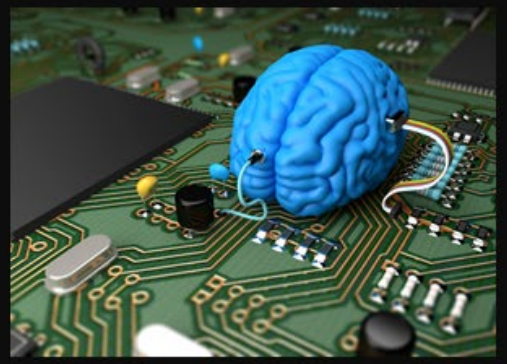
Machine Learning

What society thinks I do



What my friends think I do

What other computer scientists think I do

What mathematicians think I do

What I think I do

```
from theano import *
```

What I actually do

Review: Scaling Content

- We can support **scale** in web systems by employing:
  - **DNS tricks**: have the DNS server return multiple IP addresses
  - **Load balancing**: use a network appliance to distribute requests
  - **Replication**: use lots of copies of the same server software
  - **Content distribution networks**: use lots of copies of static assets
  - **Caching**: store local copies of everything to lower network traffic
- **Replication** can be tricky because of writes/updates
  - **Sharding** is the careful splitting of databases across multiple servers
  - **Consistency** is hard to guarantee without serialization points
- **Cloud providers** make heavy use of **virtualization**

Current challenges in Datacenters

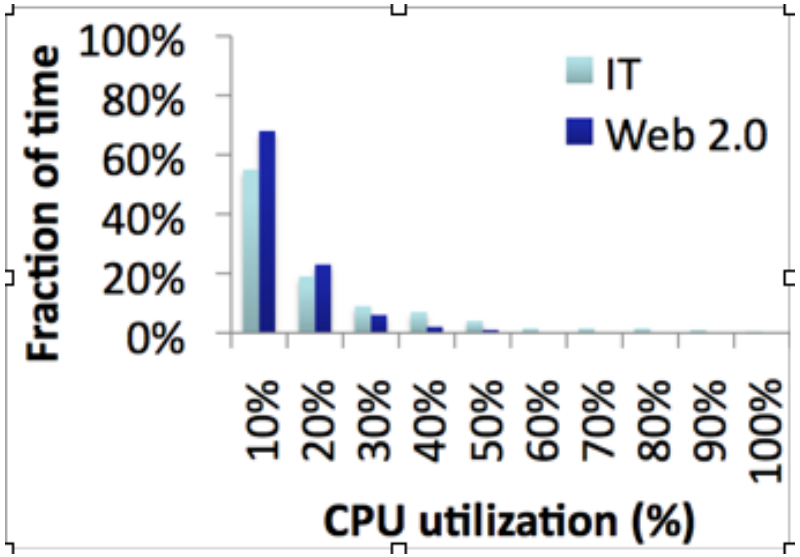
- Whole data center must be optimized, not just filled with more efficient computers
  - CPUs only account for 12% of energy!

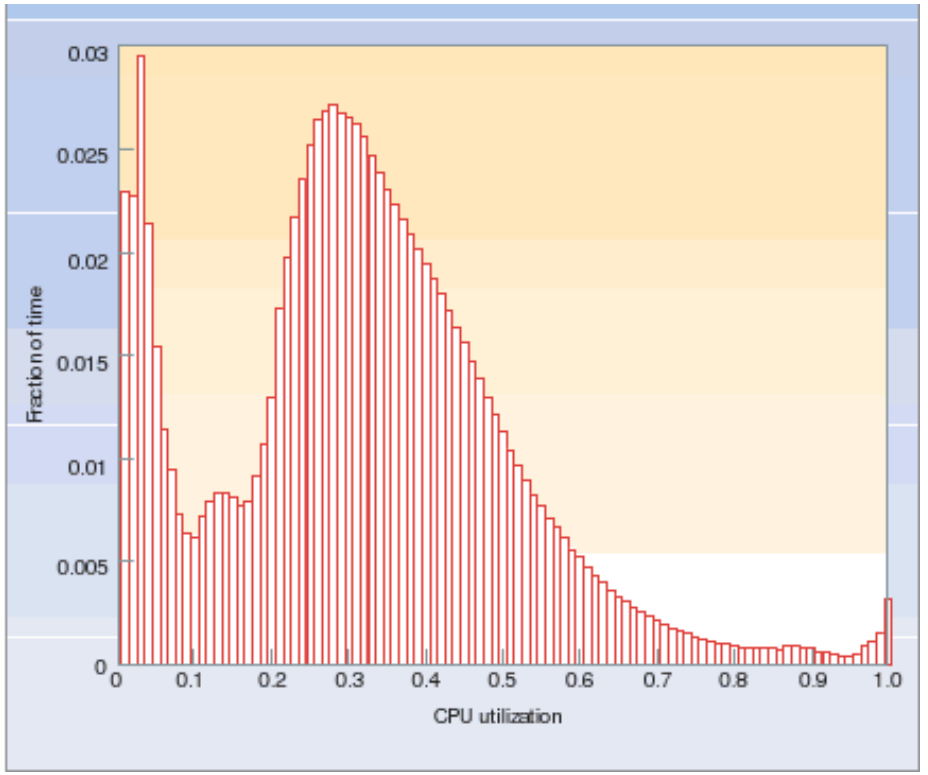
- Challenges:
  - Power infrastructure very inefficient
  - Utilization & poor energy-proportionality
  - Cooling efficiency

- Green energy (e.g., wind farms)

Utilization

- Machines usually very poorly utilized





Utilization

- Why so poorly utilized?
  - Work spread over many machines for robustness, data safety
  - Natural variance in load means most times will not be peak
    - Is it ever possible to do more work during off-peak times to reduce work during peak times?
    - Often, no
  - Server-class machines often mismatched to Web workloads
  - Many background tasks mean machines never completely idle

Cloud Computing: Let's Share Computers

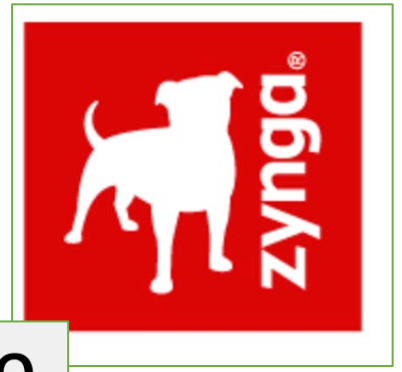
- The **cloud** refers to a collection of practices used to increase utilization and decrease costs by offering computing services **on-demand**

- **Idea:** Cloud Provider has powerful rack servers and infrastructure
  - Users (companies) "rent" CPU time, or RAM, or storage, from Cloud Provider
  - Users pay for what they need, no more, no less
  - Need a webserver? Buy a chunk of the Cloud Provider's servers

- **Benefits:** Each "tenant" may have low utilization, but with multiple "tenants", Cloud Provider has high utilization overall

Cloud services have taken over

- Most web services aren't run on dedicated hardware anymore



Netflix finishes its massive migration to the Amazon cloud

After move to Amazon, only the DVD business still uses traditional data center.

JON BRODKIN - 2/11/2016, 1

When Amazon's cloud storage fails, lots of people get wet

By MAE ANDERSON, AP TECHNOLOGY REPORTER
NEW YORK — Feb 28, 2017, 7:50 PM ET

Share with Facebook Share with Twitter

Cloud services have taken over

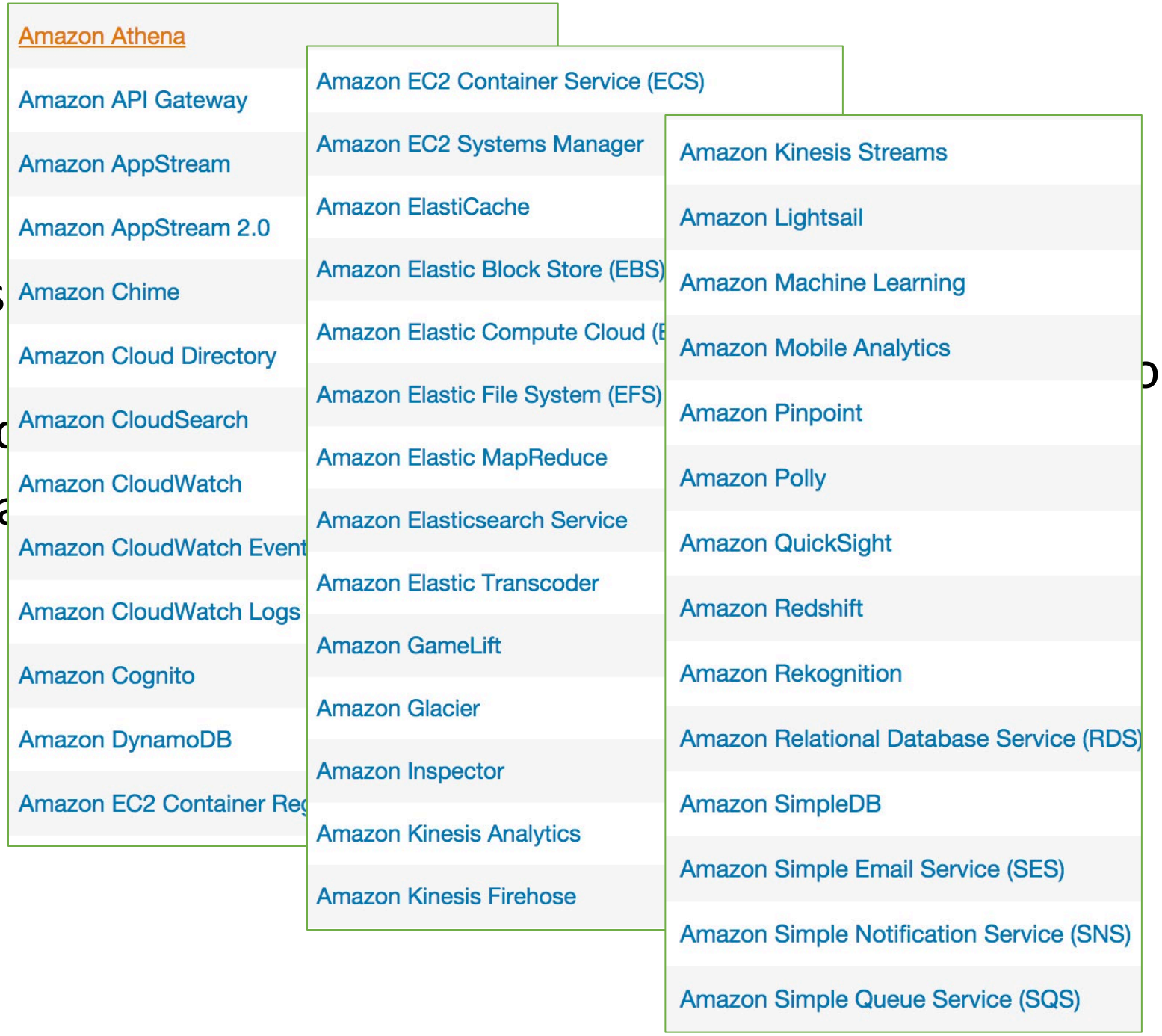
- They’re run on Amazon Web Services, Google Cloud, or Microsoft Azure

