Introduction to Data Mining

CS 145

Lecture 1:

Administrivia & Intro

Course Info

- Course homepage:
  - https://piazza.com/ucla/spring2024/cs145/home

- Lectures:
  - Monday & Wednesday 12:00PM – 1:50PM
  - Boelter Hall 3400

Instructor & Teaching Assistant

Instructor: Ziniu Hu (acgbull@gmail.com)

TA: Yanqiao Zhu (yzhu@cs.ucla.edu)

Office hours (tutorials & hackathon for project)

- Friday 12:00 – 1:50 PM (Rolfe Hall 3135)
- Friday 2:00 – 3:50 PM (Royce Hall 162)

Evaluation scheme:

- Assignments: 40%
  - (bonus) Mutual QA: 5%
- Midterm exam: 30%
- Course project: 30%
  - (bonus) Performance: 5%

Grade	Point Range
A+	[97, +∞)
A	[93, 97)
A-	[90, 93)
B+	[87, 90)
B	[83, 87)
B-	[80, 83)
C+	[77, 80)
C	[73, 77)
C-	[70, 73)
D+	[67, 70)
D	[60, 67)
F	[0, 60)

Assignments (40%)

- In total 5 weekly assignments; 10 days to finish each.
  - No late submissions permitted; each student may request a one-day extension for one of the five assignments with no penalty, provided they inform the TA **before the deadline**
- Lowest score dropped; each of remaining four worth 10%
- Most are coding questions in Jupyter notebook

Participation bonus (5%)

- Active participation in answering other students' questions on Piazza earns bonus points

Exams (30%)

In-class exam on May 15 (Wednesday, Week 7)

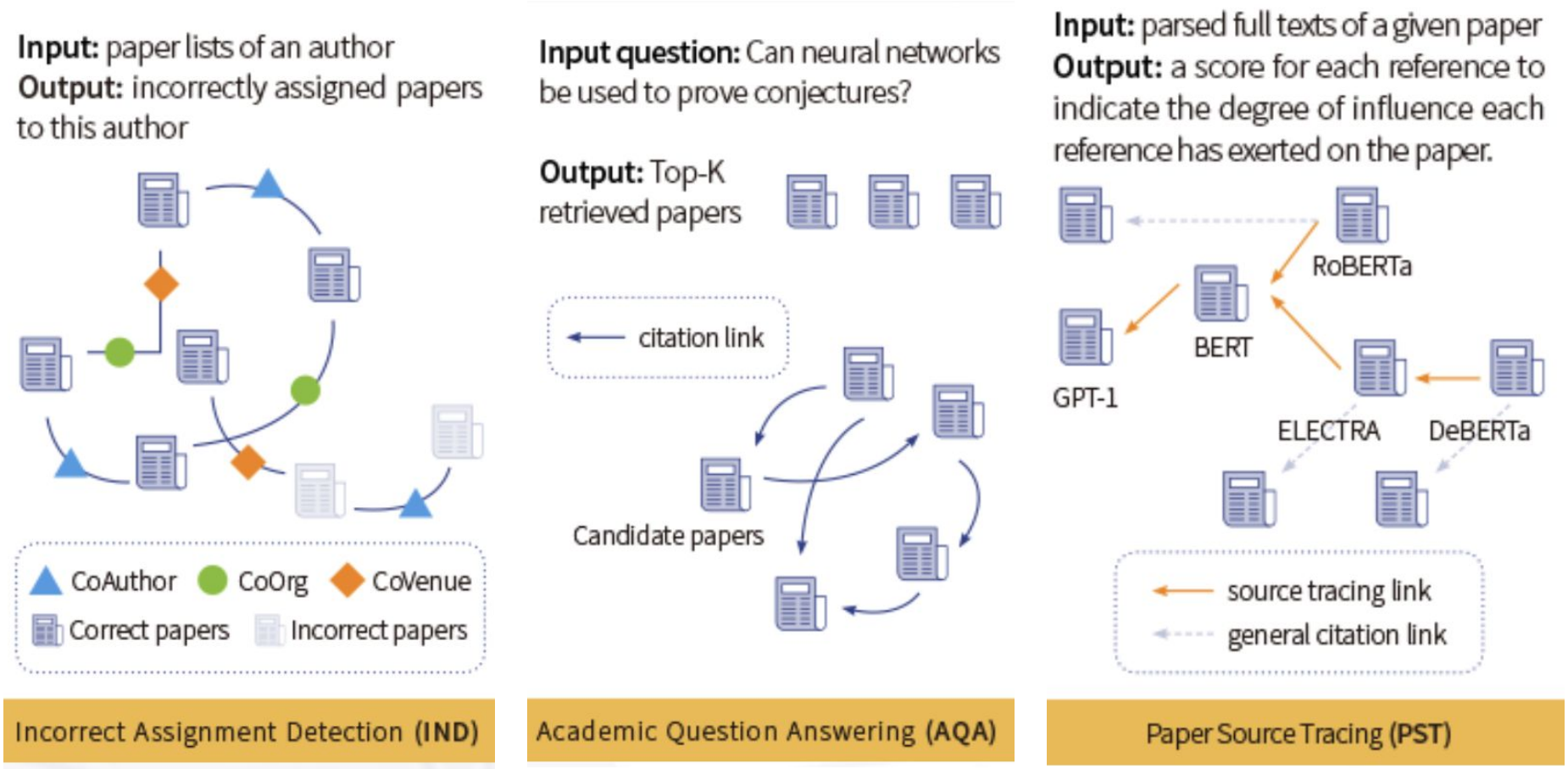
- Exams are in-person only;
- 2 “Cheat” sheets are allowed
- Simple calculators allowed
- Internet access strictly prohibited
- Scope: all topics prior to Week 7

Course project (30% + 5% bonus)

Team work: at most 6 students per group

Scope: KDD Cup 2024 Open Academic Graph Challenge

Week 2	April 12	Team formation due
Week 10	June 10 & 12	In-class project presentation
Week 10	June 12	Report & code submission



- participating teams for IND and PST can obtain a free quota of 1 million tokens for the [GLM-4 API]

Course project (30% + 5% bonus)

Evaluation criteria:

- Code Repo & Doc (10%)
- Final Tech Report (10%)
- Final presentation (5%)
- Mutual Rating (5%)
- Leaderboard Results (5% bonus)

Not necessarily to beat the leaderboard, we mainly evaluate a complete, executable & well-documented code repo, and a detailed tech report (listing all the methods and ablation you've tried)

Course project (30% + 5% bonus)

Evaluation criteria:

- Code Repo & Doc (10%)
- Final Tech Report (10%)
- Final presentation (5%)
- Mutual Rating (5%)
- Leaderboard Results (5% bonus)

For bonus, we will set 5 intervals from a baseline submission to the top one (e.g., if baseline is at rank at 80, then rank 1-16 get 5 bonus; rank 17-32 get 4 bonus, … rank 65-80 get 1 bonus)

Course project (30% + 5% bonus)

Team sign-up form:

Deadline: April 12, 11:59PM

Looking for teammates?

- Use Piazza to collaborate with other classmates and form your teams
- Check the sign-up form and email the team leader to see if you are a good fit

Friday Tutorials & Hackathon

Friday 12:00 – 1:50 PM (Rolfe Hall 3135)

Friday 2:00 – 3:50 PM (Royce Hall 162)

The Friday discussion session is set to be mainly tutorial (Python/Pytorch/coding) and hackathon for the final project.

Instructor and TA will be available to answer course/project related questions, discuss potential ideas, and provide technical support.

And we encourage the whole team to work together~

What Is Data Mining & Knowledge Discovery?

What is **Data** and what is **Knowledge**?

Which technique is used to **mine** and **discovered**?

What Is Knowledge Discovery from Data?

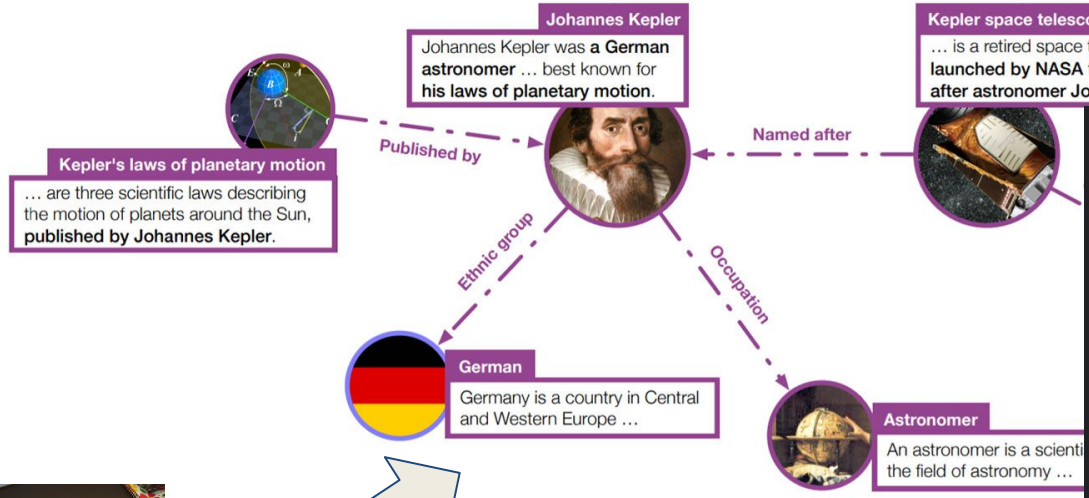
Mine / discover

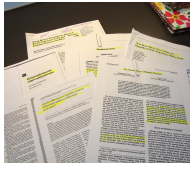
Knowledge

Large amount of data on the web

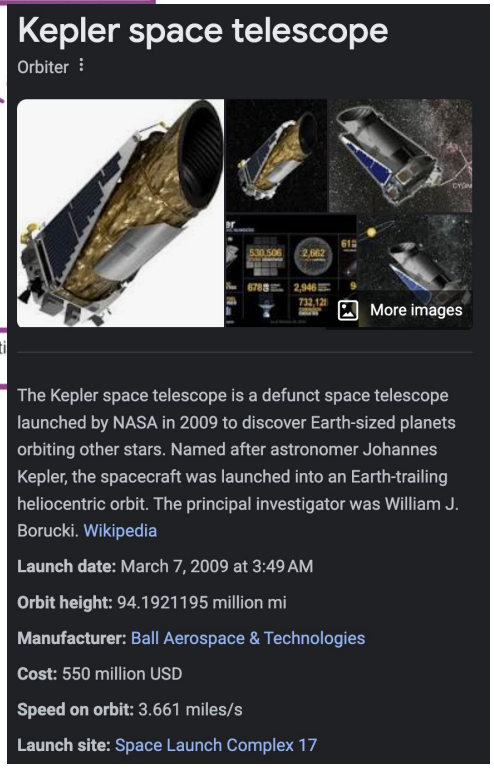
Youtube videos, news, scientific papers, Tiktok, …

