Hacettepe Üniversitesi
Bilgisayar Mühendisliği Bölümü

BBM467 - Data Intensive Applications

What is Big Data?

Dr. Fuat Akal
akal@hacettepe.edu.tr

BIG DATA

Simple Questions

- How long would it take you (together with few friends maybe) to develop an application like WhatsApp?
- Why do you think Facebook bought WhatsApp for $19 Billion?

Define Big

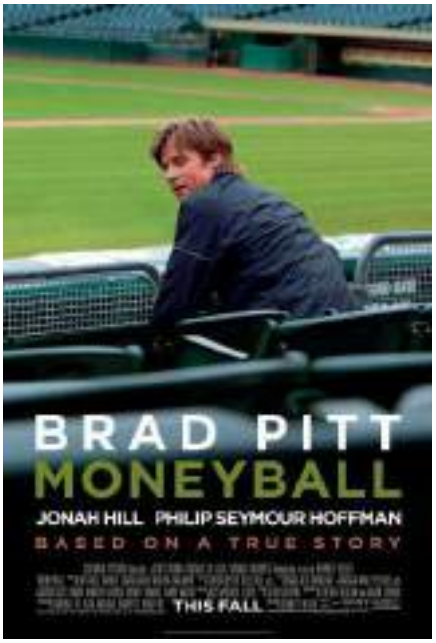
- 1 Bit = Binary Digit
- 8 Bits = 1 Byte
- 1000 Bytes = 1 Kilobyte
- 1000 Kilobytes = 1 Megabyte
- 1000 Megabytes = 1 Gigabyte
- 1000 Gigabytes = 1 Terabyte
- 1000 Terabytes = 1 Petabyte
- 1000 Petabytes = 1 Exabyte
- 1000 Exabytes = 1 Zettabyte
- 1000 Zettabytes = 1 Yottabyte
- 1000 Yottabytes = 1 Brontobyte
- 1000 Brontobytes = 1 Geopbyte

Your data (TB/PB)

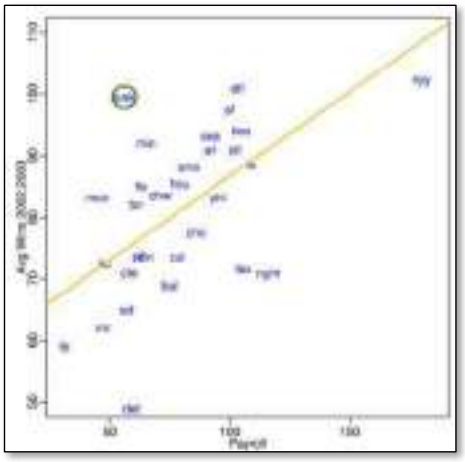
Your friend's data (TB/PB)

The World (EB)

Infographic from Lori Lewis and Chadd Callahan of Cumulus Media



Oakland A's general manager Billy Beane's successful attempt to assemble a baseball team on a lean budget by employing computer-generated analysis to acquire new players.



How to Sell More Video Games?

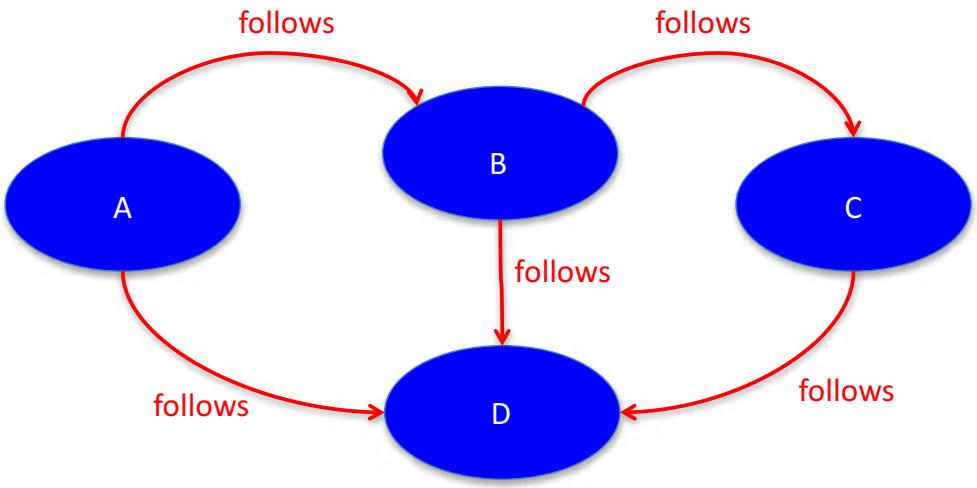
- Retail Industry
  - Busiest shopping time of the year and one video game is flying off the shelves.
  - In the past, stores that can react to this demand had the advantage.
  - Today, stores that can predict the trends and prepare for future demands had the advantage.
    - Combine data from web browsing patterns, movie releases, social media, enterprise data etc.

Better Treatments Possible?

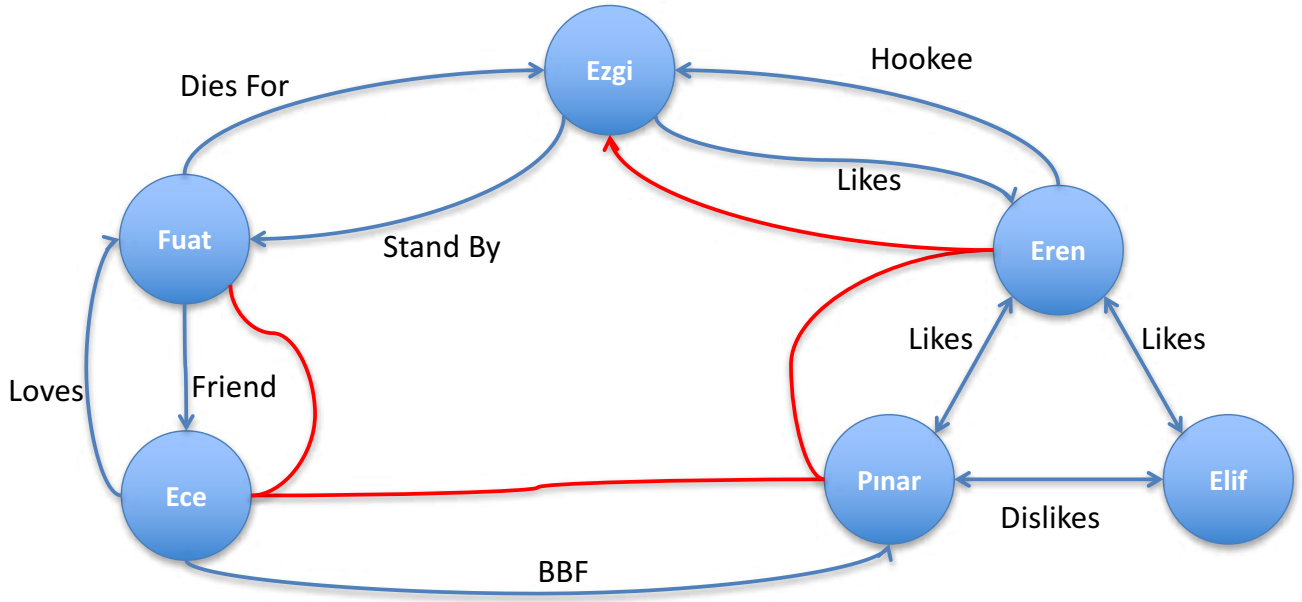
- Health Care
  - pose more complex questions to compare effectiveness of treatments
  - get a view into patients with similar illnesses.
  - a more complete picture of their patients and what is happening in the hospital.
- Being able to see new commonalities in treatment develop new protocols, and make a difference in the care delivered to the patients.