What IS the cloud?

- Unspoken: we don’t really agree on a definition
- But increasingly, it refers to compute services offered on hosted hardware and software
- Amazon, Microsoft offer a slew of services *for rent*
  - On-demand virtual machines
  - On-demand storage
  - On-demand GPUs

Wh

- Uns
- But
  hard
- Ama



on hosted

Core Amazon Web Services (AWS)

- Two services are the most important for us:
  - ***Elastic Compute Cloud*** ("EC2") offers machines for rent
  - ***Simple Storage Service*** ("S3") offers very basic Web storage

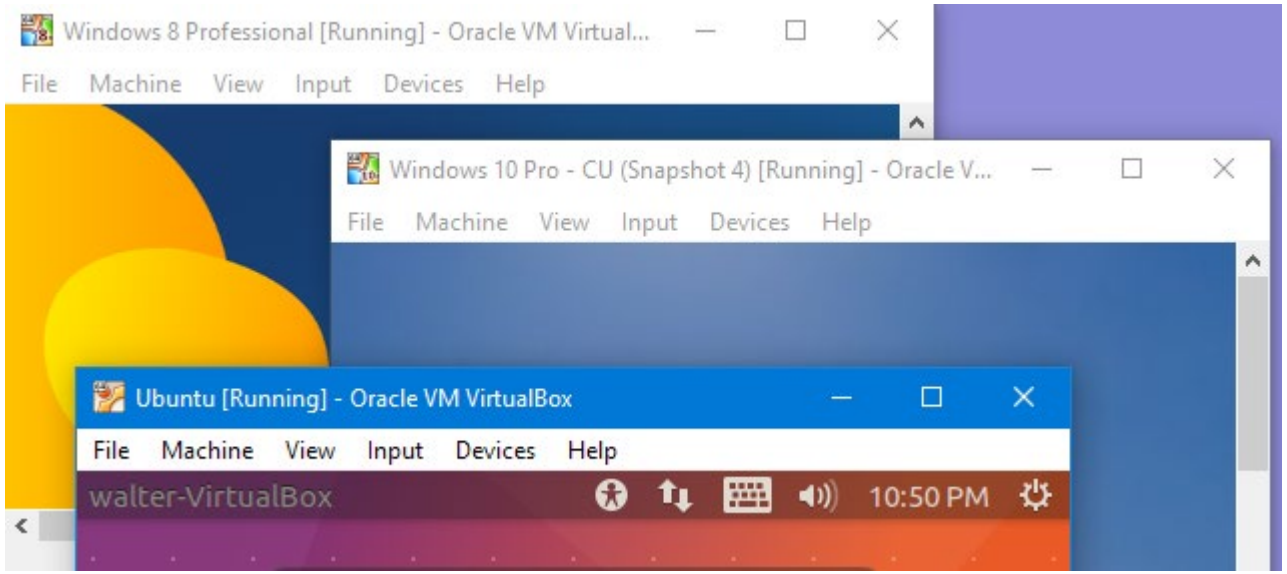
- An EC2 "instance" with 16 CPUs, 64GB costs $0.80 an hour.
  - That's $7,008 a year!
  - Dell will sell me something similar for ~$4000.
  - Why would I ever use AWS?

Advantages of Infrastructure-as-a-Service

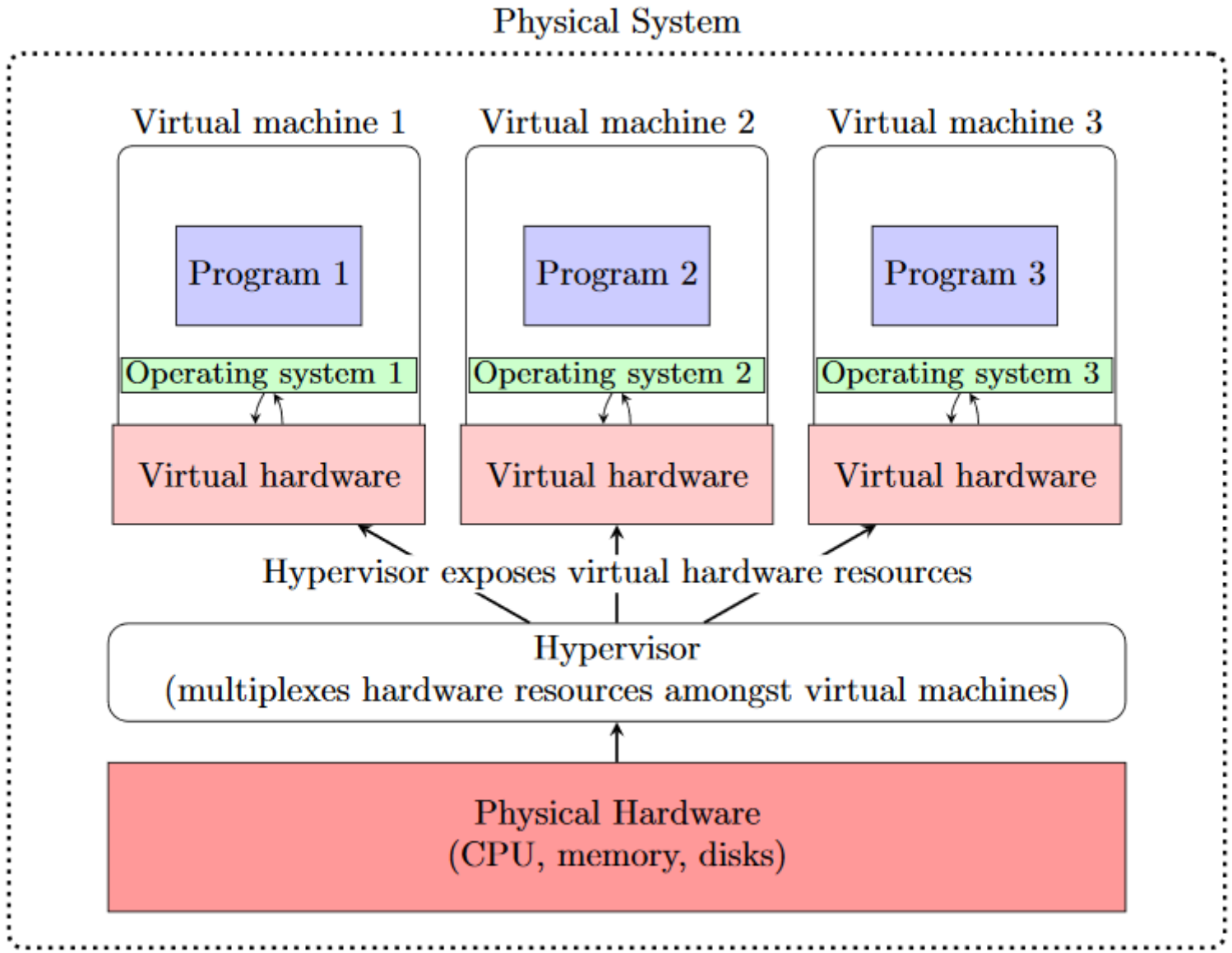
- Pay **only** for what you consume
  - LOTS of different sized machines
  - You don’t have to **overcommit** for resources
  - **Term-limited**: If your services needs to stop, you just stop paying
- Economies of scale: Cloud Provider manages racks, IT admin expertise
- Spin up new machines in minutes
  - No costly purchase and setup of new hardware
- No need for local admin expertise

Virtualization

- A key part of cloud economics is **virtualization**
- **Virtualization** is the *emulation* of hardware resources to execute multiple independent operating systems under *one physical machine*
  - Basically, you can run a whole **operating system** as an “app”



Hypervisor-based virtualization



Hardware virtualization

- Gives guest OS the illusion that it has its own dedicated hardware
- Illusion may be stronger (VMWare hypervisor) or weaker (Xen)
  - Recent trend: **containerization** (docker)

- Provides relatively strong resource and security **isolation** between operating systems
- Great properties for shared hosting environments!
  - One giant physical server with lots of small virtual machines
  - Remember your AWS instance?
    - The free tier is ~512MB RAM. Who has that little RAM anymore?