Knowledge Discovery from Data

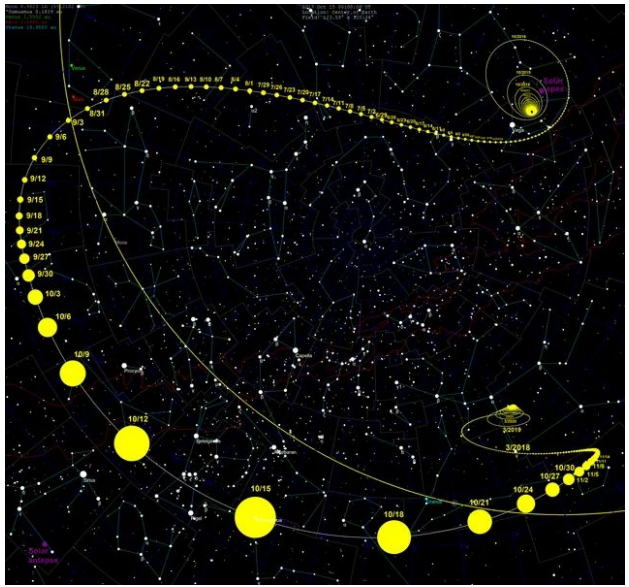


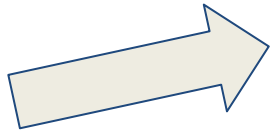


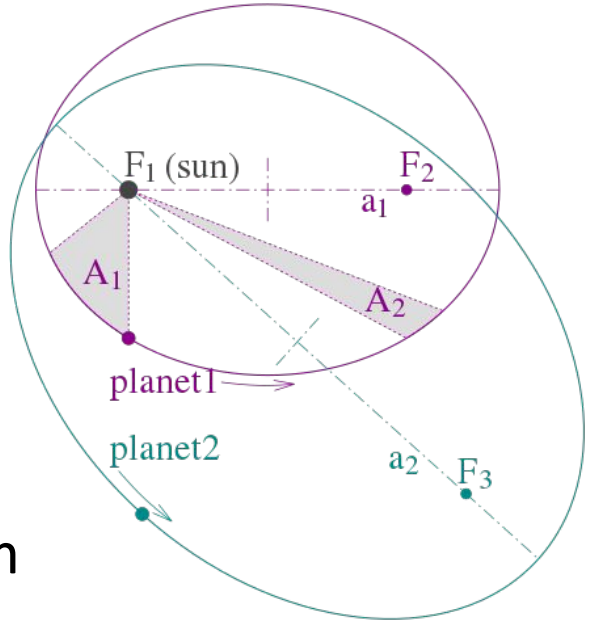
Construct knowledge graph from unstructured text data.



Knowledge Discovery from Data







Deriving symbolic equation / rules from **time-series** observational data (in physics and astronomy)

$$mr\omega^2 = G\frac{mM}{r^2}$$

What Is Data Mining & Knowledge Discovery?

Process of Converting
Unstructured Data Into
Human-Understandable
Knowledge

What Is Data Mining & Knowledge Discovery?

Process of Converting
Unstructured Data Into
Human-Understandable
Knowledge

How?
Typically via
(unsupervised) Learning

Supervised Learning

- Find function from input space *X* to output space *Y*

$$f : X \rightarrow Y$$

such that the prediction is accurate

x

y

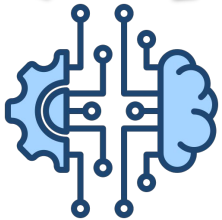
f(x) -> cat

Supervised Learning

Data: X

Target Signal: Y

Logistic Regression
Decision Tree,
Deep Neural Nets (CNN,
Transformer, etc), …



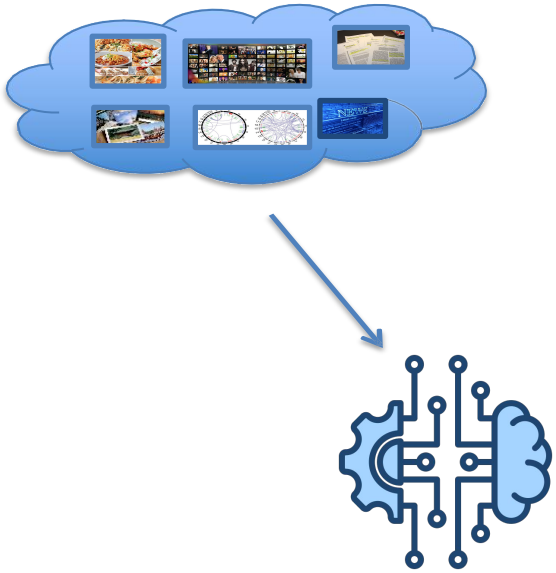
f ()

$f(x) \approx y$

For each (x,y) in X, Y

Aside: Unsupervised Learning

Data: X



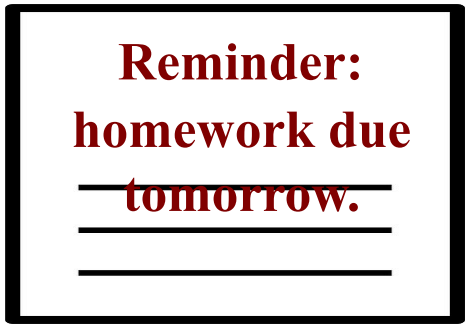
No supervised target!

Directly learn a model to extract “knowledge” from data

Example: Spam Filtering

- **Goal:** write a program to filter spam.

SPAM!



NOT SPAM

SPAM!

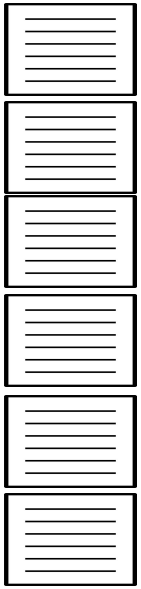
Why is Spam Filtering Hard?

- Easy for humans to recognize
- Hard for humans to write down complete **rule**
- Lots of IF statements!

```
FUNCTION SpamFilter(string document)
{
        IF(“Viagra” in document)
                RETURN TRUE
        ELSE IF(“NIGERIAN PRINCE” in document)
                RETURN TRUE
        ELSE IF(“Homework” in document)
                RETURN FALSE
        ELSE
                RETURN FALSE
        END IF
```

Machine Learning to the Rescue!

Training Set



SPAM!

SPAM!

NOT

SPAM

NOT

SPAM

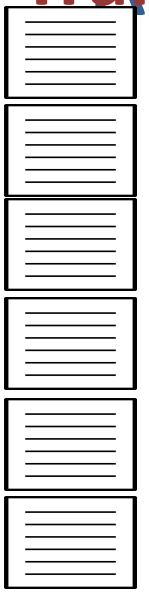
Create data **Feature / Representation**

Choose a **Classification Model**

Train model by **Learning Algorithm**

Labeled by Humans (“Supervision”)

Bag of Words Representation



Training Set

SPAM!

SPAM!

NOT

SPAM

NOT

SPAM

One "feature vector" for whether each word in the vocabulary appear in the sentence

E.g., x = (1,2,0,0,0,1)

Means word-1 appears once

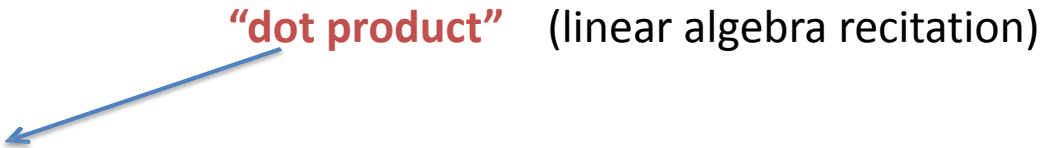
Word-2 appears twice

……

Linear Models

Let x denote the bag-of-words for an email

E.g., x = (1,2,0,0,0,1)



Linear Classifier:

$$f(x|w,b) = \text{sign}(w^T x - b)$$

$$= \text{sign}(w_1 * x_1 + \ldots w_6 * x_6 - b)$$

Formal Definitions

- Training set: $S = \{(x_i, y_i)\}_{i=1}^{N}$ $x \in R^D$ $y \in \{-1, +1\}$
- Model class: $f(x \mid w, b) = w^T x - b$ **Linear Models**

aka hypothesis class

- **Goal:** find (w,b) that predicts well on S.
  - How to quantify "well"?

Basic Supervised Learning Recipe

- Training Data: $S = \{(x_i, y_i)\}_{i=1}^{N}$ $x \in R^D$, $y \in \{-1, +1\}$
- Model Class: $f(x \mid w, b) = w^T x - b$ **Linear Models**
- Loss Function: $L\left(y_i, f(x_i \mid w, b)\right)$ **Squared Loss**
- Learning Objective: $\underset{w,b}{\operatorname{argmin}} \sum_{i=1}^{N} L\left(y_i, f(x_i \mid w, b)\right)$

Optimization Problem

Loss Function

Describe the gap between model's prediction f(x)