Discover Relationships

- Who follows whom on social media?



Discover More Complicated Relationships



Data Never Sleeps

- With every website browsed, status shared, and photo uploaded, we leave a digital trail that continually grows the hulking mass of big data.

"… İspanya Süper Kupa Finali ilk maçında bir Fenerbahçeli taraftar dikkat çekti. Fenerbahçe forması bulunan taraftar, maçın başında Mesut'un attığı golle sevinç yaşadı. Fenerli'nin Barcelona'yı öne geçiren goldeki üzüntüsü ise objektiflere böyle yansıdı. --Sabah"

User Comments on the Picture

- "son günlerde mutlu bir fenerbahçeli gördünüz mü ki? :)"
- "Her ne sebeple olursa olsun Fb li birinin SPORDA üzülmesi kadar beni mutlu eden bir olay yoktur"
- "Mesut'la ilgisi yoktur adam orda yaşıyordur Real'i destekliyordur yani ne olacak yada iyi para basmıştır"

Example: Shopping Tip ;-)

Searched for : “kırmızı pancar salatası nasıl yapılır”

Received an SMS few seconds later

Nice coincidense ☺

Example: Telecom Companies

- My phone got older and outdated
- Browsed Vodafone’s Web site regularly by using my Turkcell line
- Hoping to get a better deal from Turkcell

Did not work.

- Either they do big data very well
- Or, they do not do it at all ☺

Netflix Challenge

- October 2006: Netflix offers $1M for an improved recommender algorithm.
- 6 years of data for training: 2000-2005
- $1M grand prize for 10% improvement

What People Think?

- Social Media analysis
  - What do people think about new iPhone?
- Example: Analyse tweets about new iPhone?
  - There are emotions involved
    - It is great!
    - It is great ;-)

Other Things are Involved, too!

Big Data in Physics

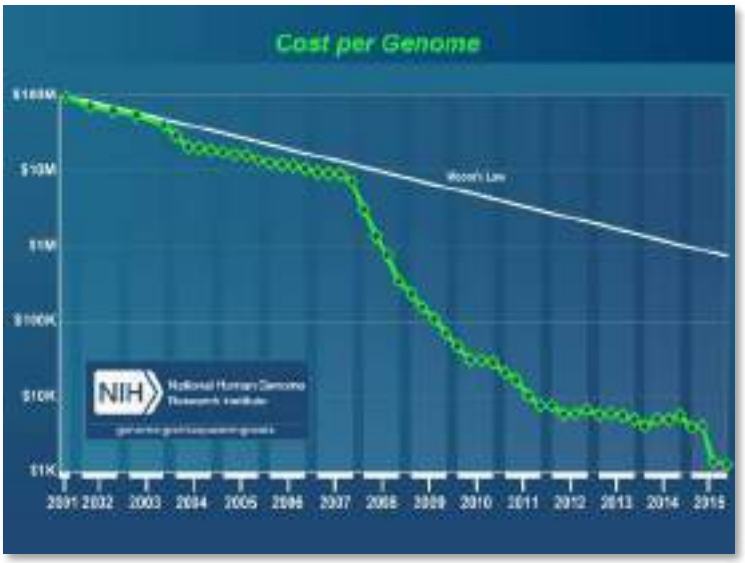
50 Petabytes of data per year!

Big Data in Health Care

- Faster and cheaper technology and data storage
- Widespread sensing devices
- An increase in “born” digital data
- Greater availability of data via repositories
- Data sharing mandates

Faster and Cheaper Technology and Data Storage

The cost to sequence a whole human genome sequence has fallen from +$100 million to less than $1,000 over the past 15 years.



Sensing Devices

- Smartwatches
- Smart jewelry
- Fitness trackers
- Sport watches
- Smart glasses
- Smart clothing…

An Increase in “born” Digital Data

Data that originates as digital data, rather than being converted or digitized later is proliferating. Think digital electronic medical records, implanted medical devices, diagnostic imaging technology…

Greater Availability of Data via Repositories

Sharing Mandates

The number of funders and journals with data sharing policies has grown significantly in the past decade…

Why is Big Data Processing Difficult?

- You need more data than your data warehouse.
  - you need more data than you have
  - logs, Twitter feeds, blogs, customer surveys, ...
- You need to ask the right questions.
  - data alone is silent
- You need technology and organization that help you concentrate on asking the right questions.

Limitations of the State of the Art

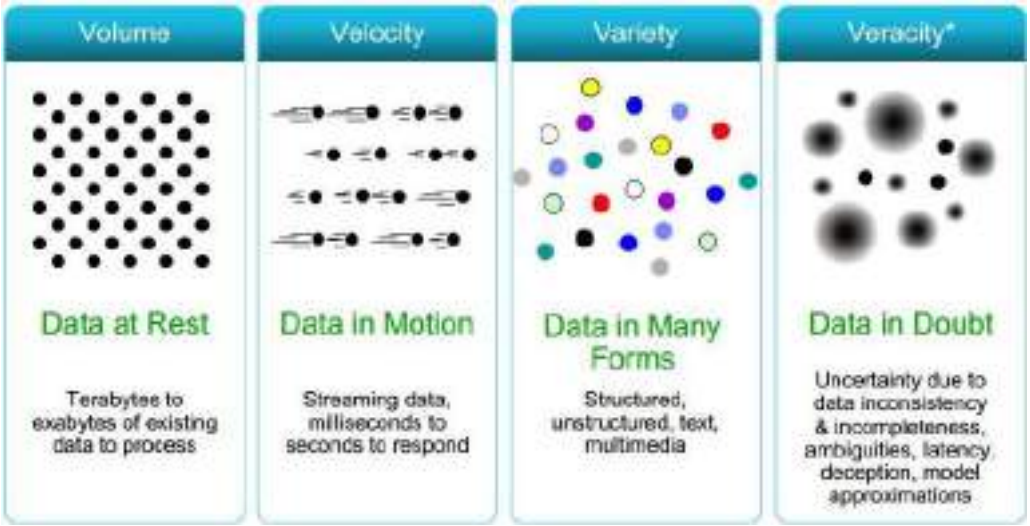
Is bigger = smarter?