One-Slide Summary: Machine Learning

- **Machine Learning** is the use of **applied statistics** to compute a function that predicts *future data* based on *historical data*
  - **Learning a function** just refers to how the target function $\hat{f}(x)$ gets computed
  - **Supervised learning** leverages **labels** in datasets to assist in learning a function
  - **Unsupervised learning** leverages *statistical properties* of datasets to learn a function
  - Once learned, the function $\hat{f}(x)$ **predicts labels** for new datapoints, x
- **Classifiers** are types of learned functions that assign each datapoint to set of **finite labels** (i.e., "classes")
- **K nearest neighbor** is a machine learning technique that gathers data into **clusters** about **centroids** in some multidimensional space

Web activity logs

- Web activity logs are an absurdly rich source of information
- People have examined logs to learn all sorts of things
- Ideas of what you could learn about someone from their web activity?

Web activity logs

- Web activity logs are an absurdly rich source of information
- People have examined logs to learn all sorts of things
  - System failures
  - Security intrusions
  - Buying habits
  - Spelling habits
  - Web browsing behavior
  - Impending disease

Machine learning introduction

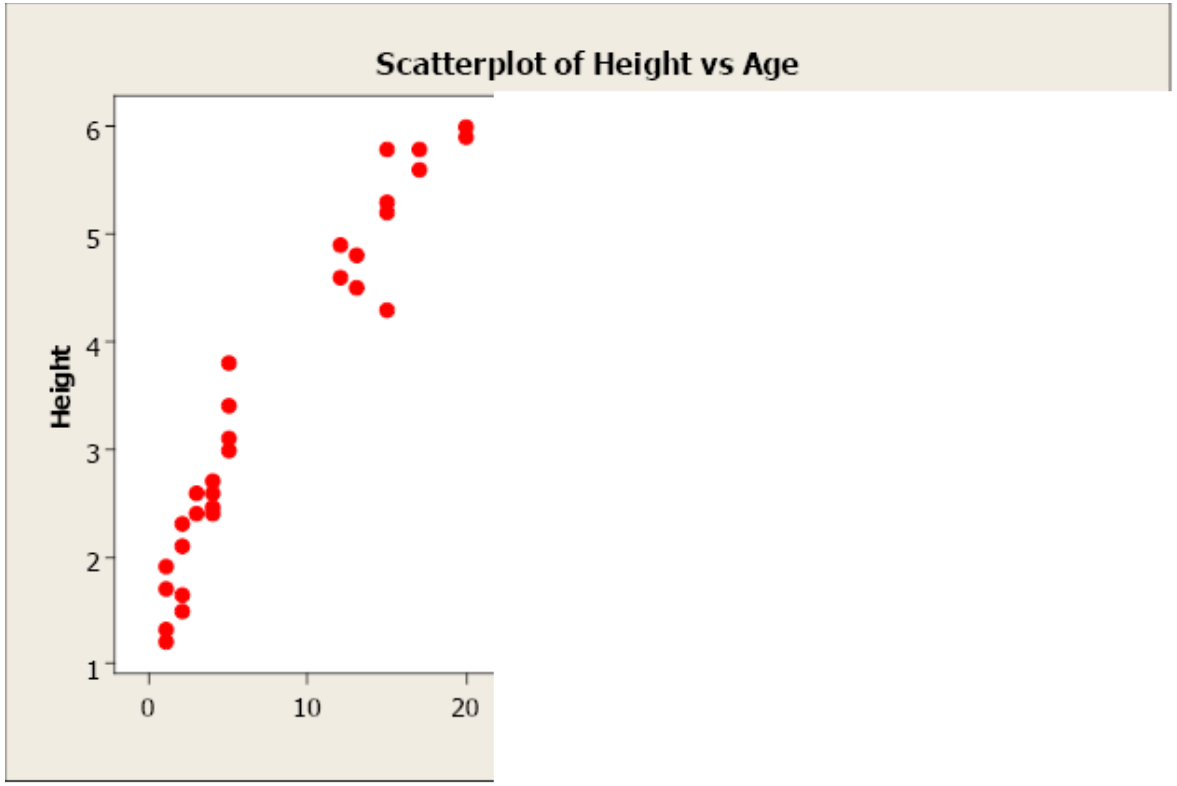
- **Machine Learning** is the application of **statistics** to large corpora of **data** to derive *predictive insights*
  - Automate science: use algorithms to find effects in data
  - Can you predict peak usage times on your service to manage scaling?

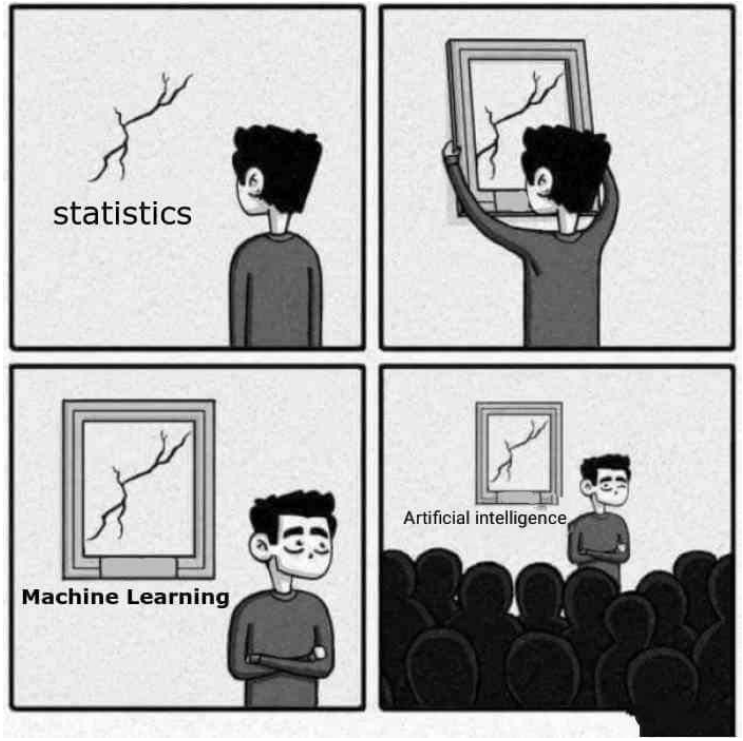
- **Motivation**: So much data is available, knowing trends is valuable

- What kinds of data have you generated today?
  - Prediction: 1 floppy disk of information (~1.5MB) generated per second for each human by 2021
  - Can we use that data to predict useful things?

Machine Learning

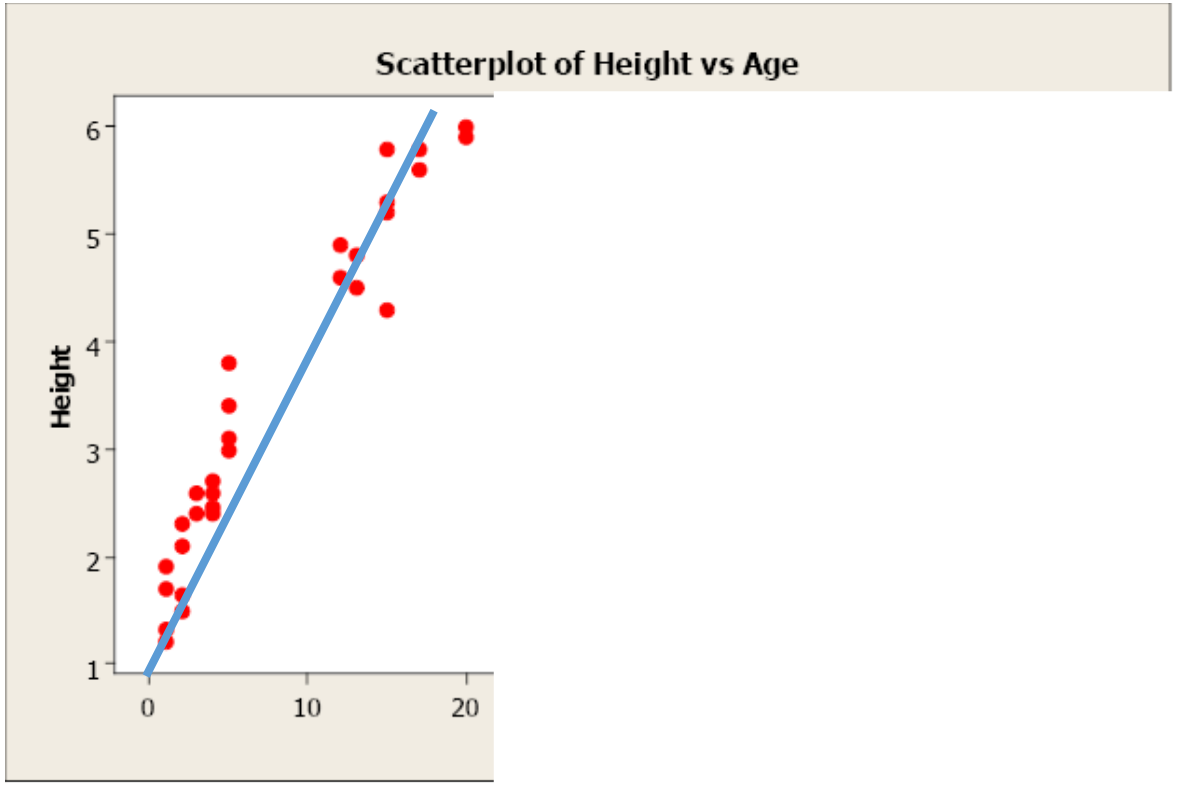
- **ML** is an application of statistics to **make predictions from existing data**