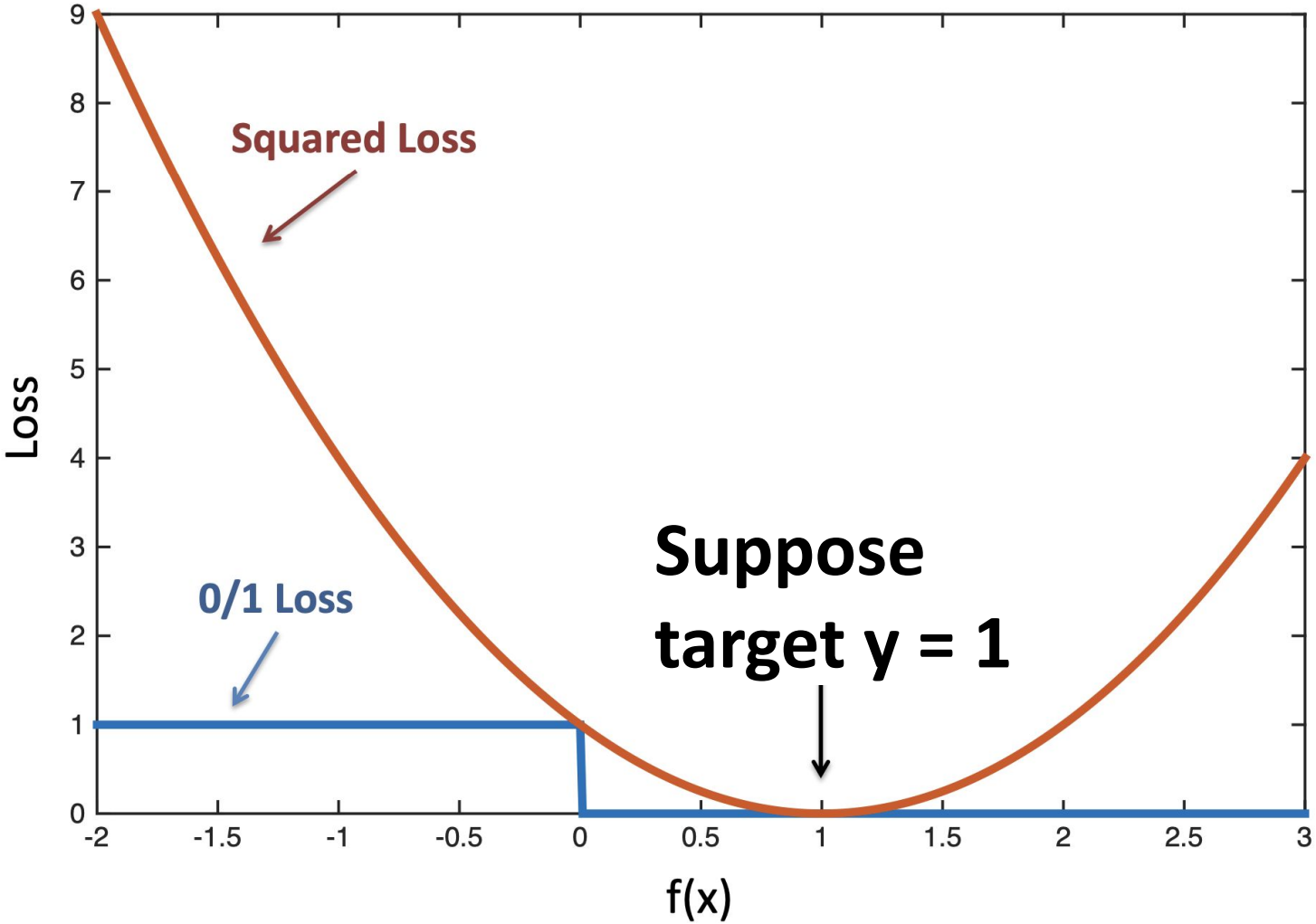
to ground-truth label y

E.g.

0/1 Loss: L(x,y; f) = 1 if sign(f(x)) != y; else 0

[prediction is incorrect]

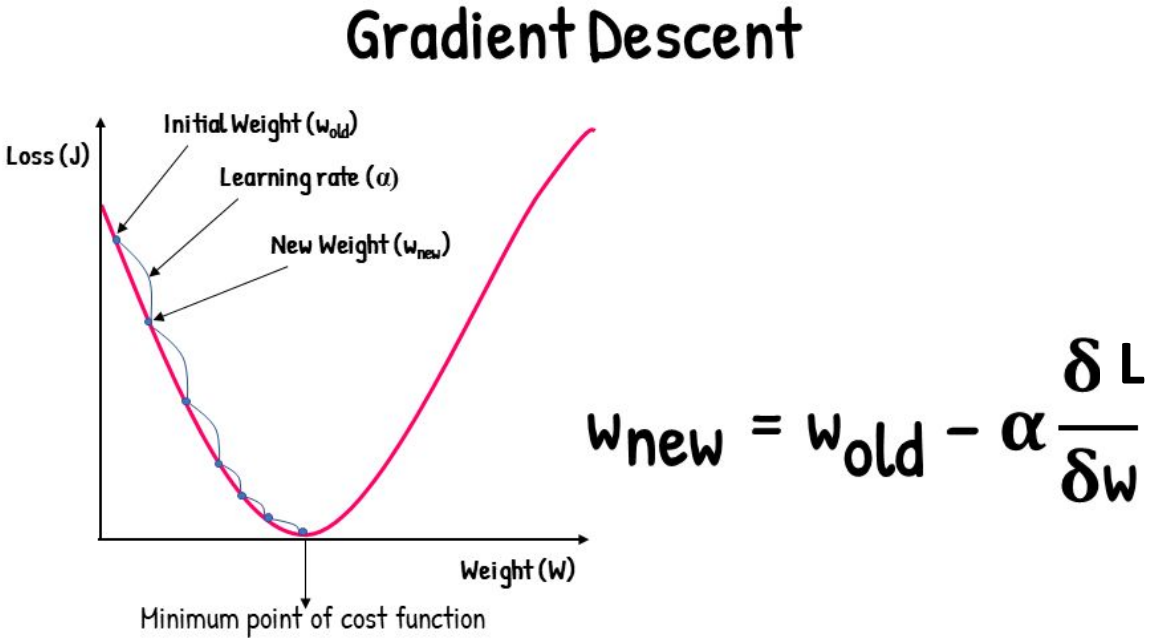
Square Loss: L(x, y; f) = (f(x) - y)^2


Squared Loss
0/1 Loss
Suppose
target y = 1
Loss
f(x)
9
8
7
6
5
4
3
2
1
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
3

Learning Algorithm

$$\underset{w,b}{\operatorname{argmin}} \sum_{i=1}^{N} L\left(y_i, f(x_i \mid w,b)\right)$$

- Typically, requires optimization algorithm.



Learning Algorithm

$$\underset{w,b}{\operatorname{argmin}} \sum_{i=1}^{N} L\left(y_i, f(x_i \mid w,b)\right)$$

- $f(x \mid w,b) = w^T x - b$ **Linear Models** .

- Simplest: **Gradient Descent**

Loop for T iterations

$$w_{t+1} \leftarrow w_t - \partial_w \sum_{i=1}^{N} L\left(y_i, f(x_i \mid w_t,b_t)\right)$$

$$b_{t+1} \leftarrow b_t - \partial_b \sum_{i=1}^{N} L\left(y_i, f(x_i \mid w_t,b_t)\right)$$

Recap: Basic Supervised Learning

- Training Data: $S = \{(x_i, y_i)\}_{i=1}^{N}$ $x \in R^D$ $y \in \{-1, +1\}$

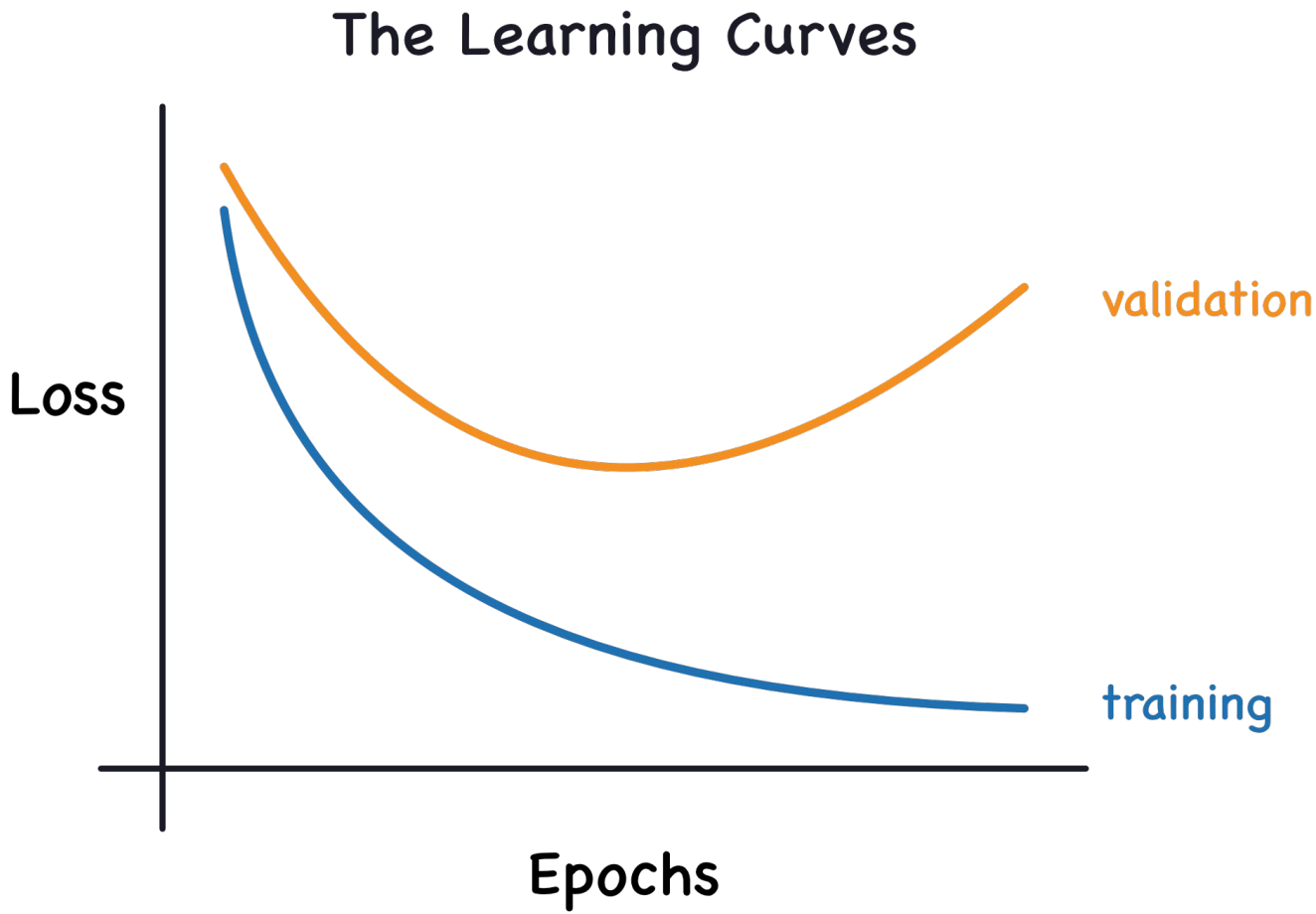
  Sometimes we need a pre-defined **feature engineer** to get proper x (e.g. bag-of-word) from raw data

- Model Class: $f(x \mid w, b) = w^T x - b$ **Linear Models**

- Loss Function: $L\big(y_i, f(x_i \mid w, b)\big)$ **Squared Loss**

- Learning Objective: $\underset{w,b}{\operatorname{argmin}} \sum_{i=1}^{N} L\big(y_i, f(x_i \mid w, b)\big)$

  Optimization Problem


The Learning Curves
Loss
validation
training
Epochs

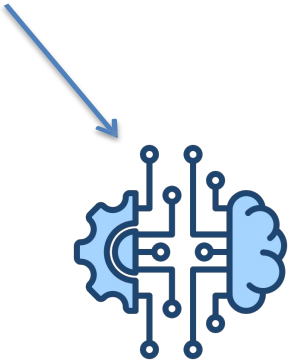
Why Does Machine Learning Work?