- Yes!
  - tolerate errors
  - discover the long tail and corner cases
  - machine learning works much better
- But!
  - more data, more error (e.g., semantic heterogeneity)
  - with enough data you can prove anything
  - still need humans to ask right questions

Big Data Success Story

- Google Translate
  - you collect snipets of translations
  - you match sentences to snipets
  - you continuously debug your system
- Why does it work?
  - there are tons of snipets on the Web
  - there is a ground truth that helps to debug system

Four V's of Big Data



Four V's of Big Data

- Volume
  - Big data is always large in volume. It actually doesn't have to be a certain number of *petabytes* to qualify.
- Velocity (or speed)
  - Refers to how fast the data is coming in, but also to how fast you need to be able to analyze and utilize it.
- Variety (diversity)
  - Points to the number of sources or incoming vectors leading to your databases.
    - That might be embedded sensor data, phone conversations, documents, video uploads or feeds, social media, and much more.
- Veracity (data in doubt)
  - If you can't trust the data itself, the source of the data, or the processes you are using to identify which data points are important, you have a veracity problem.

More V's

- Variability
  - Inconsistency in data, inconsistent speed at which big data is loaded into your database
- Validity
  - Similar to veracity, validity refers to how accurate and correct the data is for its intended use
- Vulnerability
  - Big data brings new security question
- Volatility
  - You cannot store data indefinitely anymore
- Visualization
  - Challenge to visualize, need different ways to represent data
- Value
  - The other Vs are meaningless if you don't derive business value from the data

DATA SCIENCE

Sexy Jobs

- "I keep saying the sexy job in the next ten years will be statisticians. People think I'm joking, but who would've guessed that computer engineers would've been the sexy job of the 1990s?"

- "The ability to take data, to be able to understand it, to process it, to extract value from it, to visualize it, to communicate it's going to be a at the professional level but even at the educational level for elementary school kids, for high school kids, for college kids. Because now we really do have essentially free and ubiquitous data."

Hal Varian, Google's Chief Economist

Types of Data We Have

- Relational Data (Tables/Transaction/Legacy Data)
- Text Data (Web)
- Semi-structured Data (XML)
- Graph Data
- Social Network, Semantic Web (RDF), …
- Streaming Data
- …

What to Do with This Data?

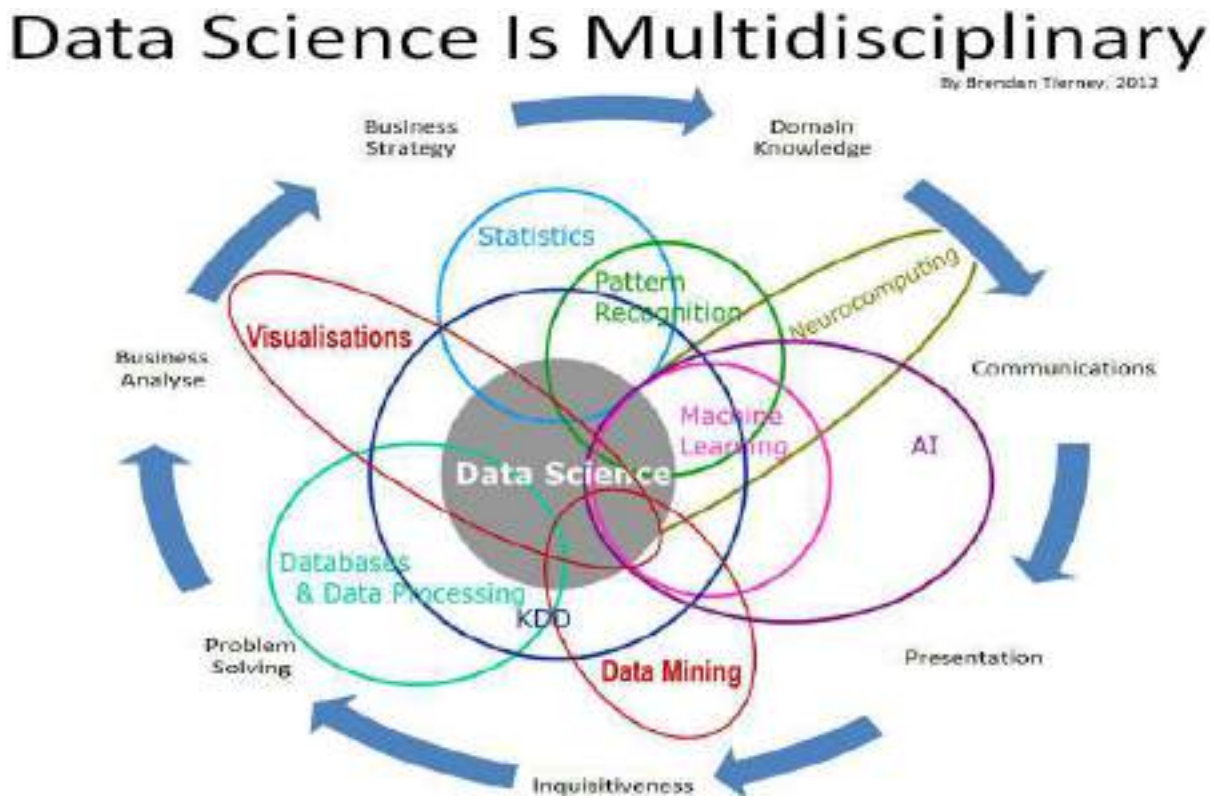
- **Explore**: identify patterns
- **Predict**: make informed guesses
- **Infer**: quantify what you know

What is Data Science?