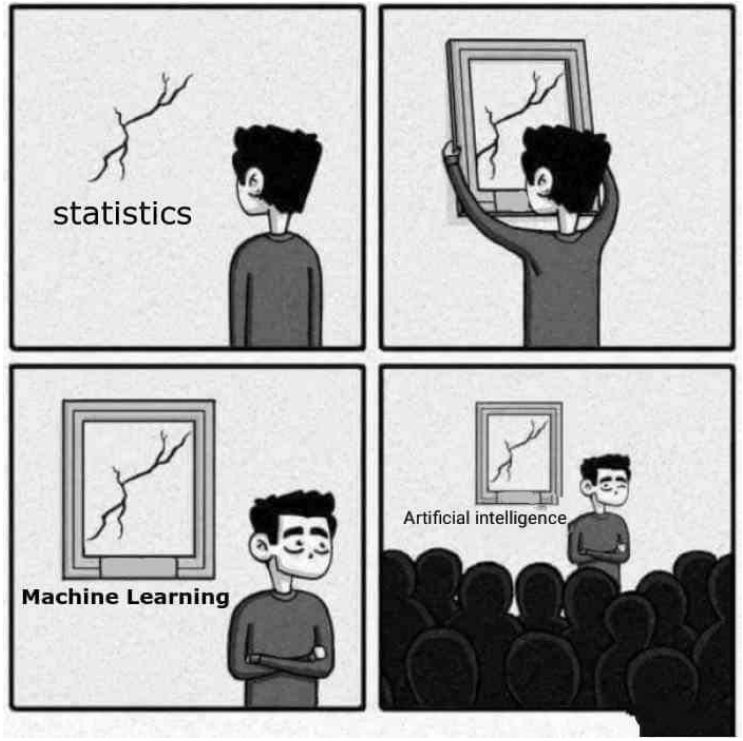




Machine Learning

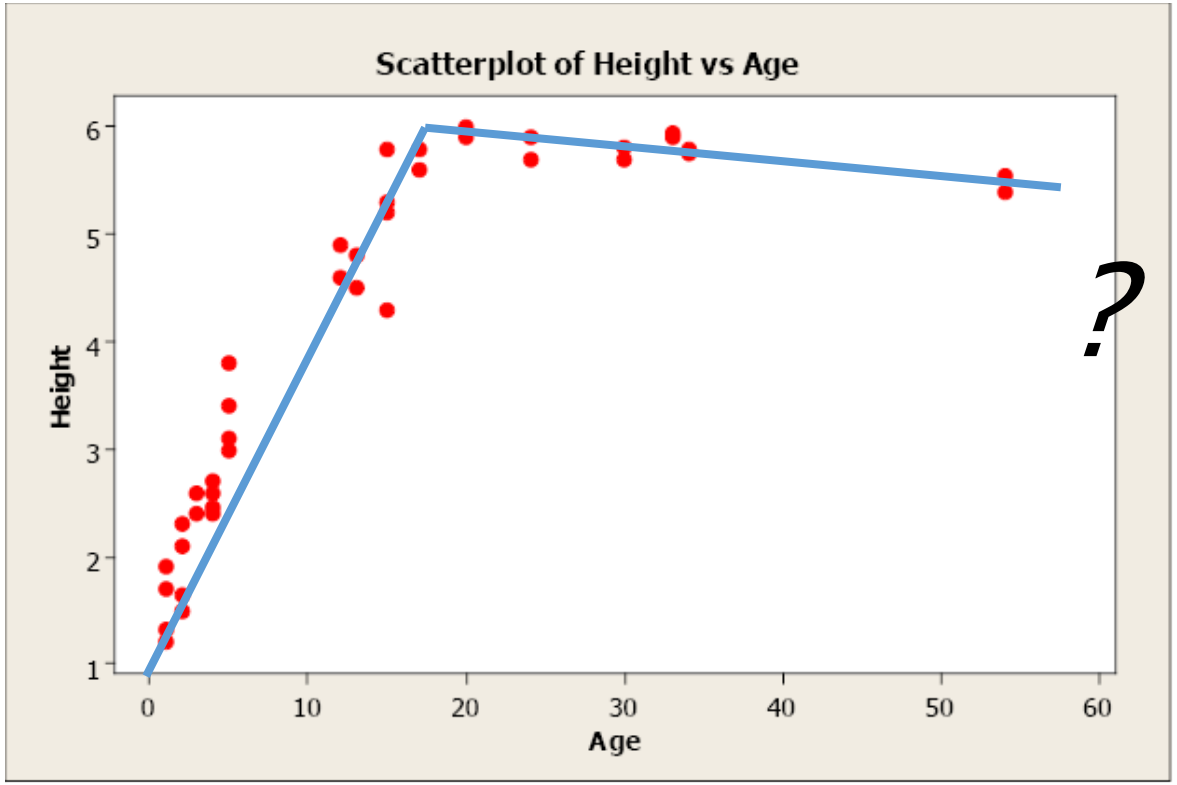
- **ML** is an application of statistics to **make predictions from existing data**

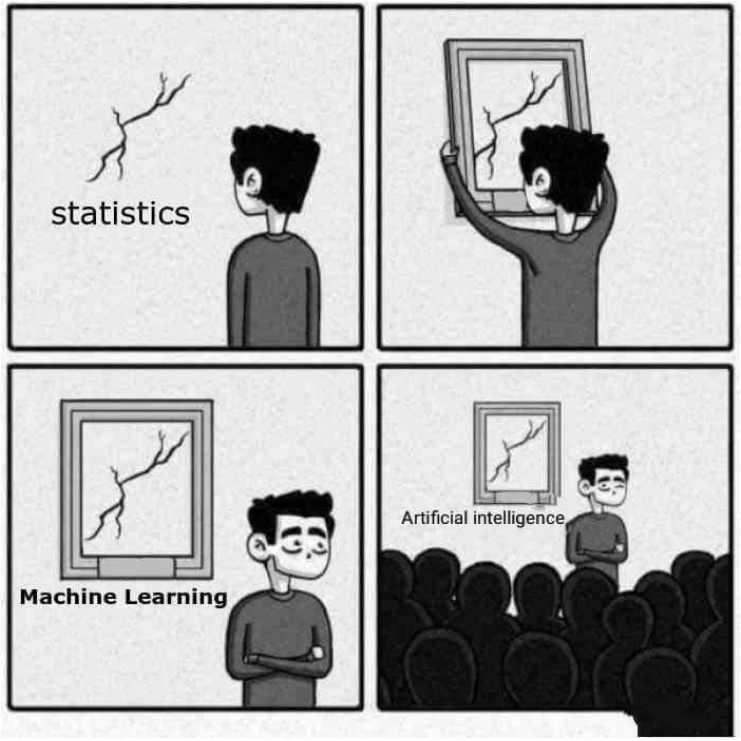




Machine Learning

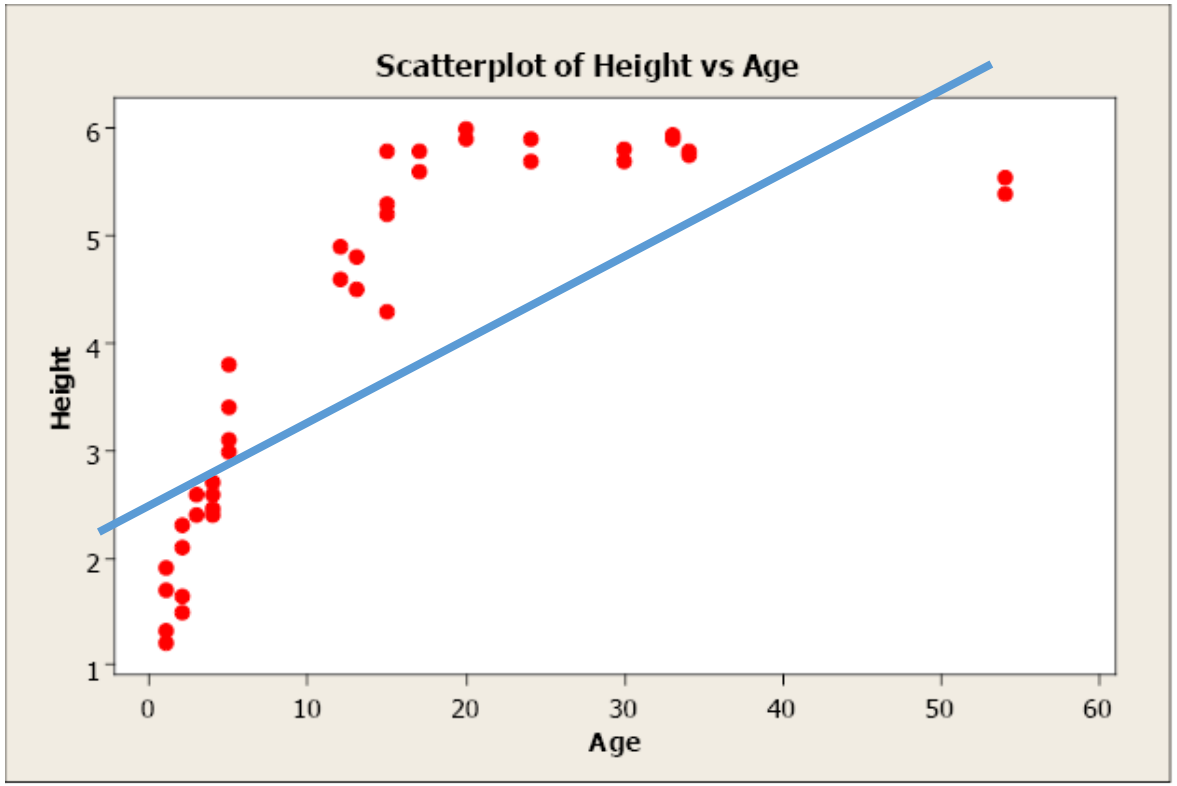
- **ML** is an application of statistics to **make predictions from existing data**