- Repeated patterns in the data
  - Typically in the features
  - E.g., “Nigerian Prince” is indicative of spam

- Machine learning will find those patterns
  - Linear model over features (wi * xi)
  - E.g., high weight on the pattern “Nigerian Prince”

Aside: Unsupervised Learning

Data: X

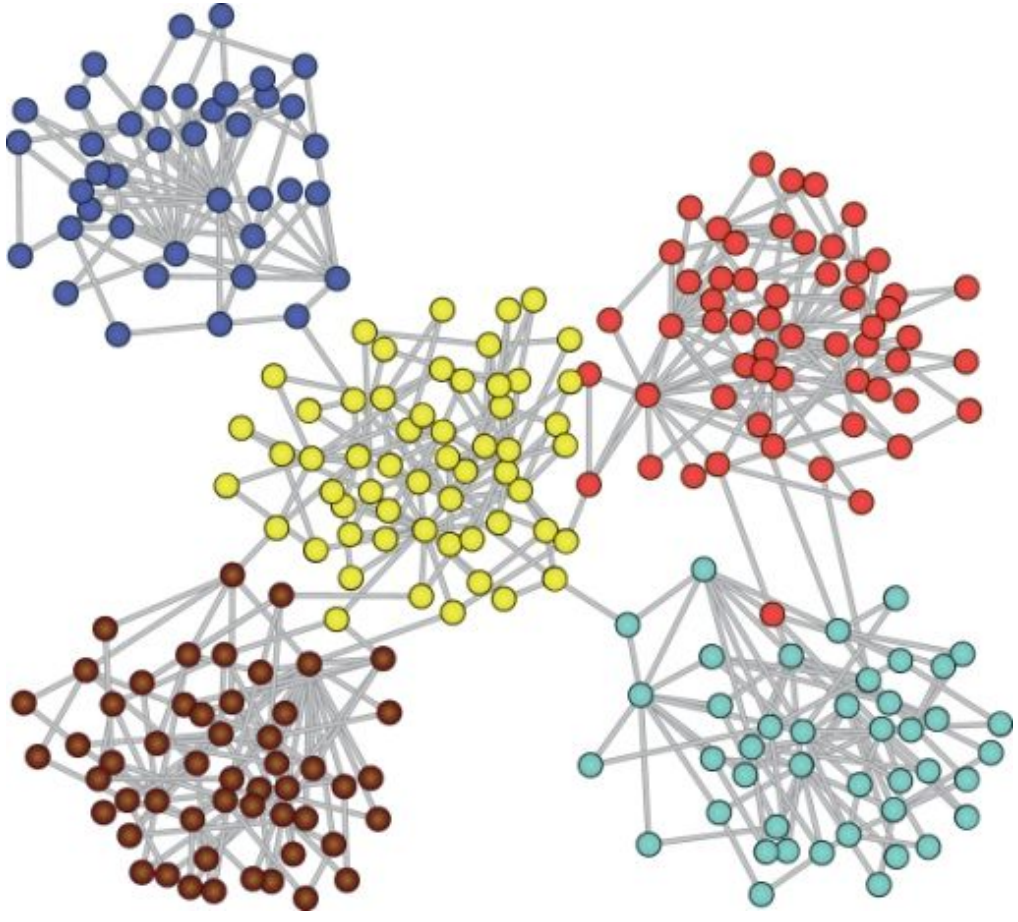


No supervised target!

Loss Functio $L(y_i, f(x_i \mid w, b))$

Learning objective is usually to reconstruction / compression of data (e.g., model P(x)).

Example: Community Detection



Input is a social network

Output is five sub-groups, each with a unique label (e.g., people with different interests)

Core idea: build a model to reconstruct links.

Example: Contrastive Learning

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$$

Create two sub-component of same data

Maximize similarity of in-data samples, contrastive to cross-data pairs.

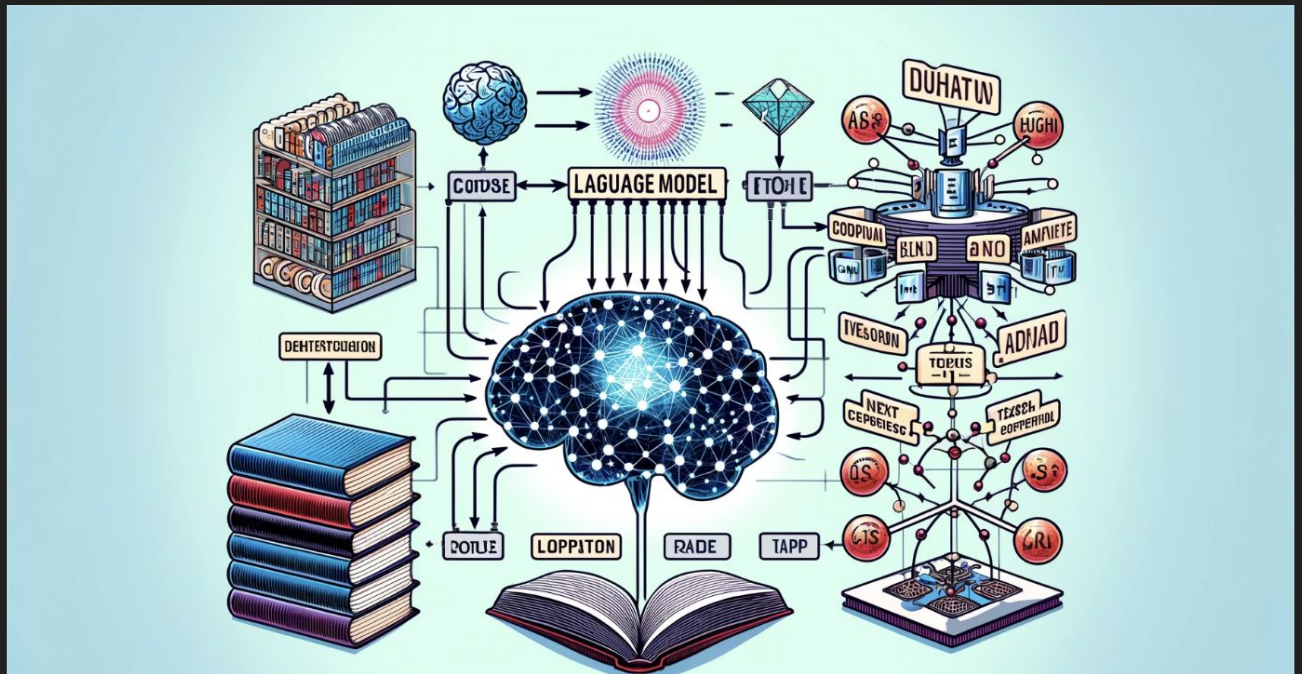
Example: Large Language Model (GPT)

You

Draw a procedural figure to build GPT (from left to right), starting from text corpus on the left, language model at the middle, output a GPT model at the right.

< 13 / 13 >

ChatGPT



Language Models

- Language Model
  - P(X): calculate probability of language

Sentence/Document

  - Example: P(movie was great) 😊

- Auto-regressive LM
  - $P(X) = \prod_1^I P(x_{i+1}|x_1, \cdots, x_i)$:

Next Token Context

  - calculate probability of next token

Image source: Transformer: A General Framework from Machine Translation to Others

Language Models

- Language Model
  - P(X): calculate probability of language

Sentence/Document

  - Example: P(movie was great) 😊

- Auto-regressive LM
  - $P(X) = \prod_1^I P(x_{i+1}|x_1, \cdots, x_i)$:

Next Token Context

  - calculate probability of next token

  - Example: "the movie was great"