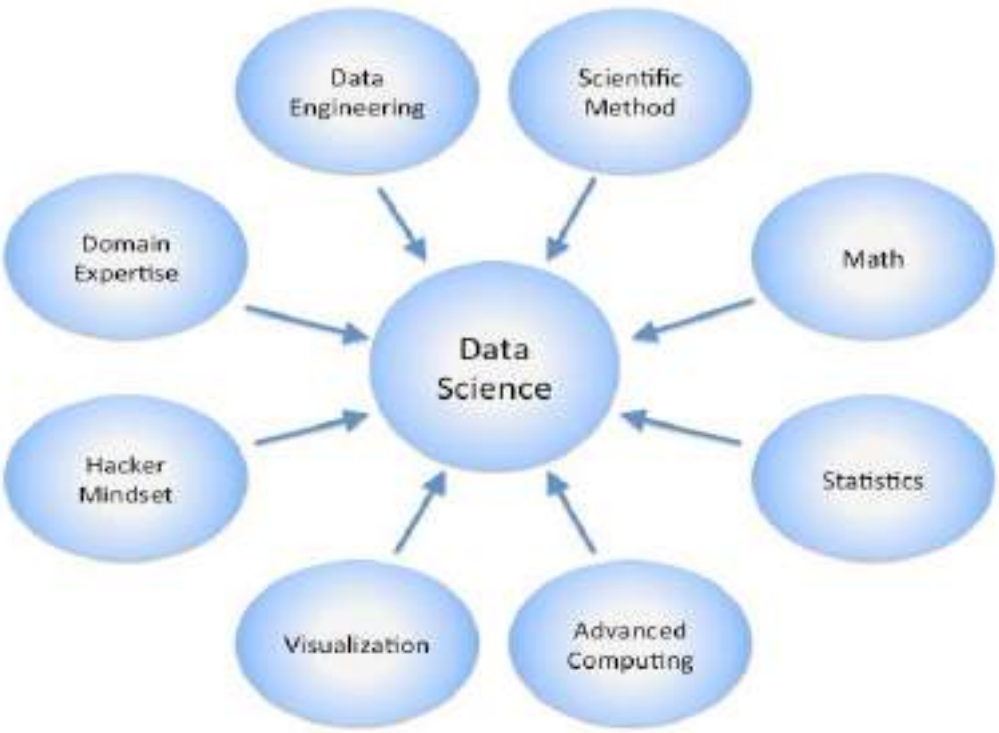
- An area that manages, manipulates, extracts, and interprets knowledge from tremendous amount of data
- Data science (DS) is a multidisciplinary field of study with goal to address the challenges in big data
- Data science principles apply to all data – big and small

Combined Effort

- Theories and techniques from many fields and disciplines are used to investigate and analyze a large amount of data to help decision makers in many industries such as science, engineering, economics, politics, finance, and education
  - Computer Science
    - Pattern recognition, visualization, data warehousing, High performance computing, Databases, AI
  - Mathematics
    - Mathematical Modeling
  - Statistics
    - Statistical and Stochastic modeling, Probability.



Components

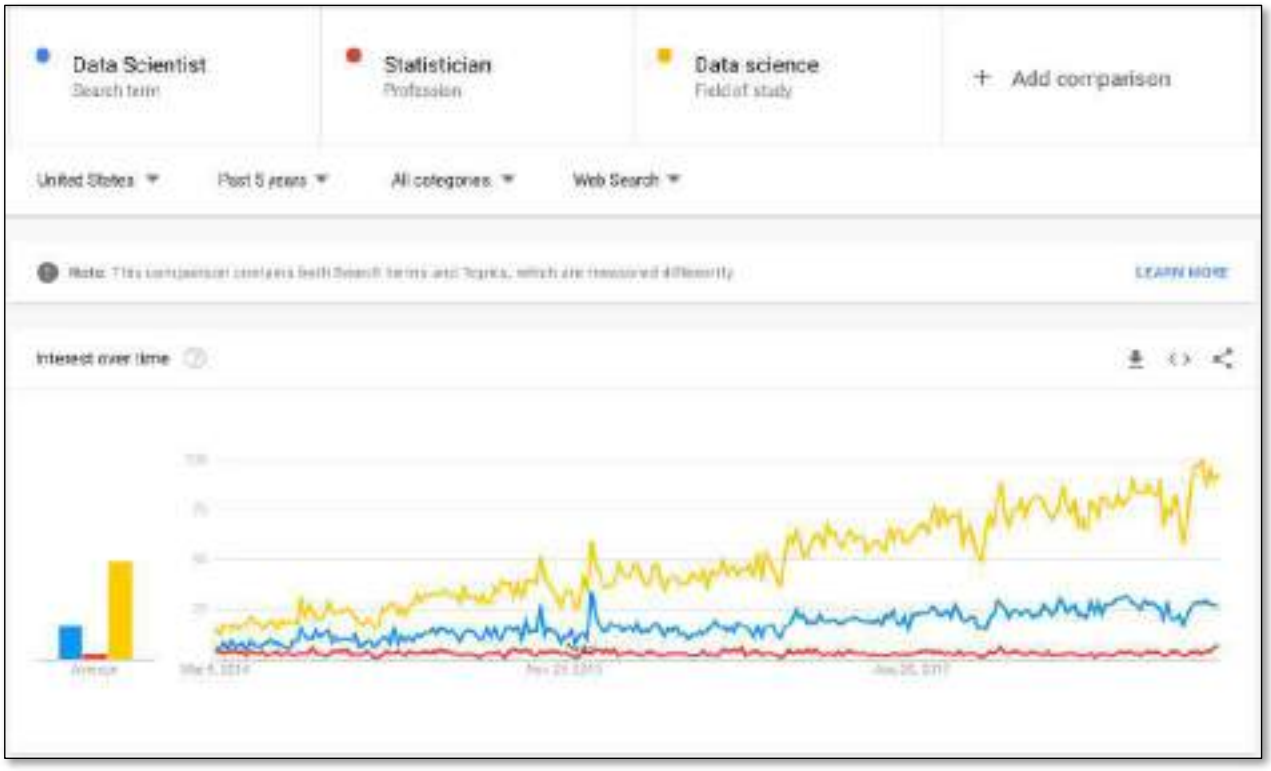


Who is Data Scientist?