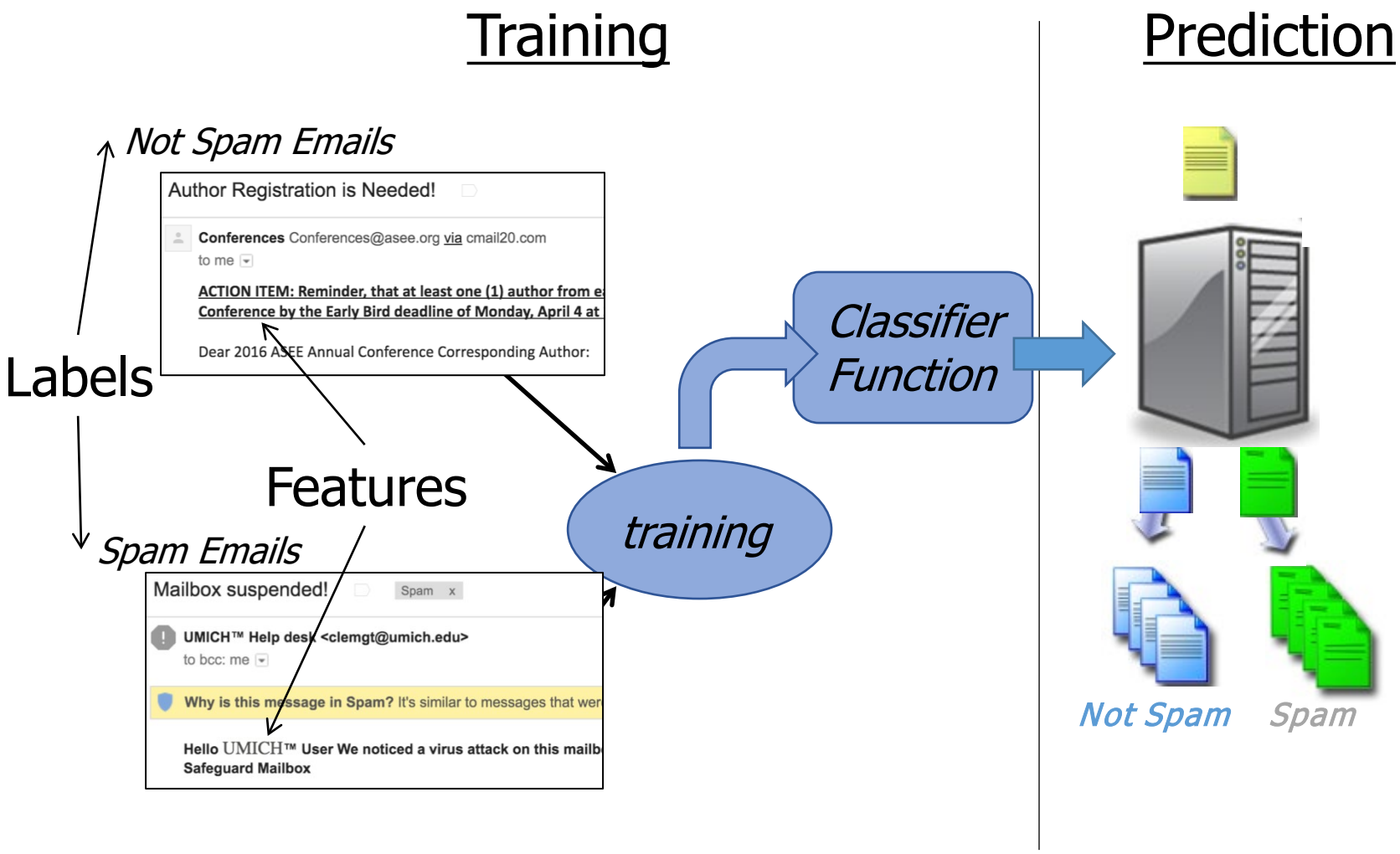


Machine Learning

- Must manually
  - Select features (e.g., age)
  - Hypothesize relationship (e.g., linear, piecewise, quadratic…)
- **Time consuming, but interpretable**
- Relies on domain knowledge



Example: spam filtering



Types of learning

- Supervised learning
  - Example: Spam filtering
  - Training data with known correct labels
- Unsupervised learning
  - Example: grouping similar news articles
  - Training data without any labels
- Reinforcement learning
  - Example: robots that adapt to their environments
  - Training takes form of positive or negative encouragement, not correct labels

Supervised learning

- Inductive learning, or "prediction"
  - Given examples of a function (X, F(X))
    predict F(X) for a novel value X

- **Classification**
  - F(X) is discrete; is page relevant or not?
- **Regression**
  - F(X) is continuous; value of GOOG?

Supervised learning

- Three ingredients
  - **Data**, with labels (the (x, f(x)) pairs, e.g., (time, stock price))
  - **Features** (how the data is represented, the components of "x", e.g., time, #employees)
  - **Machine learning algorithm** (e.g., linear regression, neural network, decision tree)

- Where do the labels come from?
- Where do the features come from?
- The set of labels tell the algorithm how to weigh competing evidence supplied by the features

- In many tasks, labeled data is in short supply and is the bottleneck

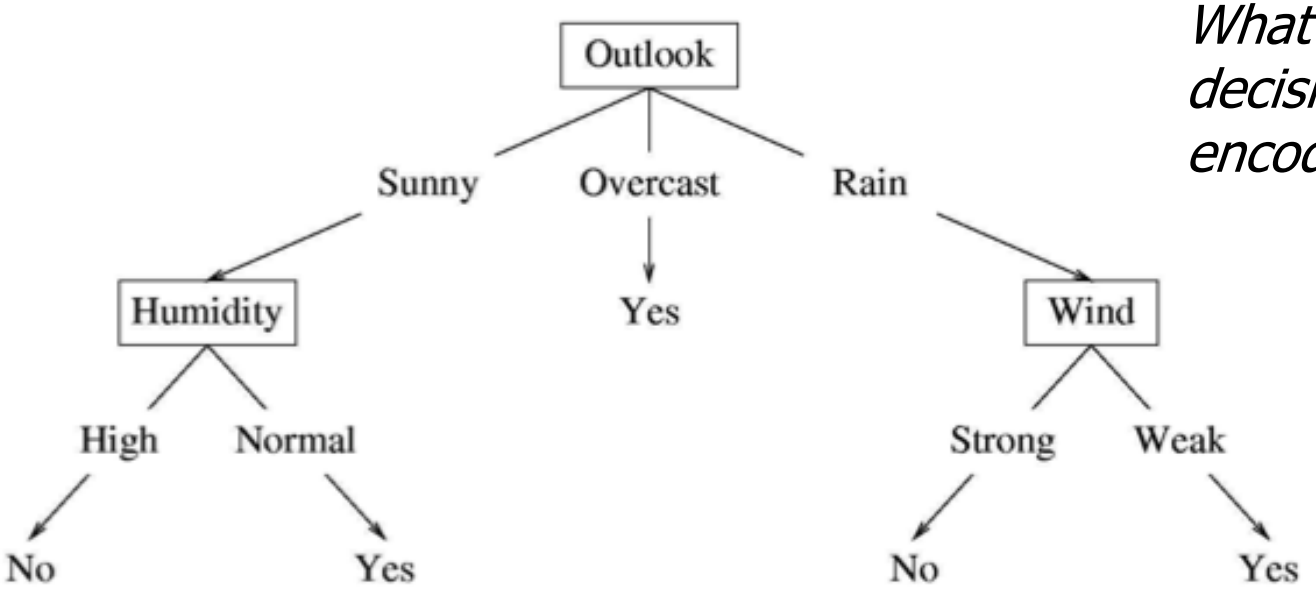
Supervised learning thought questions

- Imagine you want to predict the **age** and **gender** of a Twitter user
  - What kind of supervised task is this?
  - What are the features?
  - What dataset is used to generate features?
  - How do you generate the labels?