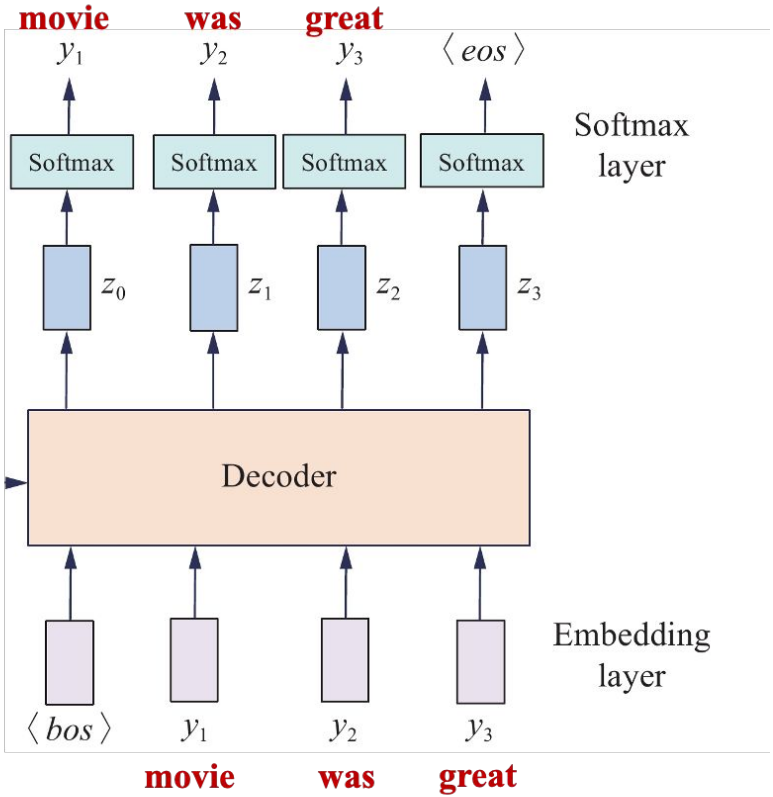
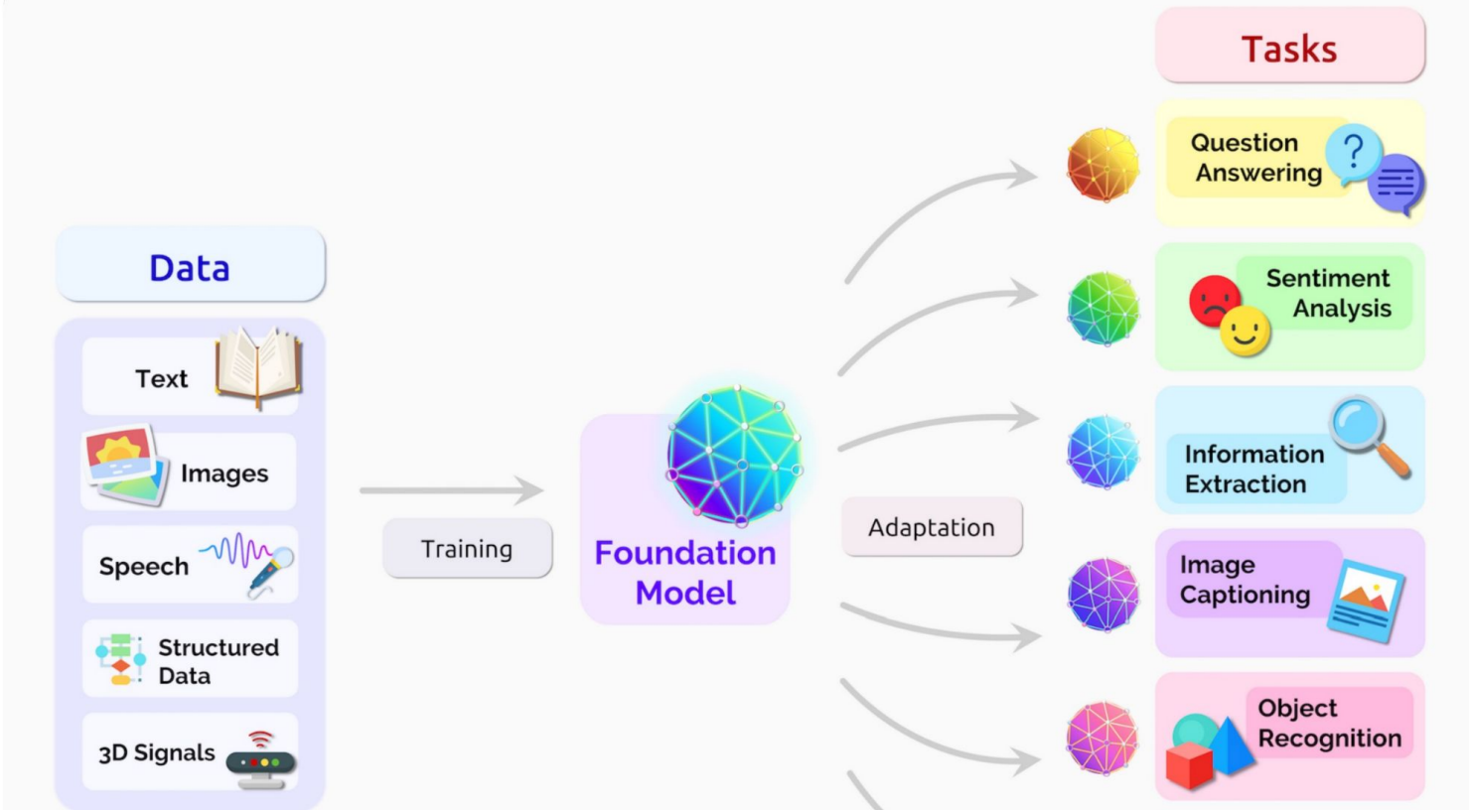


Image source: Transformer: A General Framework from Machine Translation to Others



Pre-training language model (and other foundation model) normally use **large-scale high-quality unlabelled** data

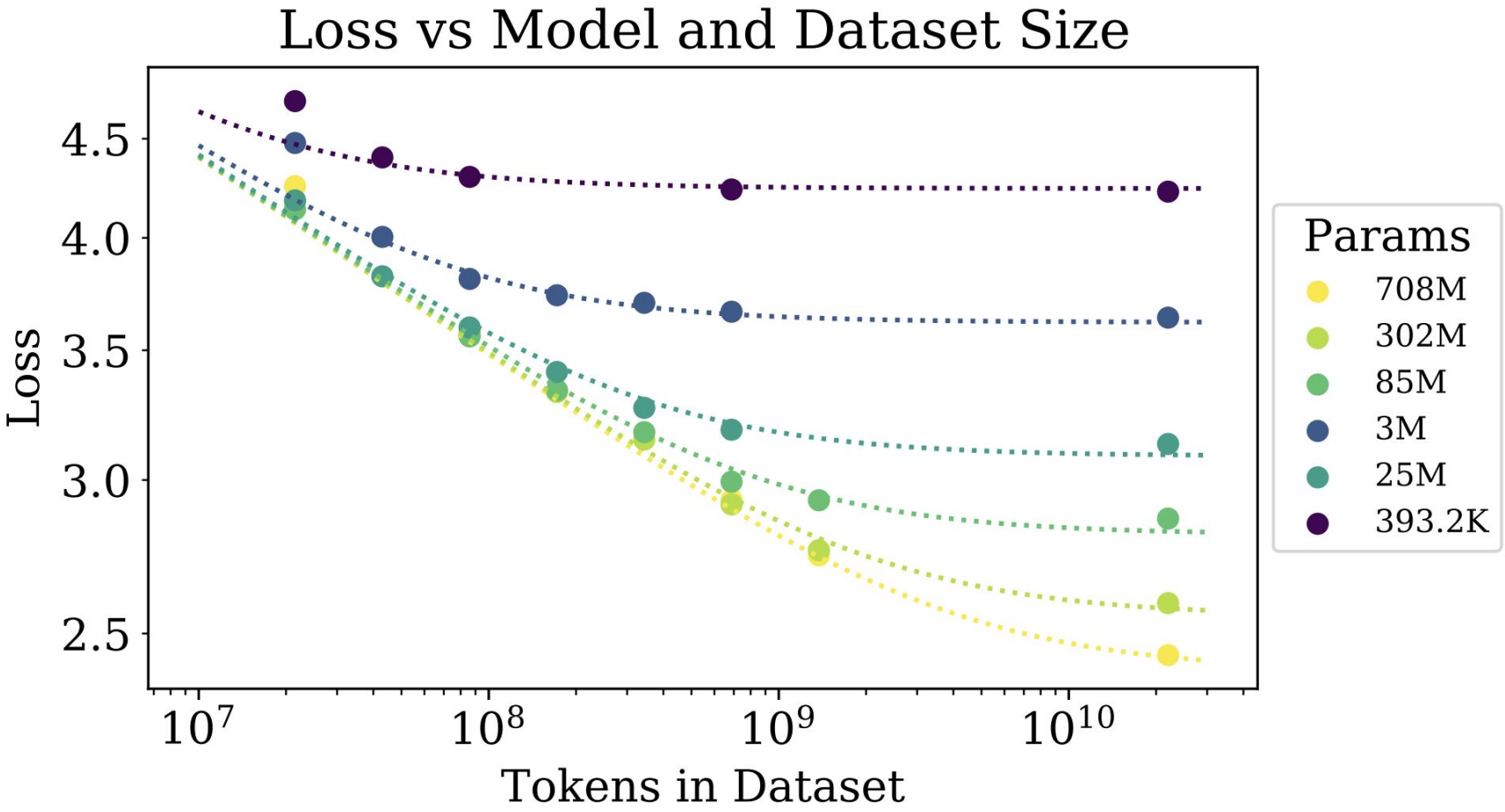


Figure from Scaling Laws for Neural Language Models

Week 1	4/1	**Administrivia & Intro**	
	4/3	**Linear Regression & Backpropagation**	HW 1 out
Week 2	4/8	Logistic Regression, MLP & Evaluation	
	4/10	Regularization, Lasso	HW 2 out
Week 3	4/15	Decision Trees, Bagging, Random Forests	Team formation due
	4/17	Boosting, Ensemble Selection, MoE	HW 3 out
Week 4	4/22	Clustering (K-Means, GMM), Dimensionality Reduction	
	4/24	Latent Factor Models, Matrix Factorization	HW 4 out
Week 5	4/29	Discrete Representation Learning (VAE, VQ-VAE)	
	5/1	Graph and Network: Random Walk	HW 5 out

Week 6	5/6	Graph and Network: Label Propagation and Spectral Clustering	
	5/8	Word Vectors and Seq2Seq Language Models	
Week 7	5/13	Transformers	
	5/15	**In-class exam**	
Week 8	5/27	*Memorial Day (No Class)*	
	5/29	Pre-Training	
Week 9	6/3	Post-training (SFT, RLHF)	
	6/5	LLM Planning and Reasoning	
Week 10	6/10	**Project Presentation**	
	6/12	**Project Presentation**	

Basic Statistical Descriptions of Data

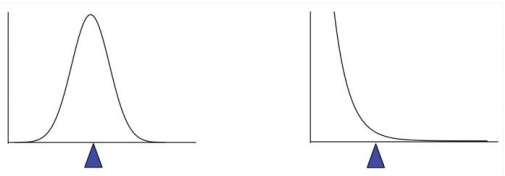
Central Tendency

Dispersion of the Data

Sample mean

- The **mean** of a set of *n* observations of a variable is denoted $\bar{x}$ and is defined as:

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n} = \frac{1}{n}\sum_{i=1}^{n} x_i$$



- The mean describes what a "typical" sample value looks like, or where is the "center" of the distribution of the data.
- Key theme: there is always uncertainty involved when calculating a sample mean to estimate a population mean.

Sample median

The **median** of a set of n number of observations in a sample, ordered by value, of a variable is is defined by

$$\text{Median} = \begin{cases} x_{(n+1)/2} & \text{if } n \text{ is odd} \\ \frac{x_{n/2} + x_{(n+1)/2}}{2} & \text{if } n \text{ is even} \end{cases}$$

Example (already in order):

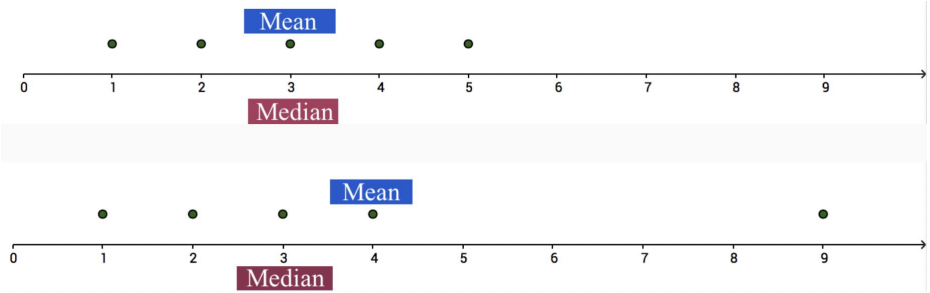
Ages: 17, 19, 21, 22, 23, 23, 23, 38

Median = (22+23)/2 = 22.5

The median also describes what a typical observation looks like, or where is the center of the distribution of the sample of observations.

