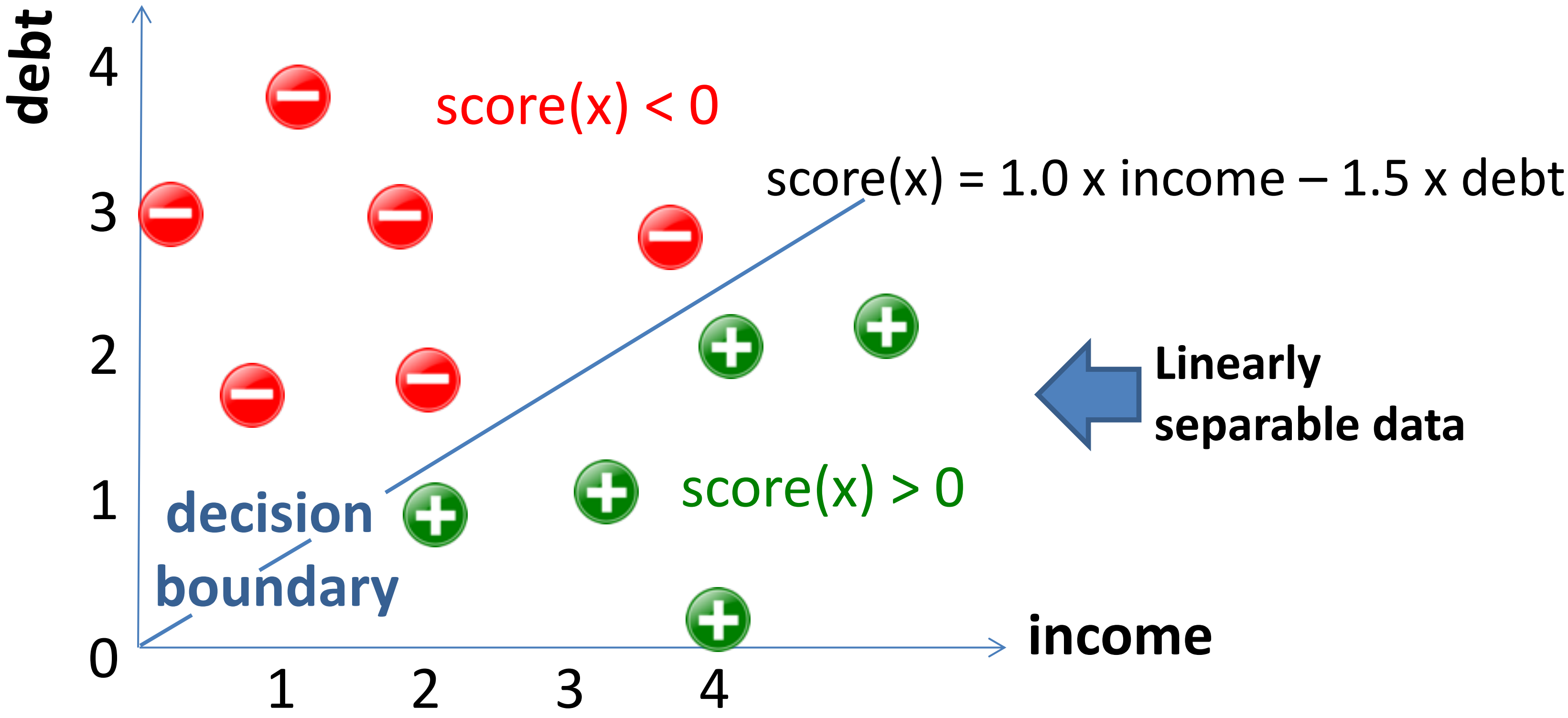
	Parametric	**Non-parametric**
Generative	Naive-Bayes*	
Discriminative	Linear Regression* Logistic Regression* Perceptron* Bagging/Boosting	Linear SVM* Nonlinear SVM Decision Trees kNN**