- Imagine you want to classify a Jeopardy question as “geographic” or not
  - Same questions as above

Classifier algorithms

- Lots of classifier algorithms possible
- One example: decision trees
  - Build a tree in which input variables are at internal nodes outputs at leaves

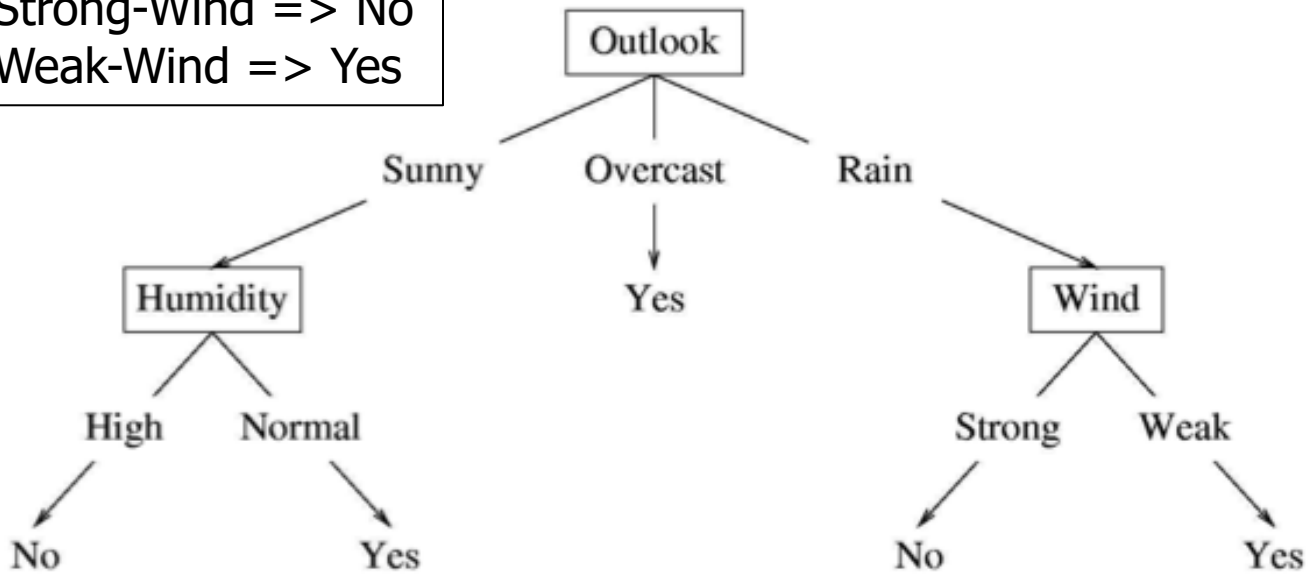


What is this decision tree encoding?

Classifiers

- Build a series of rules that are conjunctions of tests on input variables

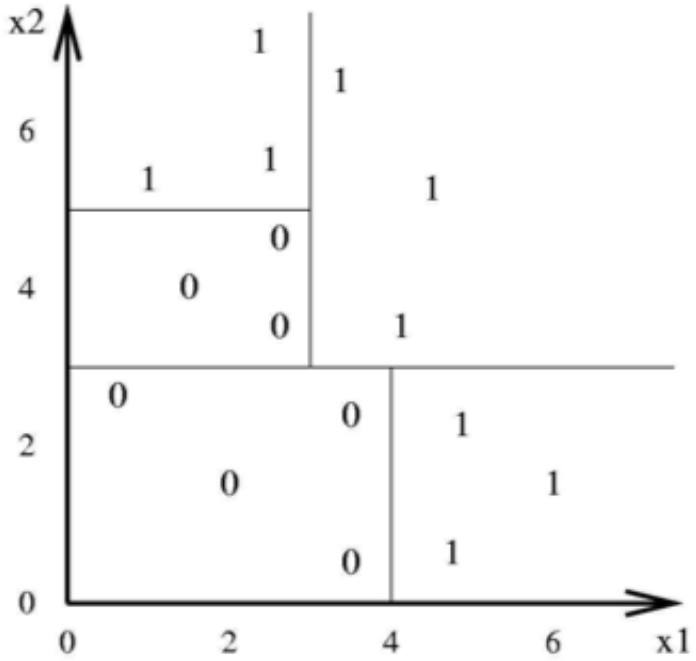
Overcast => Yes
Sunny & Humid => No
Sunny & Normal => Yes
Rain & Strong-Wind => No
Rain & Weak-Wind => Yes

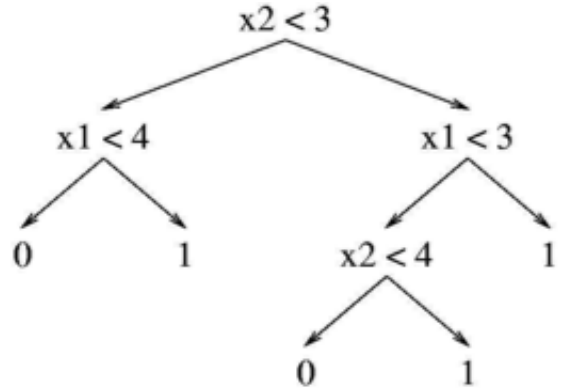


Should I bring an umbrella?

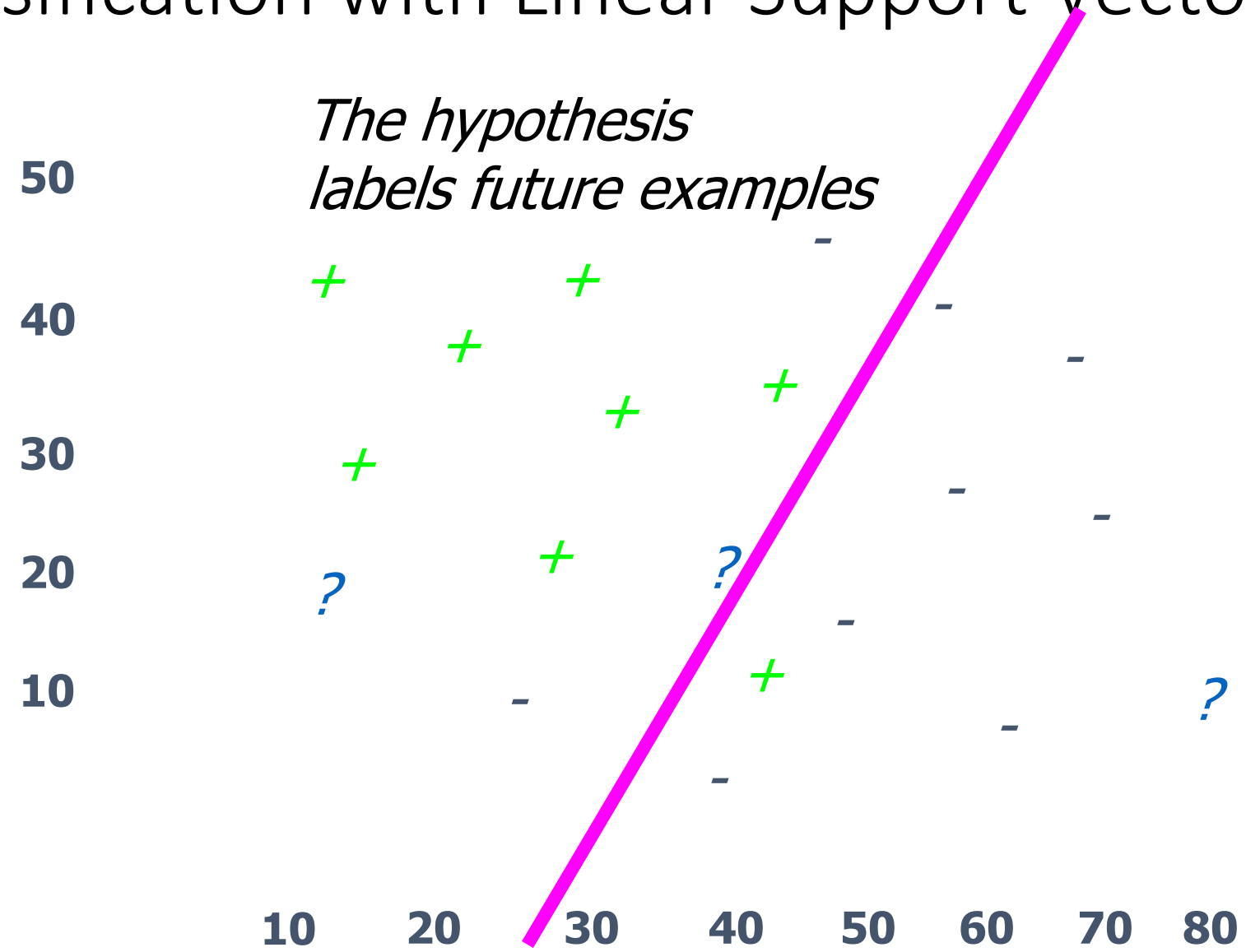
Decision trees

- Why are decision trees good?
  - Can represent any boolean function
  - Can handle discrete & continuous params
  - Easy for humans to **understand**, debug





Classification with Linear Support Vector Machine



Linear SVM for NLU Intent Classification

How do we pick features?
(hint: it's hard)