Mean vs. Median

The mean is sensitive to extreme values (**outliers**)



Measures of Spread: Variance

- The (sample) **variance**, denoted s^2, measures how much on average the sample values deviate from the mean:

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n} |x_i - \bar{x}|^2$$

- Note: the term $|x_i - \bar{x}|$ measures the amount by which each x_i deviates from the mean $\bar{x}$. Squaring these deviations means that s^2 is sensitive to extreme values (outliers).
- Note: s^2 doesn't have the same units as the x_i :(
- What does a variance of 1,008 mean? Or 0.0001?

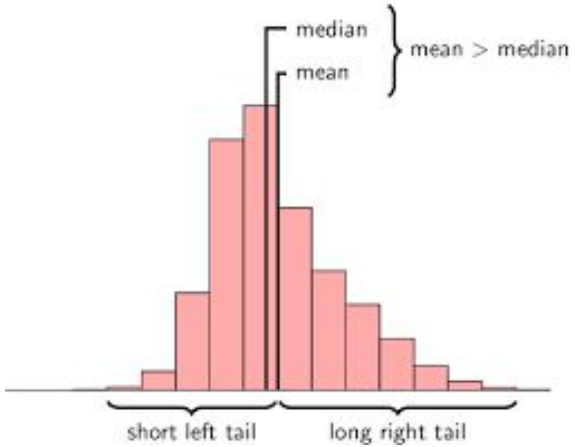
Measures of Spread: Standard Deviation

- The (sample) **standard deviation**, denoted s, is the square root of
- the variance

$$s = \sqrt{s^2} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} |x_i - \bar{x}|^2}$$

- Note: s does have the same units as the x_i. Phew!

Mean, median, and skewness

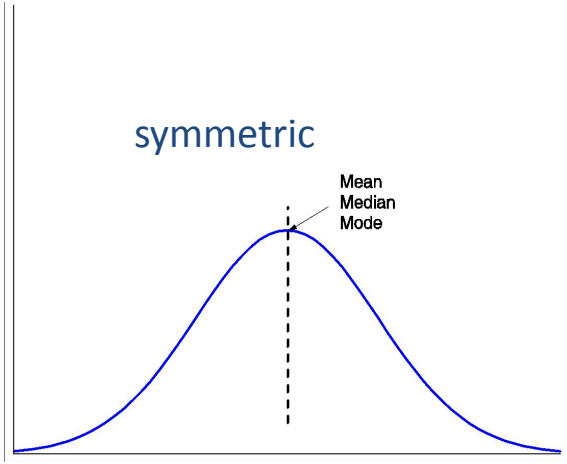


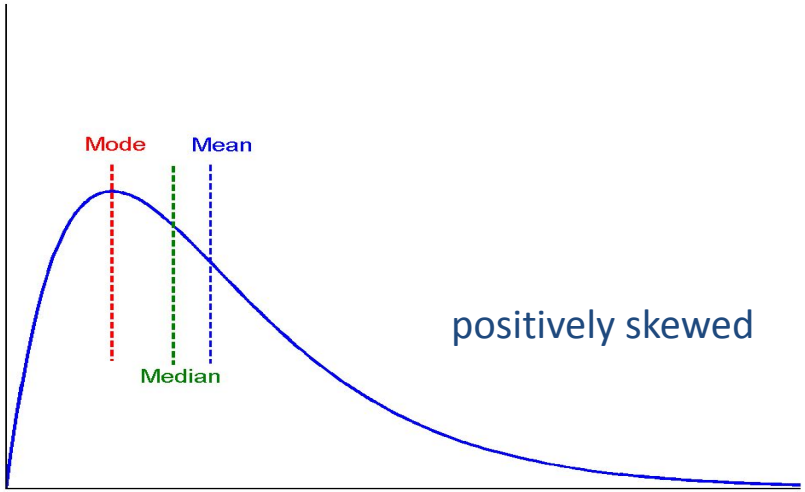
$$\gamma_1 := \tilde{\mu}_3 = \mathrm{E}\left[\left(\frac{X - \mu}{\sigma}\right)^3\right]$$

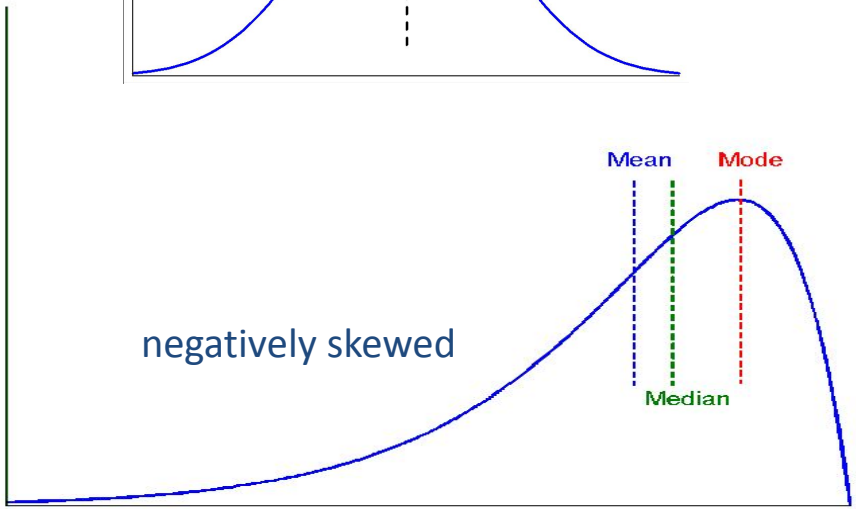
The above distribution is called **right-skewed** since the mean is greater than the median. Note: **skewness** often "follows the longer tail".

Symmetric vs. Skewed Data

Median, mean and mode of symmetric, positively and negatively skewed data





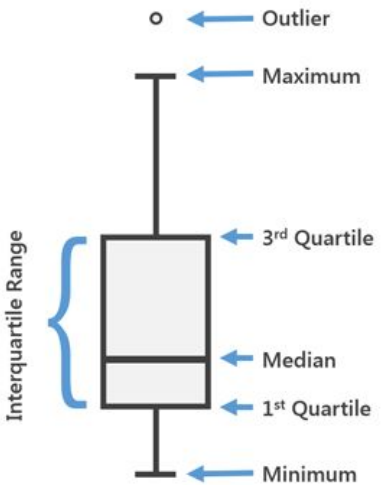


Question

Is income positively or negatively skewed?

Measuring the Dispersion of Data

- Quartiles, outliers and boxplots
  - **Quartiles**: Q_1 (25th percentile), Q_3 (75th percentile)
  - **Inter-quartile range**: IQR = $Q_3 - Q_1$
  - **Five number summary**: min, Q_1, median, Q_3, max
  - **Outlier**: usually, a value higher/lower than 1.5 x IQR of Q_3 or Q_1



Box Plot

- Variance and standard deviation (*sample: s, population: σ)*
  - **Variance**: (algebraic, scalable computation)
    - $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}[\sum_{i=1}^{n} x_i^2 - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2]$
    - $\sigma^2 = E[(X - E(X))^2] = E(X^2) - (E(X))^2$
  - **Standard deviation** *s (or σ)* is the square root of variance s^2 *(or* σ^2*)*

Anscombe's Data

The following four data sets comprise the Anscombe's Quartet; all four sets of data have identical simple summary statistics.

G. E. M. Anscombe

FBA

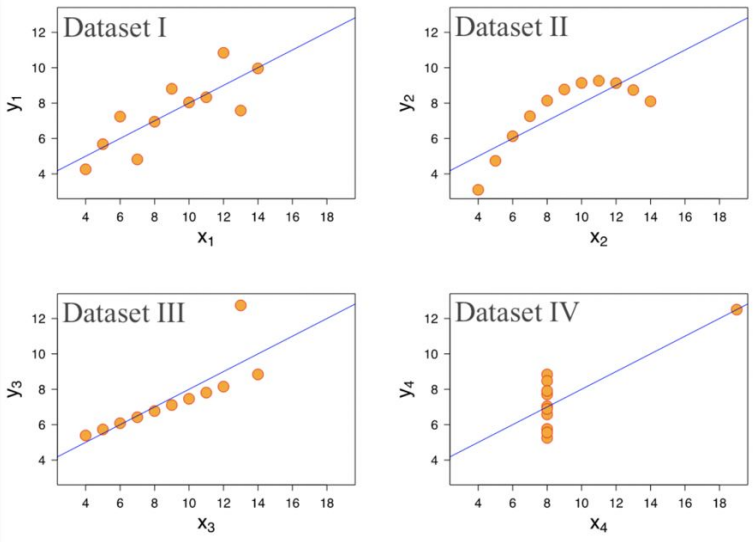
Anscombe as a young woman

Born	Gertrude Elizabeth Margaret Anscombe 18 March 1919 Limerick, Ireland
Died	5 January 2001 (aged 81) Cambridge, England

	Dataset I		Dataset II		Dataset III		Dataset IV	
	x	y	x	y	x	y	x	y
	10	8.04	10	9.14	10	7.46	8	6.58
	8	6.95	8	8.14	8	6.77	8	5.76
	13	7.58	13	8.74	13	12.74	8	7.71
	9	8.81	9	8.77	9	7.11	8	8.84
	11	8.33	11	9.26	11	7.81	8	8.47
	14	9.96	14	8.1	14	8.84	8	7.04
	6	7.24	6	6.13	6	6.08	8	5.25
	4	4.26	4	3.1	4	5.39	19	12.5
	12	10.84	12	9.13	12	8.15	8	5.56
	7	4.82	7	7.26	7	6.42	8	7.91
	5	5.68	5	4.74	5	5.73	8	6.89
Sum:	99.00	82.51	99.00	82.51	99.00	82.51	99.00	82.51
Avg:	9.00	7.50	9.00	7.50	9.00	7.50	9.00	7.50
Std:	3.32	2.03	3.32	2.03	3.32	2.03	3.32	2.03

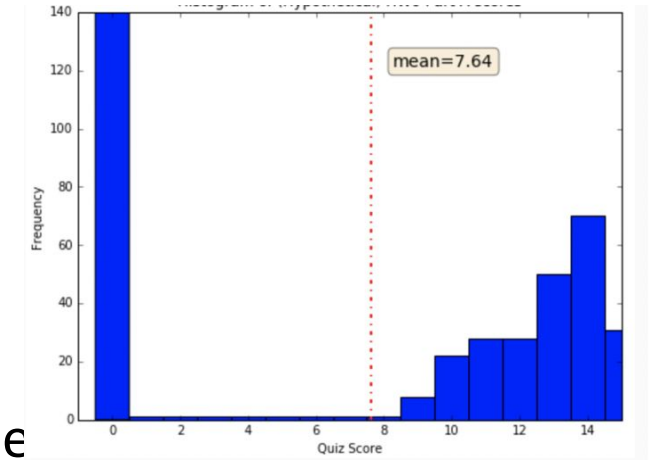
Anscombe's Data (cont.)