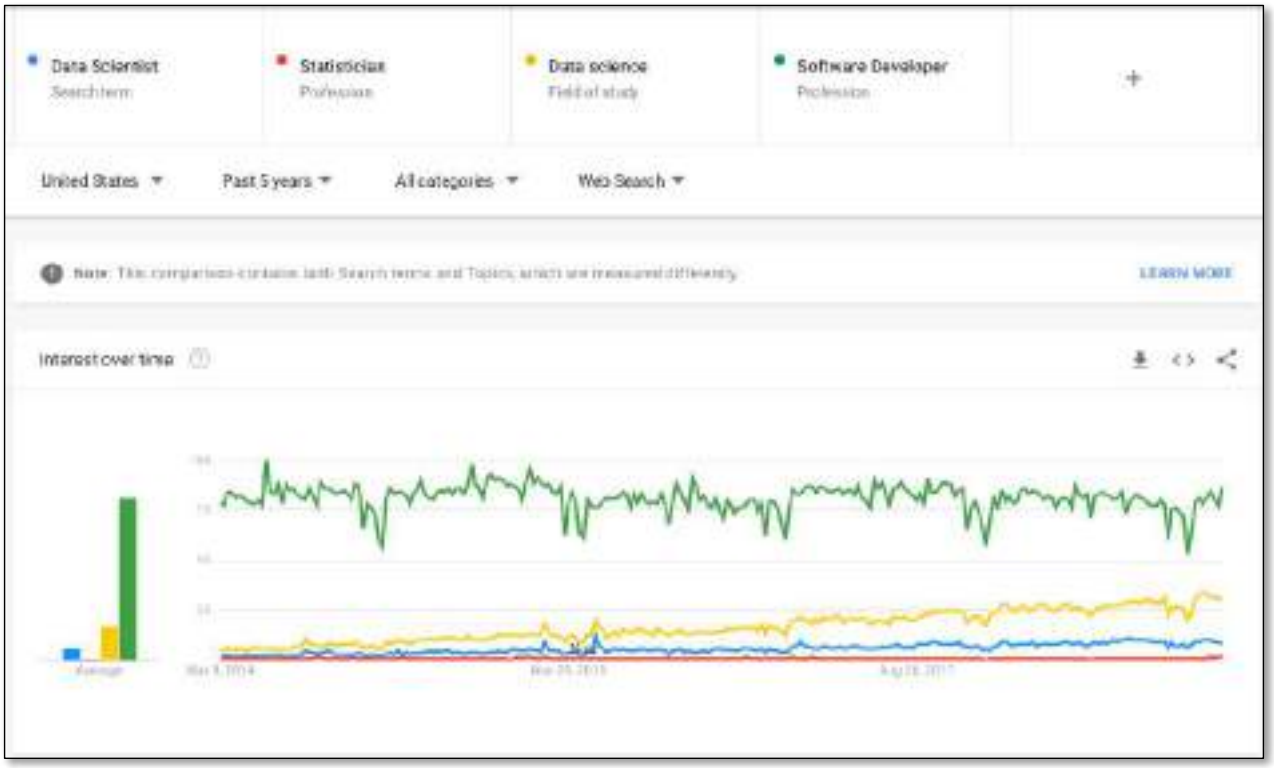
- "A person employed to analyse and interpret complex digital data, such as the usage statistics of a website, especially in order to assist a business in its decision-making."

- "A *data scientist* is a rare hybrid, a computer *scientist* with the programming abilities to build software to scrape, combine, and manage *data* from a variety of sources and a statistician who knows how to derive insights from the information within."

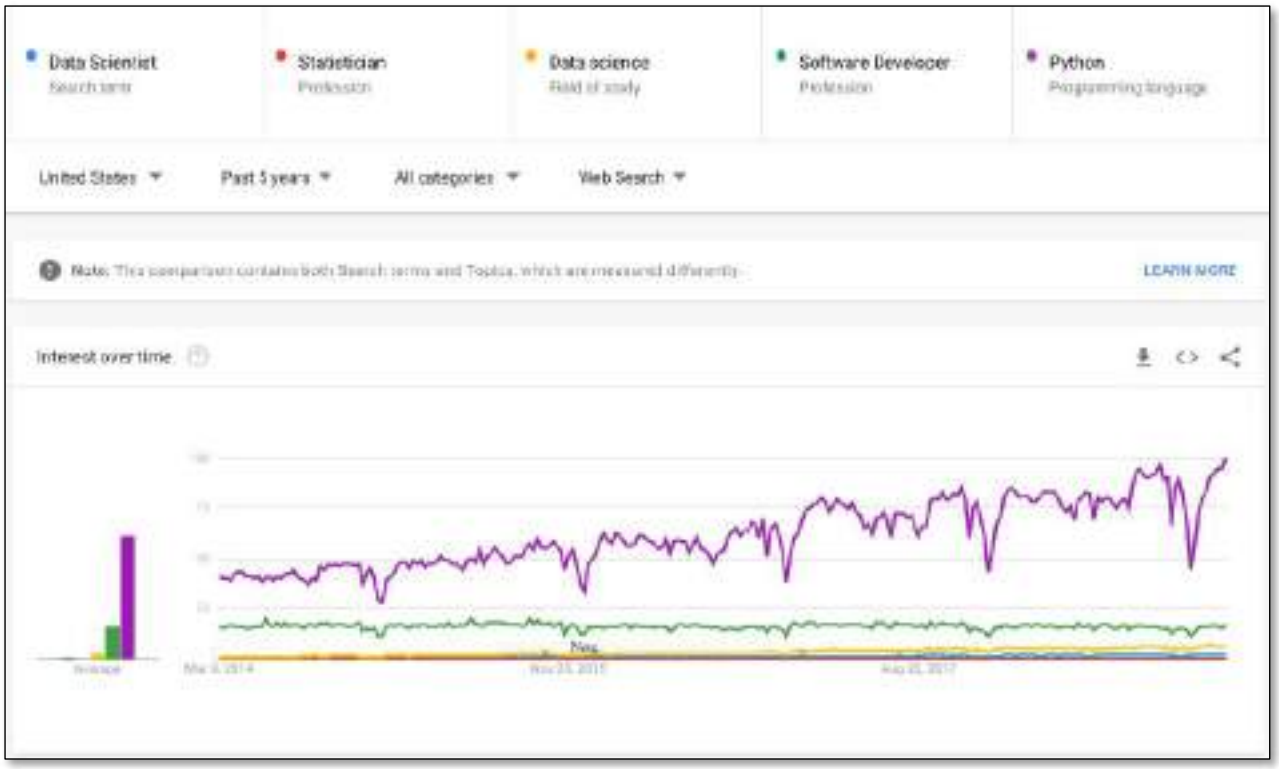
Data Scientist is Sexy?