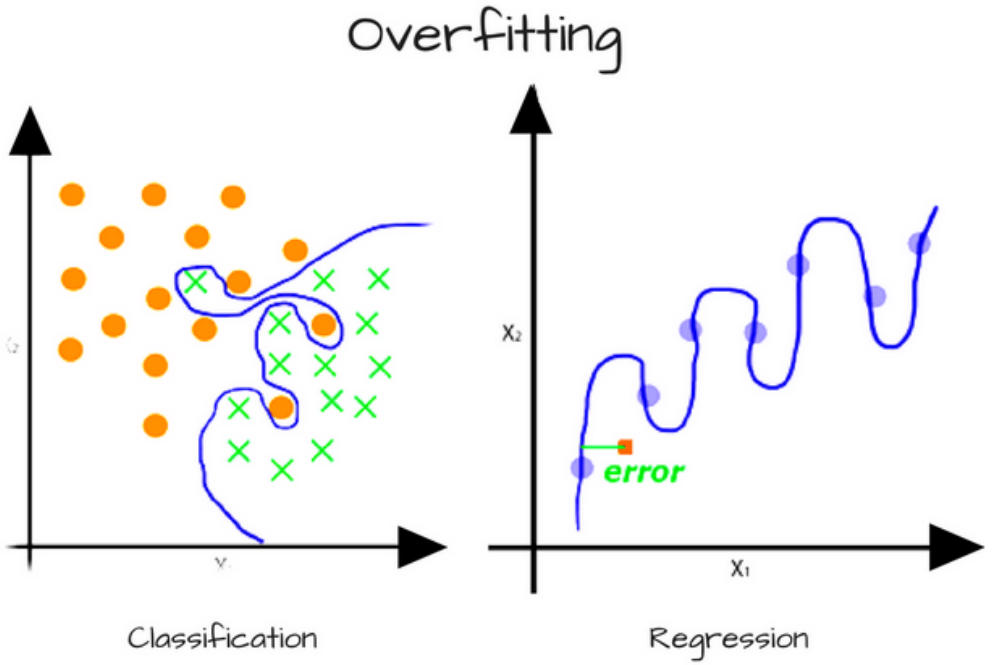
Thought question

- Which one is best?
  - Perpendicular lines?
  - Angled straight lines?
  - Arbitrary curves?

- True or false: "The best learner is the most flexible"

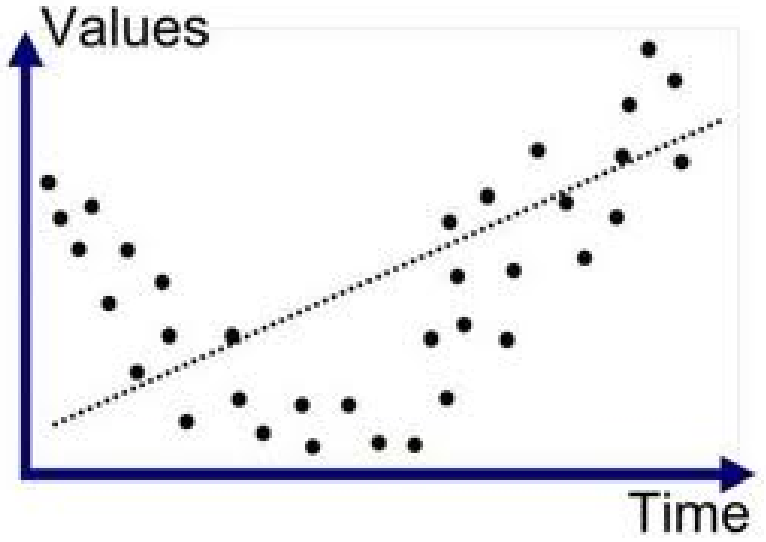
Overfitting

- Overfitting: model describes noise instead of any underlying principle
  - Doesn't generalize well to new data
  - "Model is too complicated"



Underfitting

- Underfitting: model doesn't capture the underlying relationship
  - Doesn't generalize well to new data
  - "Model is too simple"



Performs poorly on both training data and new data

Classification training set

- Training set S = {(**x**, y), …}
  - **x** is a vector of inputs
  - y is the desired output (label) for **x**
  - If **x** describes Outlook, Humidity, Wind then one value of **x** is [Sunny, High, Weak]
  - If y describes “whether or not to carry an umbrella”, then one value of y is “No”
- S is a set of (**x**, y) pairs
  - That is, many examples that describe the weather, plus whether or not to carry an umbrella

Evaluation

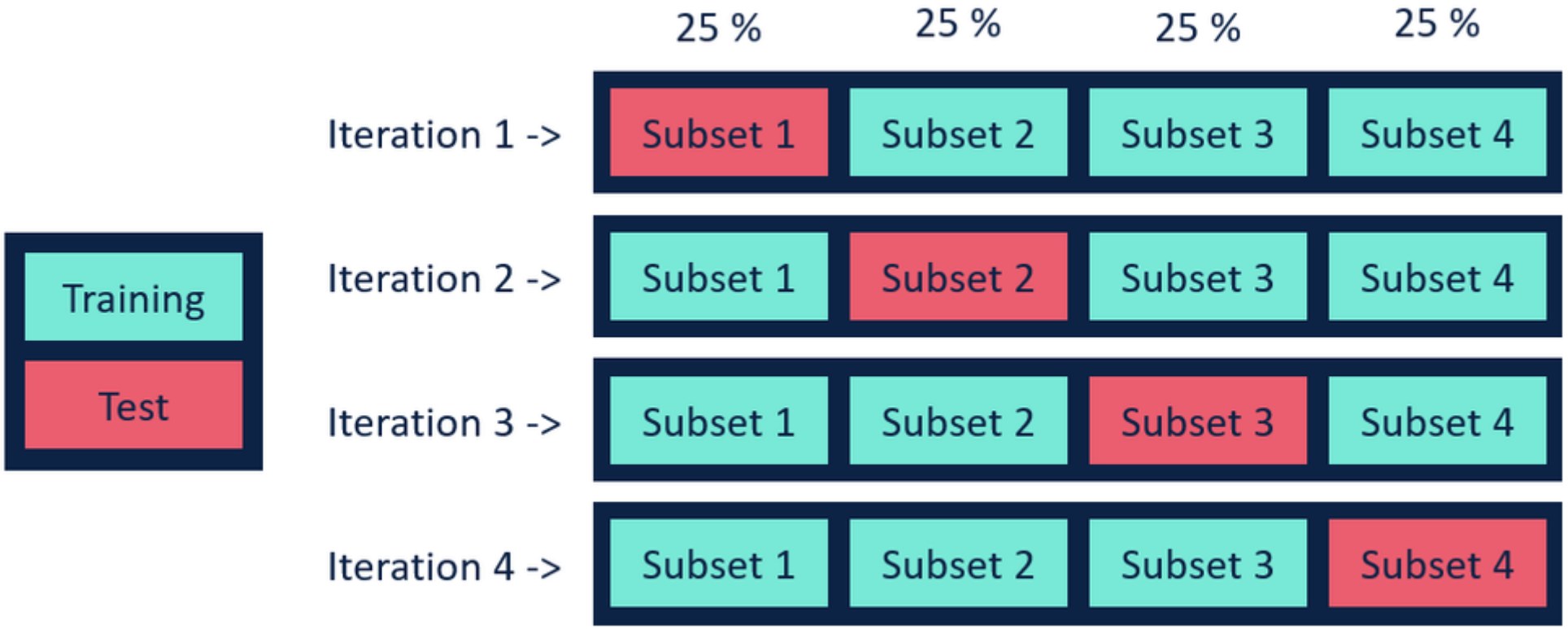
- How do we estimate the performance of classifier on unseen data?
- Can’t just look at accuracy on training data – this will yield an over optimistic estimate of performance
- The test set must be held out during training
- Want to maximize training size, but still get accurate picture of performance
- Lots of data? Use 70/30 train/test split
- Performance == accuracy on test data

Evaluation

- What if you don't have much data?
  - Say, 10 data points?
  - More training data is better, but test set must be representative of future tasks

- Partition examples into *k* disjoint sets
- Now create *k* training sets
  - Each set is union of all equivalent classes ***except one***
  - So each set has (k-1)/k of the original training data

Cross validation



Cross validation

- Leave-one-out
  - Hold out one example, train on remaining
  - Train k learners; average the test results
  - Use if <~100 examples

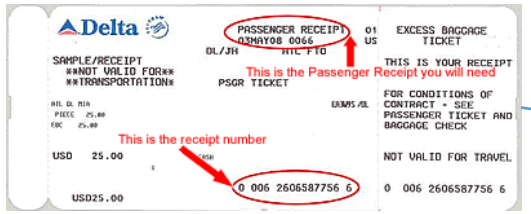
- k-fold cross validation
  - Train k learners; use 1/k of data for test
  - If have ~100-~1000s of examples