Summary statistics clearly don't tell the story of how they differ. But a picture can be worth a thousand words:



More Visualization Motivation

If I tell you that the average score for a Homework is: 7.64/15 = 50.9%, what does that suggest?



And what doe

Types of Visualizations

What do you want your visualization to show about your data?

Distribution: how a variable or variables in the dataset distribute over a range of possible values.

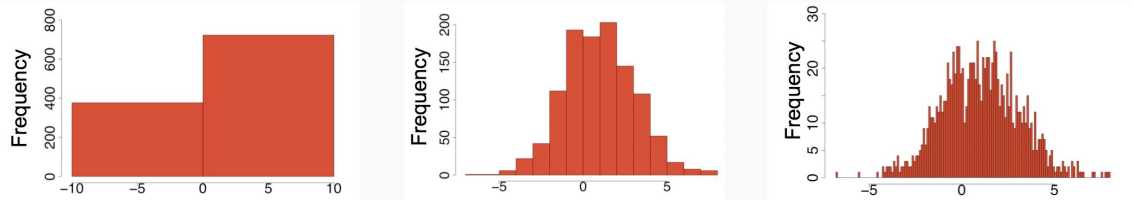
Relationship: how the values of multiple variables in the dataset relate

Composition: how the dataset breaks down into subgroups

Comparison: how trends in multiple variable or datasets compare

Histograms to visualize distribution

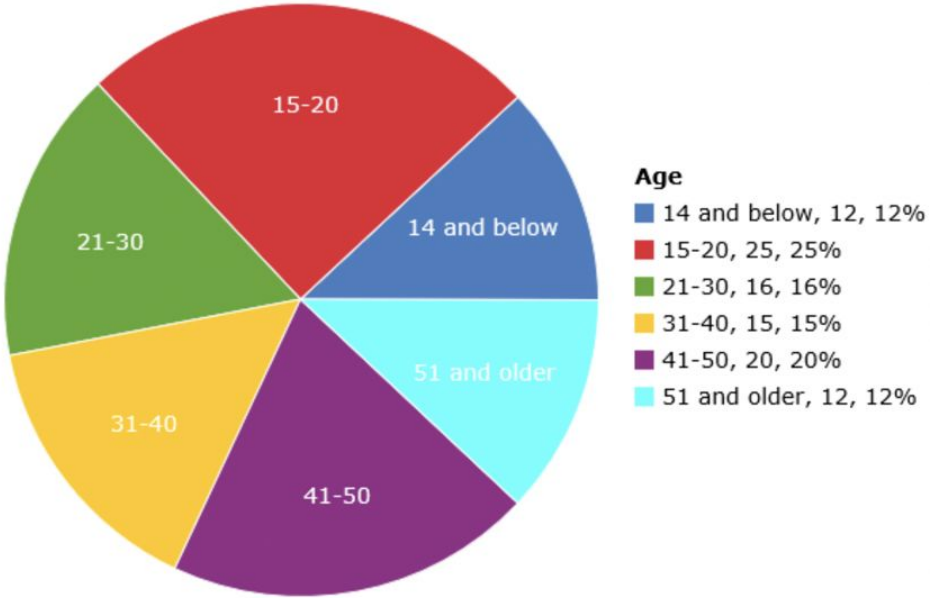
A **histogram** is a way to visualize how 1-dimensional data is distributed across certain values.



Note: Trends in histograms are sensitive to number of bins.

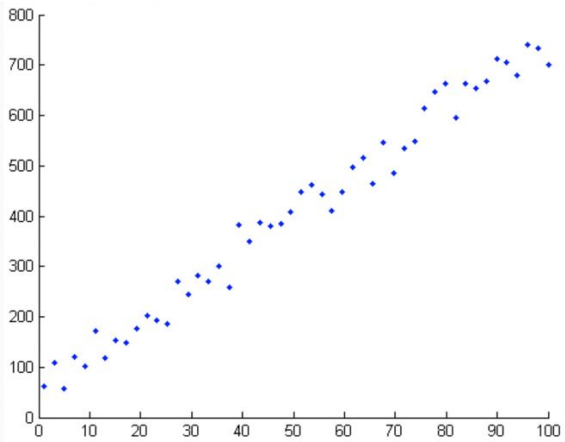
Pie chart for a categorical variable

A **pie chart** is a way to visualize the static composition (aka, distribution) of a variable (or single group).



Scatter plots to visualize relationships

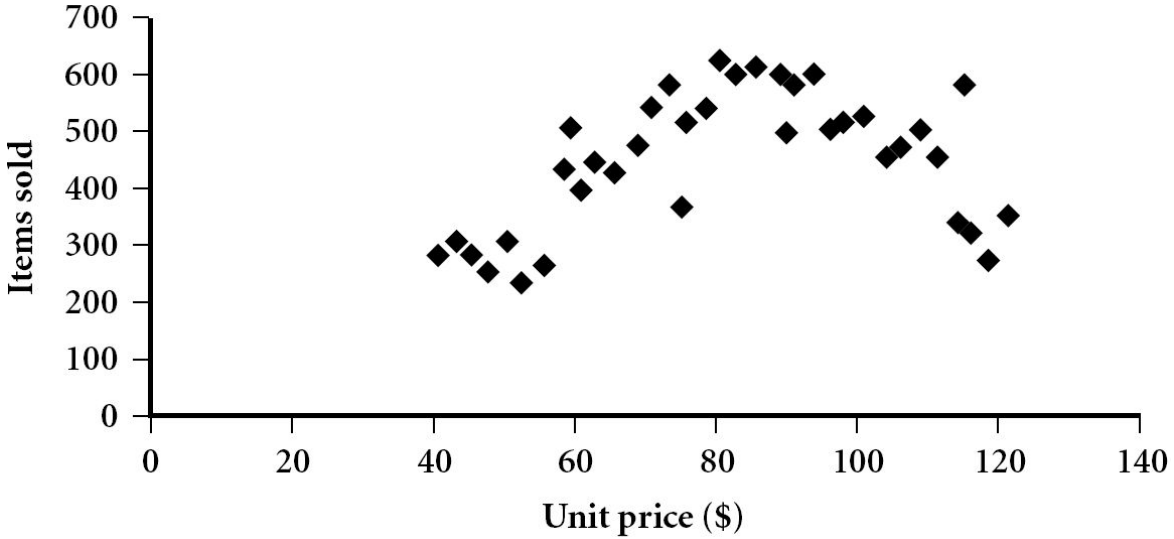
A **scatter plot** is a way to visualize the relationship between two different attributes of multi-dimensional data.



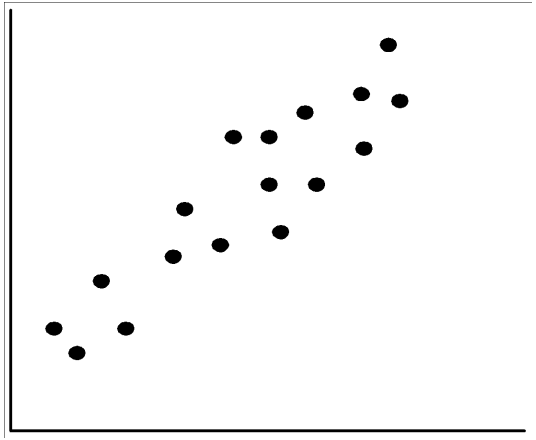
Scatter plot

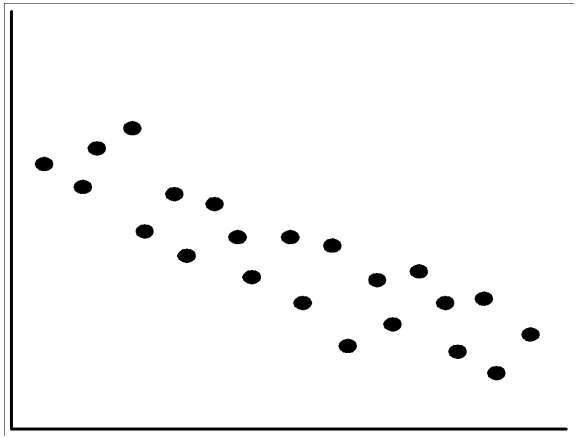
Provides a first look at bivariate data to see clusters of points, outliers, etc

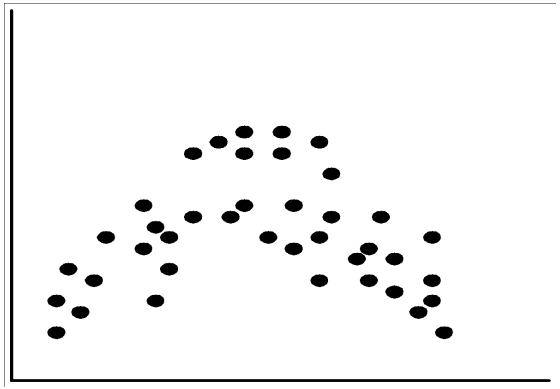
Each pair of values is treated as a pair of coordinates and plotted as points in the plane



Positively and Negatively Correlated Data



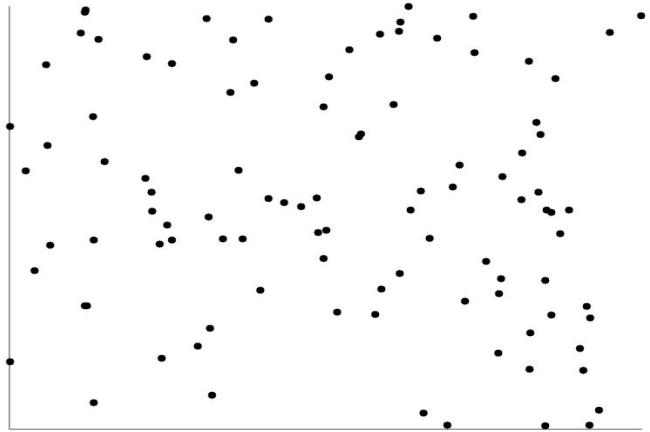


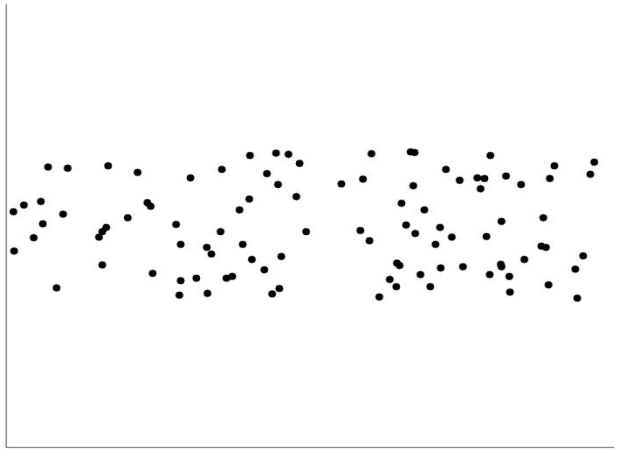


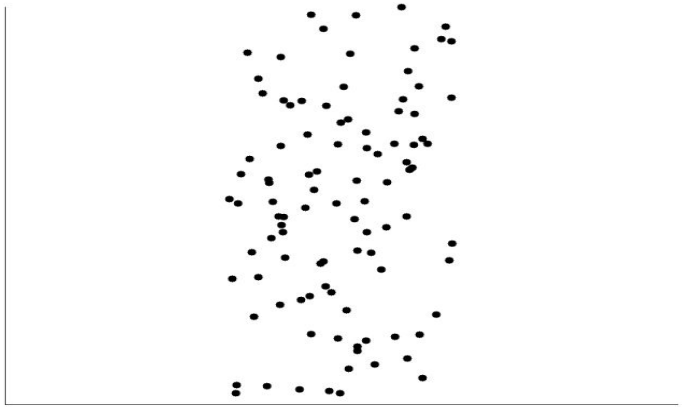
The left half fragment is positively correlated