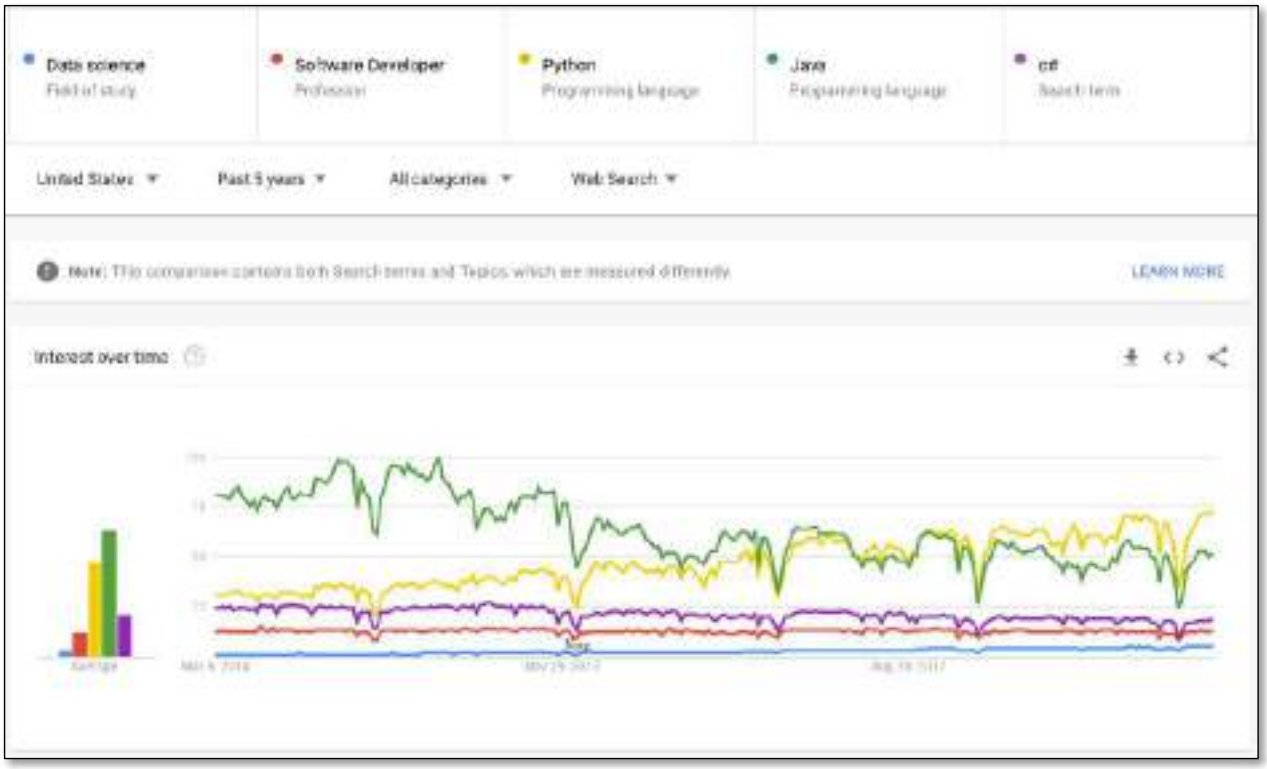
Maybe not?



Maybe It is



Not Quite Sure



Smile

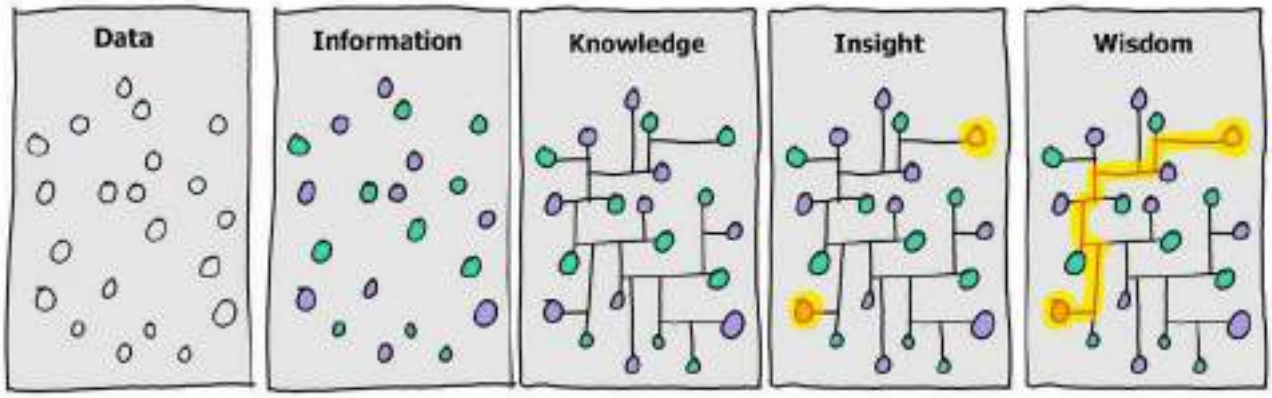
- What is Big Data?
  - “Saklamak için dahi odalar dolusu hard disk gerektiren veridir”
- What are four Vs?
  - ”Hocam sabah bakmıştım ama şu an hatırımdan çıkmış durumda walla.”

Midterm questions and answers from Fall2013 ☺

So Far…

- We have learnt a number of buzzwords, some cool examples
  - Which should suffice to survive any discussion with your friends.

Information Continuum



Cartoon by David Somerville, based on a two pane version by Hugh McLeod

Data Science Methodology

Welcome

IBM Developer

SKILLS NETWORK

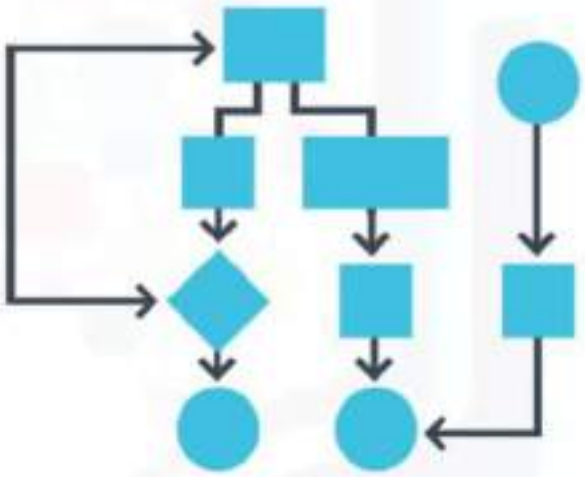
What is a methodology anyway?

A methodology is a defined way of....

meth·od·ol·o·gy

noun
noun: **methodology**; plural noun: **methodologies**

1 a system of methods used in a particular area of study or activity.
"a methodology for investigating the concept of focal points"



IBM Developer

SKILLS NETWORK

The CRoss Industry Standard Process for Data Mining (CRISP-DM)

Methodology by John Rollins based on CRISP-DM

John Rollins
Data Scientist, IBM Analytics, IBM

John B. Rollins, Ph.D., P.E., is a Data Scientist, IBM Analytics, IBM. Prior to joining IBM Netezza, he was an engineering consultant, professor and researcher. He has authored many patents, papers and books. He holds doctoral degrees in economics and petroleum engineering and is a registered professional engineer in Texas.

In a nutshell...