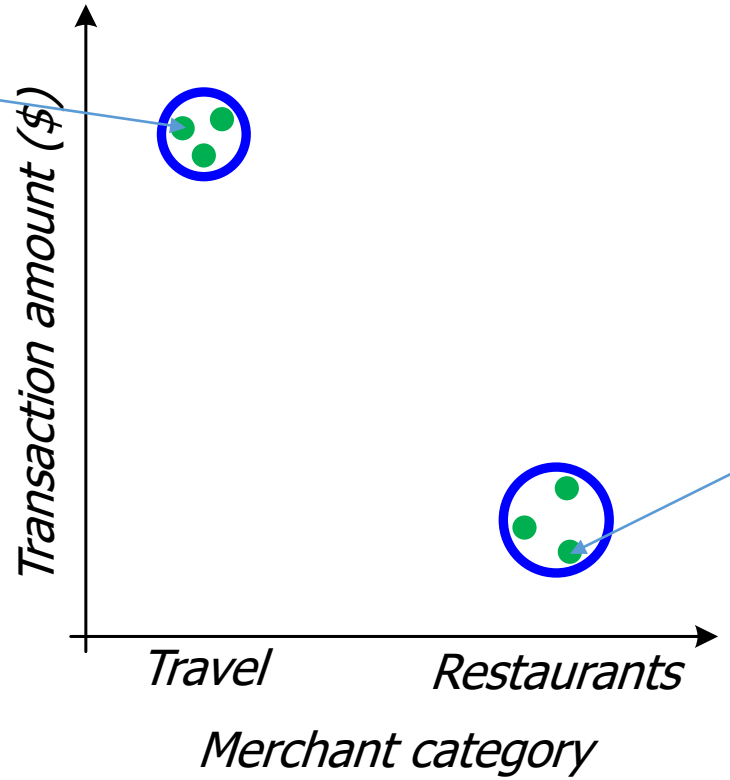
Unsupervised learning

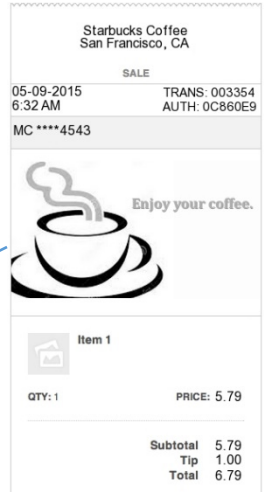
- Common use: **clustering**
- Lots of applications in big data
  - Bioinformatics: group like genes
  - Web: page deduplication, friend management
  - Vision: recognize similar objects
- Can you learn the **structure** of the input data *without* any **labels**?
  - Group together things that are similar
  - Don’t group together things that are dissimilar
- Unsupervised learning depends *a lot* on how you engineer **features**

Unsupervised learning example

- Example: credit card fraud detection

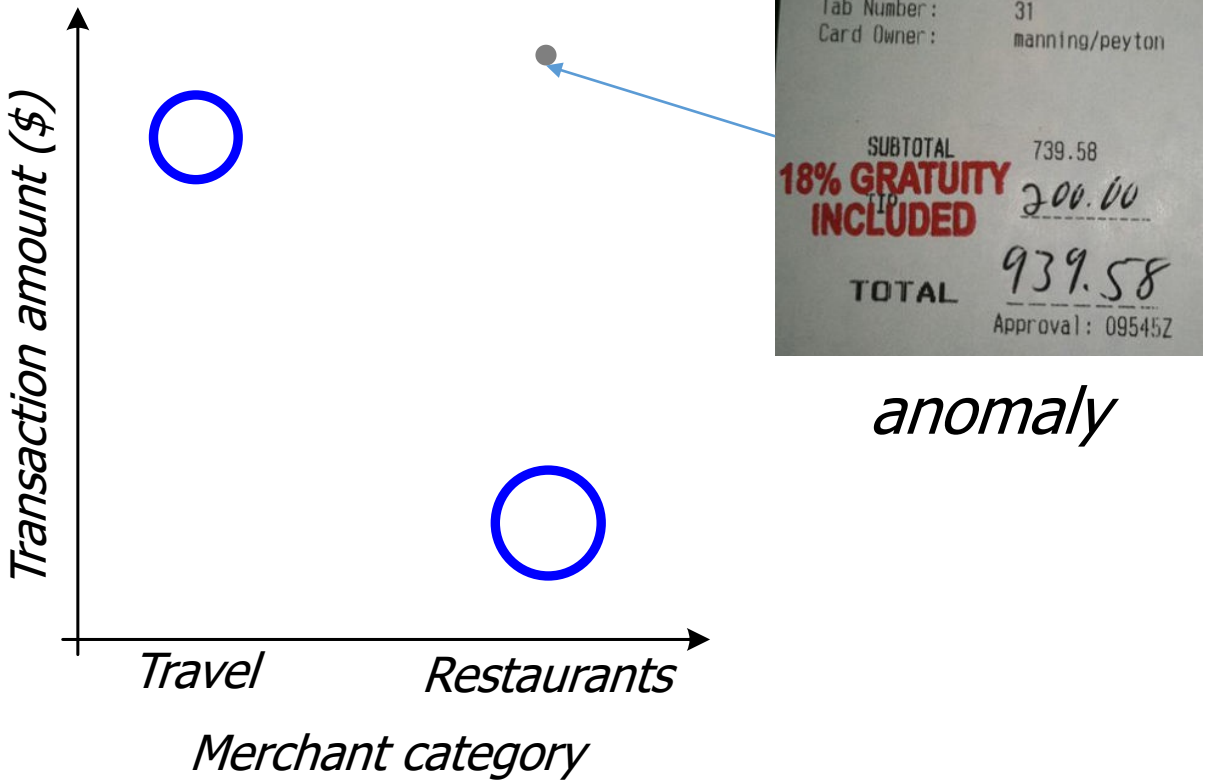




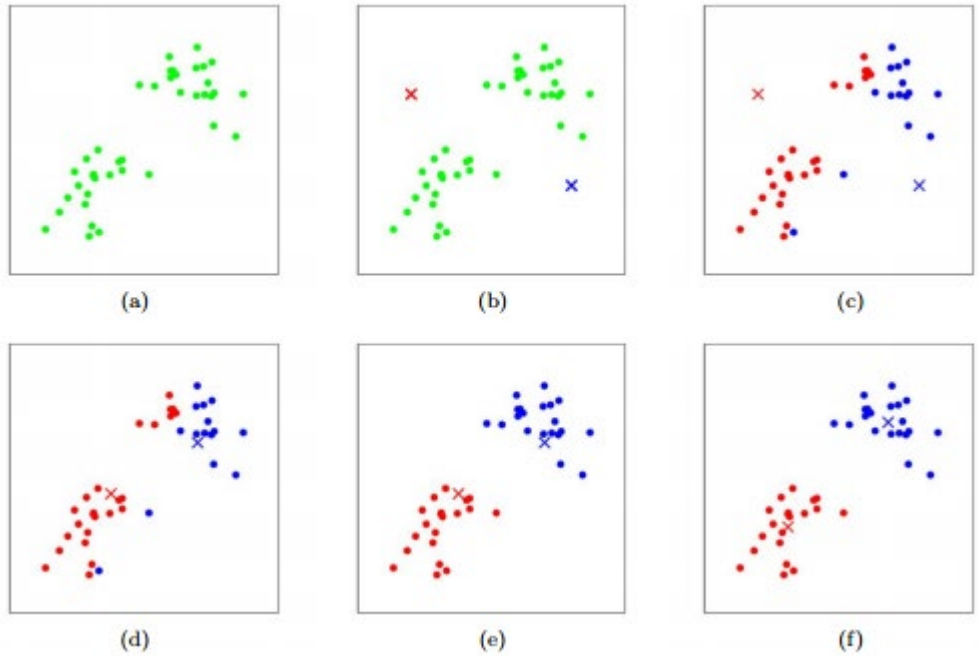


Unsupervised learning example

- Good for anomaly detection



K-means clustering



- Very popular technique, as follows:
  - Grab a distance metric between two points
  - Choose the number of clusters = k
  - Generate k random **centroids**
- Repeat the following:
  - Assign each data item to the closest center
  - Choose new cluster centroids
  - Iterate until centroids converge (i.e., they don't change much)

- In practice, **distance metric** matters more than clustering algorithm

When does k-means fail?

- What if clusters are **oblongs**?
  - Rectangles?
  - Hourglasses?
- What if clusters **overlap**?
  - Document subsets?
  - Image closeups?
- What if clusters are different **sizes**?
  - People cloning wikipedia.org vs. people cloning cafarella.com
  - Consider both volume and # points

How to pick *k*?

- Difficult without domain knowledge
- Agglomerative clustering
  - Start one cluster per example
  - Merge the two closest clusters
  - Repeat until you've got one cluster
  - Output result

- How do you measure cluster closeness?
  - Distance between centroids?
  - Min distance between pairs? (or max?)

Cluster evaluation

- Need the "right" number of "good" clusters
- Correctness of a cluster is easy
  - Do members belong together?
  - Roughly similar to precision
- Testing whether clusters are "right" is harder
- Multiple good clusterings possible for a single dataset
- In general, evaluation is **much** harder than with supervised learning

Important questions

- How do you measure similarity?
- How do you construct the clustering?
- How do you evaluate the outcome?

Cluster similarity measurement

- Euclidean distance (for reals)
- Jaccard distance (for set overlap)
- Bit distance (for vectors of booleans)

- Many others possible, depending on your application
- How would you measure similarity when clustering:
  - Images?
  - Videos?
  - Schemas?

POLICY & LAW | US & WORLD | NET NEUTRALITY

Congress just cleared the way for internet providers to sell your web browsing history

Resolution is now off to the president's desk

by Jacob Kastrenakes | Mar 28, 2017, 5:57pm EDT

By THE ASSOCIATED PRESS MARCH 23, 2017, 6:54 P.M. E.D.T.

NEW YORK — The Senate voted to kill Obama-era online privacy regulations , a first step toward allowing internet providers such as Comcast, AT&T and Verizon to sell your browsing habits and other personal information as they expand their own online ad businesses.

Ethics and machine learning

- Many data mining projects are ethically and politically contentious
  - Credit card offers
  - Financial trades
  - Total Information Awareness project (TIA)
- Many data-mining projects are ethically complicated because of the data used
  - Is the privacy-leaking AOL data OK?
  - What's so wrong about collecting WiFi info?

Summary

- Supervised learning
  - Example: Spam filtering
  - Training data with known correct labels

- Unsupervised learning
  - Example: grouping similar news articles
  - Training data without any labels