The right half is negative correlated

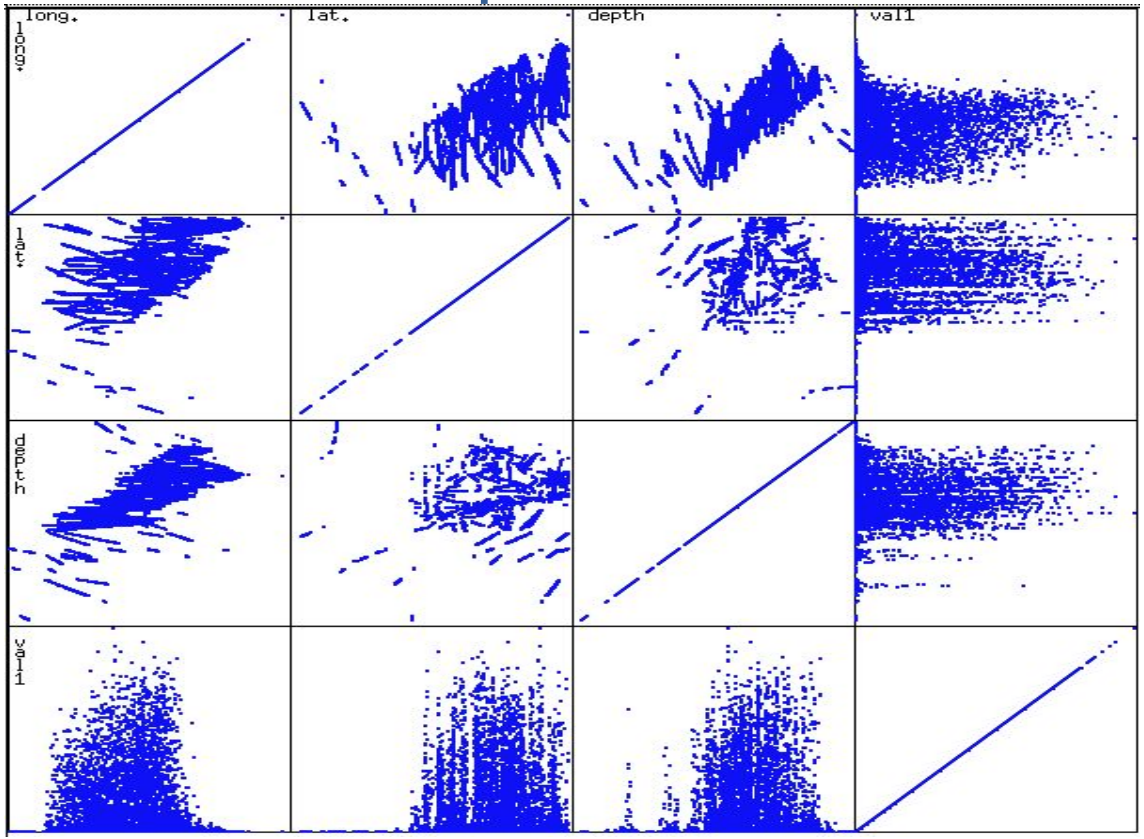
Uncorrelated Data







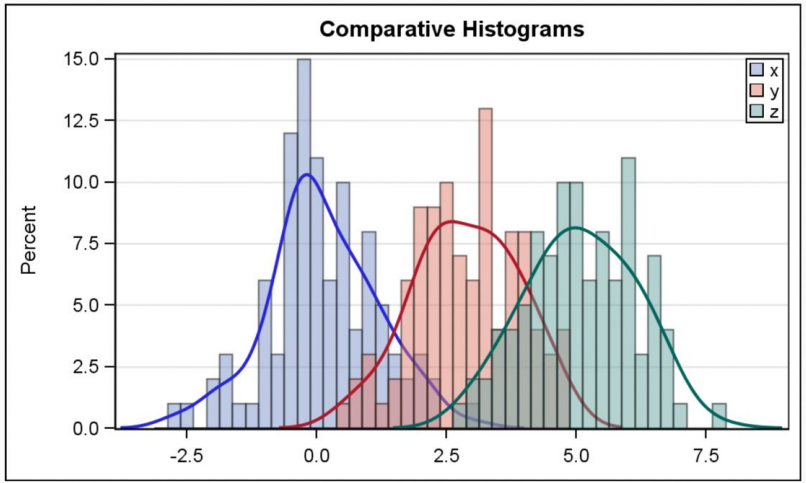
Scatterplot Matrices



Matrix of scatterplots (x-y-diagrams) of the k-dim. data [total of $\binom{k}{2} + k$ unique scatterplots]

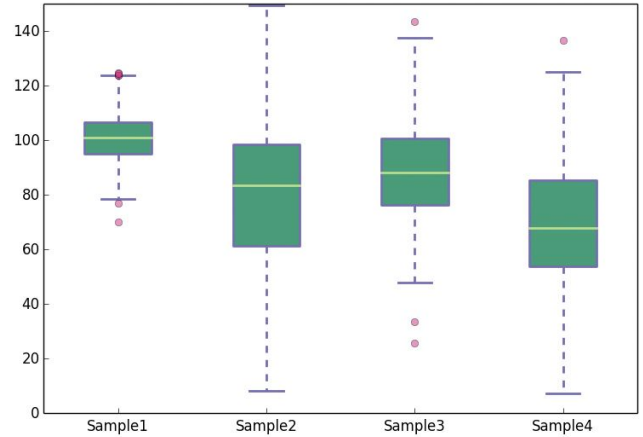
Multiple histograms

Plotting **multiple histograms** on the same axes is a way to visualize how different variables compare (or how a variable differs over specific groups).



Boxplots

A **boxplot** is a simplified visualization to compare a quantitative variable across groups. It highlights the range, quartiles, median and any outliers present in a data set.



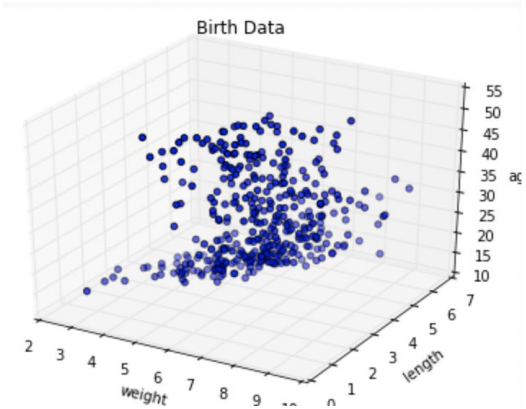
[Not] Anything is possible!

Often your dataset seem too complex to visualize:

- Data is too high dimensional (how do you plot 100 variables on the same set of axes?)
- Some variables are categorical (how do you plot values like Cat or No?)

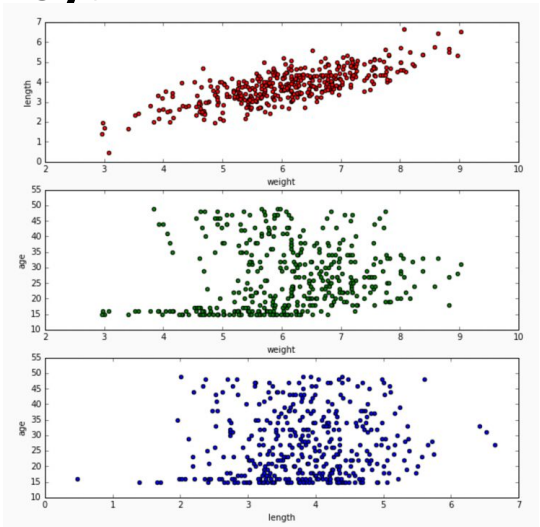
More dimensions not always better

When the data is high dimensional, a scatter plot of all data attributes can be impossible or unhelpful



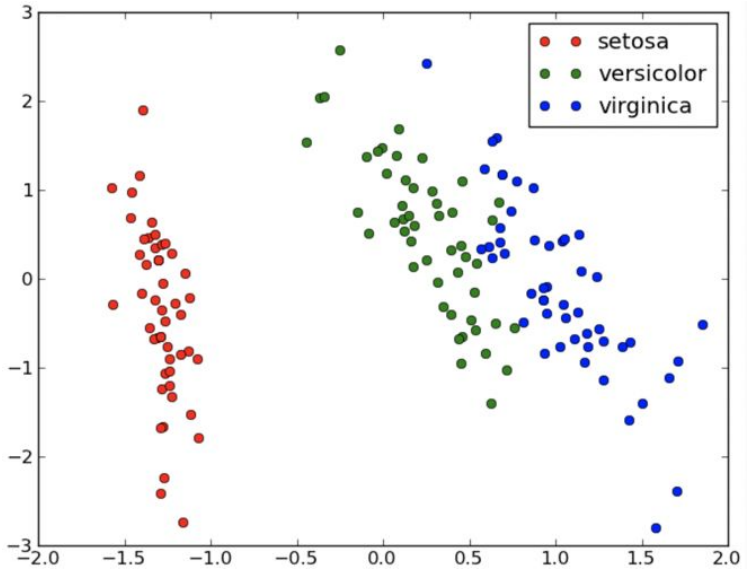
Reducing complexity

Relationships may be easier to spot by producing multiple plots of lower dimensionality.



Reducing complexity

For 3D data, color coding a categorical attribute can be “effective”



This visualizes a set of Iris measurements. The variables are: petal length, sepal length, Iris type (setosa, versicolor, virginica).