The **Data Science Methodology** aims to answer the following 10 questions in this prescribed sequence:

From problem to approach:

1. *What is the problem that you are trying to solve?*
2. *How can you use data to answer the question?*

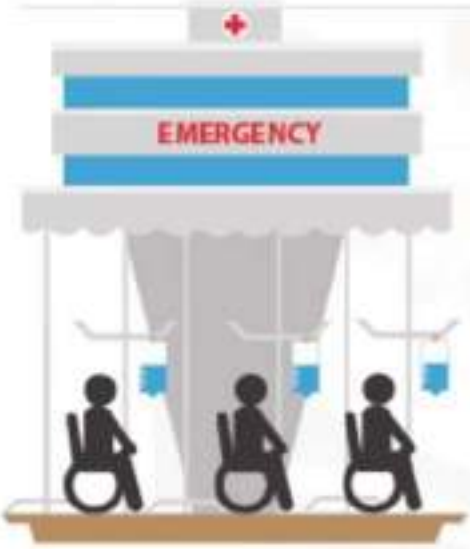
Working with the data:

3. *What data do you need to answer the question?*
4. *Where is the data coming from (identify all sources) and how will you get it?*
5. *Is the data that you collected representative of the problem to be solved?*
6. *What additional work is required to manipulate and work with the data?*

Deriving the answer:

7. *In what way can the data be visualized to get to the answer that is required?*
8. *Does the model used really answer the initial question or does it need to be adjusted?*
9. *Can you put the model into practice?*
10. *Can you get constructive feedback into answering the question?*

About the Case Study



Glossary of Data Science Terms

- analytic approach
- analytics
- cohort
- cohort study
- comorbidities
- congestive heart failure (CHF)
- CRISP-DM
- data analysis
- data cleansing
- data science
- data scientist
- decision tree
- decision tree classification
- descriptive modeling
- descriptive statistics
- domain knowledge
- dominating decision rule
- histogram
- hospital readmission
- Iterative process > Iteration
- machine learning
- mean
- median
- methodology
- model > conceptual model
- pairwise comparison (correlation)
- patient cohort
- pattern
- predictive modeling
- predictors
- ROC curve
- standard deviation
- statistics
- structured data > data model
- text analysis > data mining
- training set
- univariate
- unstructured data
- visualization techniques

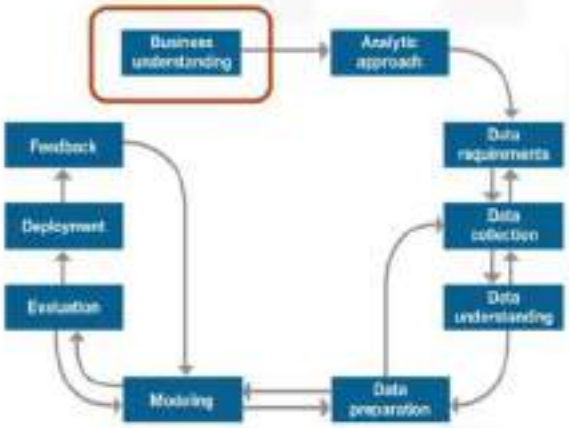
Data Science Methodology

Business Understanding

The Importance of Questions

From Understanding to Approach

From Understanding to Approach



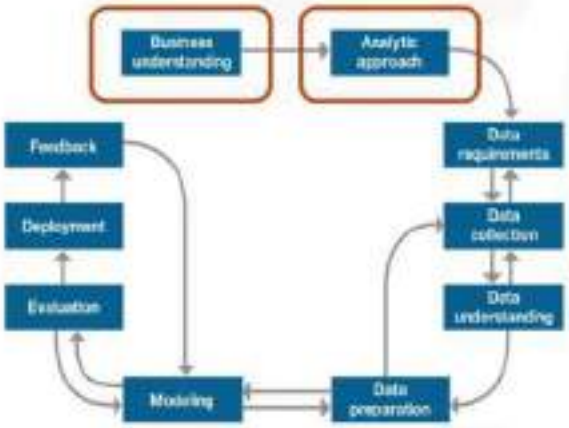
Business understanding

- *What is the problem that you are trying to solve?*

IBM Developer

SKILLS NETWORK

From Understanding to Approach



Business understanding

- *What is the problem that you are trying to solve?*

Analytic approach

- *How can you use data to answer the question?*

IBM Developer

SKILLS NETWORK

Seeking clarification – What's the goal?

Supporting the Goal

Getting stakeholder "buy-in" and support

Case Study – Applying the concepts

What is the best way to allocate the limited healthcare budget to maximize its use in providing quality care?

Case Study – What are the goals & objectives?

Define the GOALS

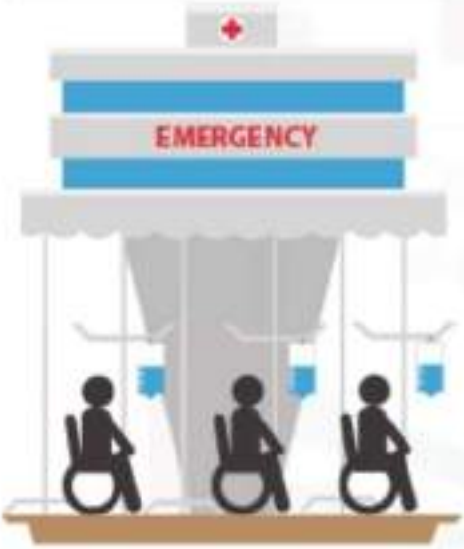
- To provide quality care without increasing costs

Define the OBJECTIVES

- To review the process to identify inefficiencies

After spending time to determine the goals and objectives, the team prioritized "patient readmissions" as an effective area for review.

Case Study – Examining Hospital Readmissions



Roughly 25-35% of patients who complete rehab treatment will be readmitted to a rehabilitation center within one year and roughly 50% will be readmitted within five years.

After reviewing some records,

- It was discovered that the patients with congestive heart failure (CHF) were at the top of the readmission list.
- It was further determined that a decision-tree model could be applied to review this scenario, to determine why this was occurring.

Case Study – Pilot project kickoff

Case Study – What's the sponsor's involvement?

1. Set overall direction.
2. Remained engaged and provided guidance.
3. Ensured necessary support, where needed.

Case Study – Identifying the business requirements

1. Predict CHF readmission outcome (Y or N) for each patient
2. Predict the readmission risk for each patient
3. Understand explicitly what combination of events led to the predicted outcome for each patient
4. Easy to understand and apply to new patients to predict their readmission risk

Data Science Methodology

Analytic Approach

IBM Developer

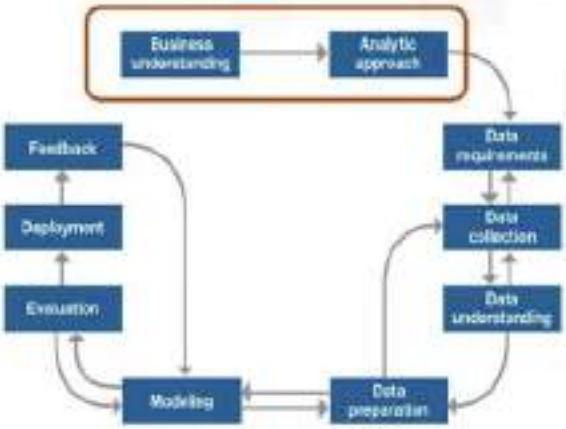
SKILLS NETWORK

Which analytic approach to pick?