* Linear classifiers

** Lazy learners

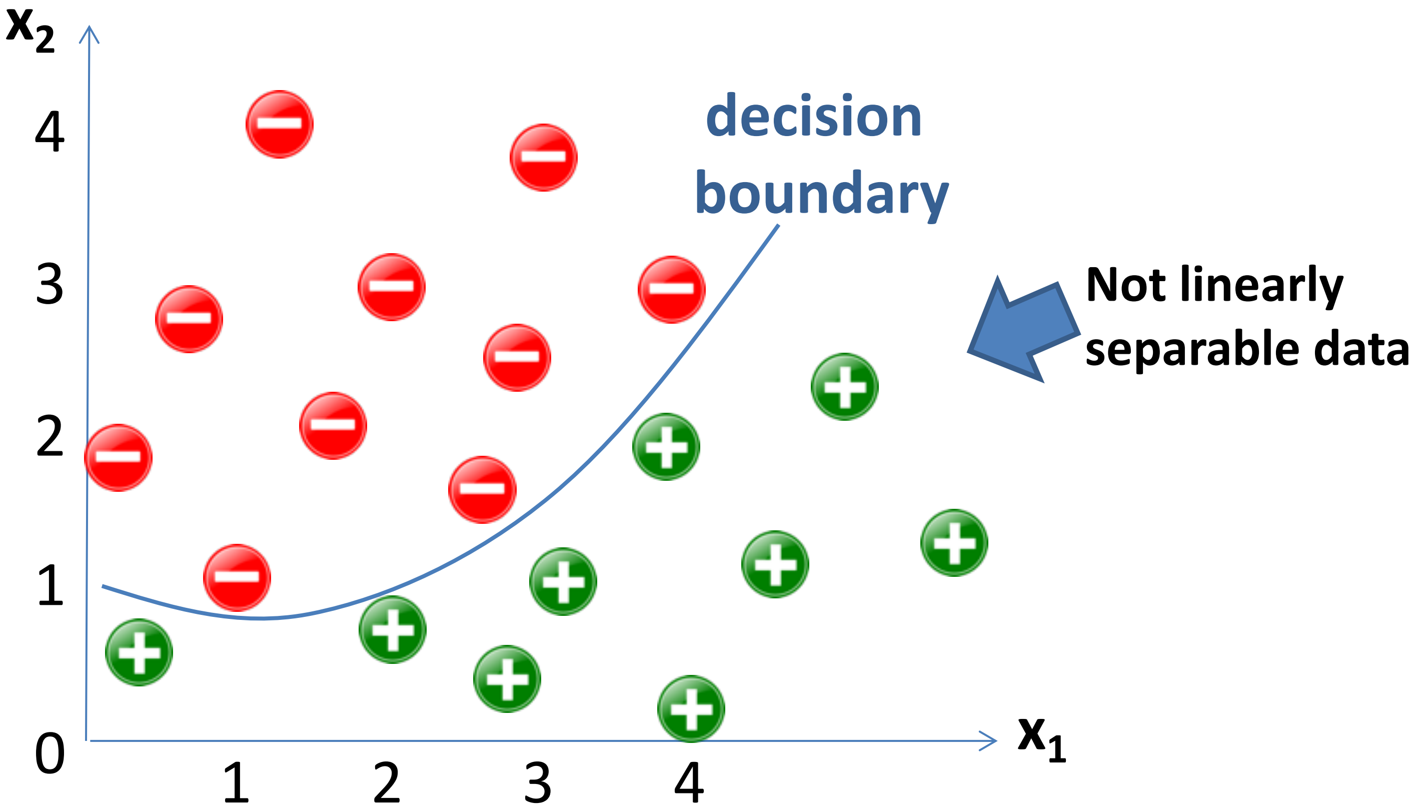
Types of Classifiers – cont'd

- **Linear classifiers**
  - Separate input vectors into classes using linear (line, plane, hyperplane) decision boundaries (classification is based on the value of a linear combination of the weighted features).
  - **Linear regression**, **Logistic regression**, **Perceptron**, **Linear SVM**, **Naive-Bayes** (under certain circumstances)



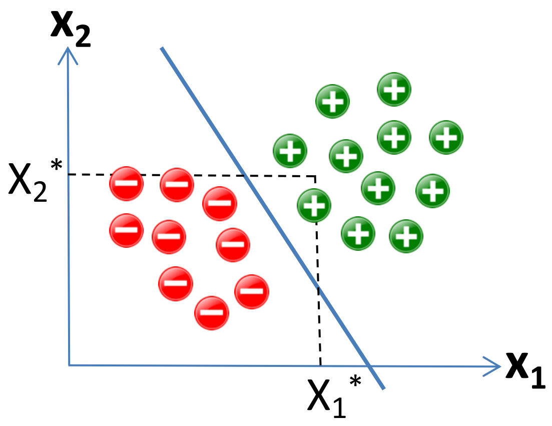
Types of Classifiers – cont'd

- **Non-linear classifiers**
  - Non-linear classifiers have non-linear, possibly discontinuous decision boundaries (**kNN**, **Multi-layer perceptron**, **Neural networks**, **SVM with a kernel**, **Decision trees**)



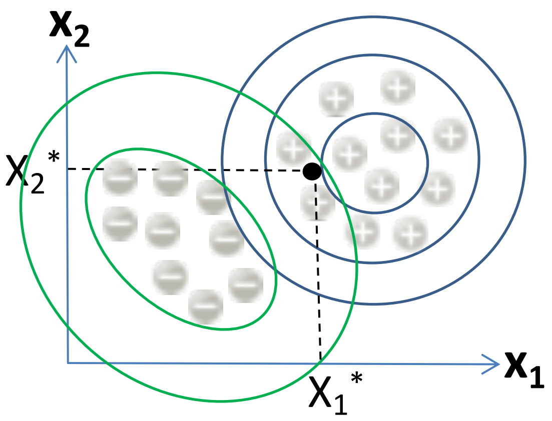
Types of Classifiers – cont’d

Discriminative classifiers



In determining the class label, a discriminative classifier finds the optimum separating line between the classes and focuses on which side of the decision boundary the new data point falls (e.g. Linear/Logistic Reg., SVM, LDA, Perceptron, Decision Tree).

Generative classifiers



In determining the class label, a generative classifier builds models based on what the (+) and (-) classes look like (a complete probability distribution for each class). The new data point takes the class label of the model it lies in (e.g. Naive-Bayes).

Types of Classifiers – cont'd

- **Parametric classifiers:** These summarize data with a set of parameters of fixed size. We choose a form for the function and then learn the coefficients from the training data.
  - Examples: Linear Regression, Logistic Regression, Perceptron, LDA, Naive Bayes, Simple Neural Networks, Bagging/Boosting*
- **Non-parametric classifiers:** These don't use parameters to specify their model but instead use the training data to learn the mapping function while maintaining some ability to generalize to unseen data.
  - Examples: kNN (k-nearest neighbors), Decision trees, Support Vector Machines (SVM)

* Although the individual models in an ensemble could be non-parametric (like decision trees), the ensemble model itself is considered parametric