IBM Developer

SKILLS NETWORK

From Understanding to Approach



Business understanding

- *What is the problem that you are trying to solve?*

Analytic approach

- *How can you use data to answer the question?*

Pick analytic approach based on type of question

Descriptive

- Current status

Diagnostic (Statistical Analysis)

- What happened?
- Why is this happening?

Predictive (Forecasting)

- What if these trends continue?
- What will happen next?

Prescriptive

- How do we solve it?

What are the types of questions?

If the question is to determine probabilities of an action
- Use a Predictive model

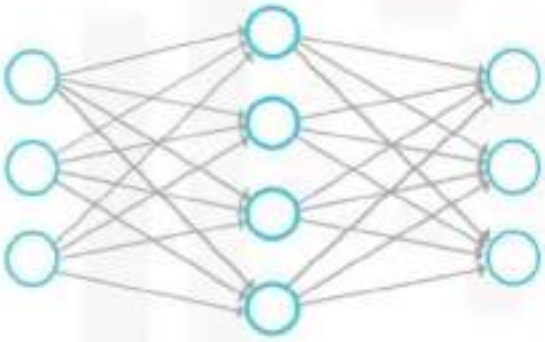
If the question is to show relationships
- Use a descriptive model

If the question requires a yes/no answer
- Use a classification model

Analytic approach
- *How can you use data to answer the question?*

- The correct approach depends on business requirements for the model

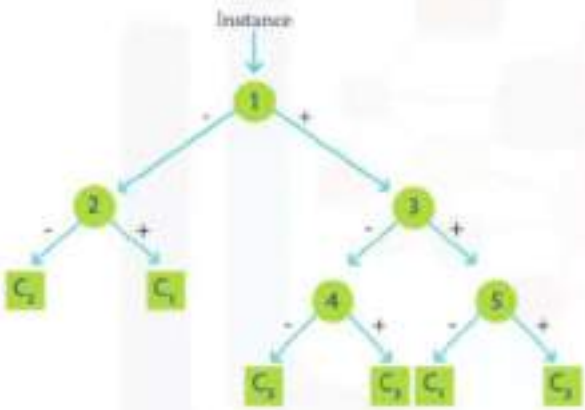
Will machine learning be utilized?



Machine Learning
- Learning without being explicitly programmed
- Identifies relationships and trends in data that might otherwise not be accessible or identified
- Uses clustering association approaches

Case Study – Applying the concepts

Case Study – Decision tree classification selected!



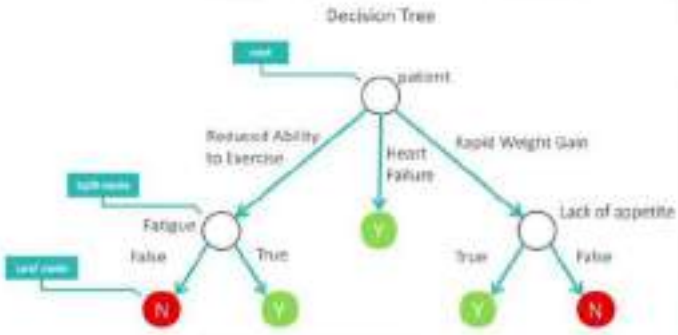
Predictive model

- To predict an outcome

Decision tree classification

- Categorical outcome
- Explicit "decision path" showing conditions leading to high risk
- Likelihood of classified outcome
- Easy to understand and apply

Case Study – Example of decision tree classification



Predictive model

- To predict an outcome

Decision tree classification

- Categorical outcome
- Explicit "decision path" showing conditions leading to high risk
- Likelihood of classified outcome
- Easy to understand and apply

Data Science Methodology

Data Requirements

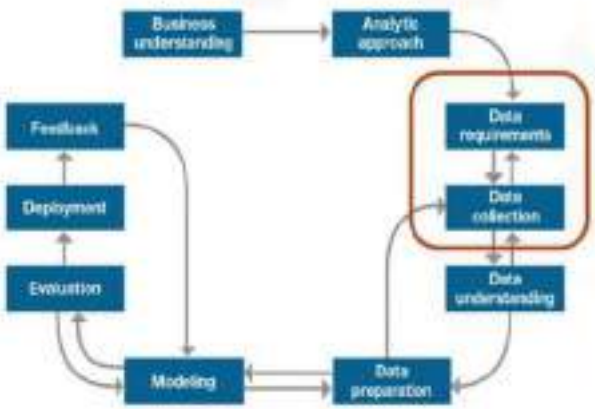
Data requirements – Cooking with data

Data requirements – Cooking with data

IBM Developer

SKILLS NETWORK

From Requirements to Collection



Data Requirements

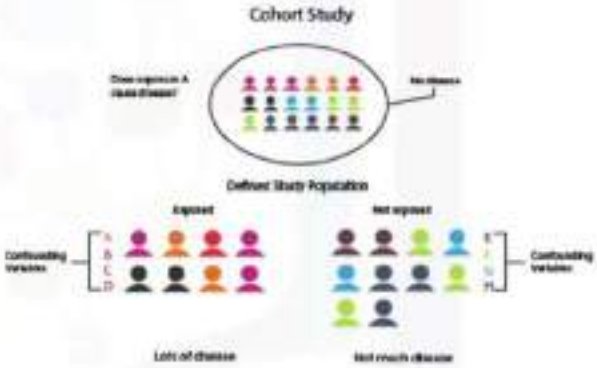
- *What are data requirements?*

Data Collection

- *What occurs during data collection?*

Case Study – Selecting the cohort

- Define and select cohort
  - In-patient within health insurance provider's service area
  - Primary diagnosis of CHF in one year
  - Continuous enrollment for at least 6 months prior to primary CHF admission
  - Disqualifying conditions



Case Study – Defining the data

- Content, formats, representations suitable for decision tree classifier
  - One record per patient with columns representing variables (dependent variable and predictors)
  - Content covering all aspects of each patient's clinical history
    - Transactional format
    - Transformations required

Data Science Methodology

Data Collection

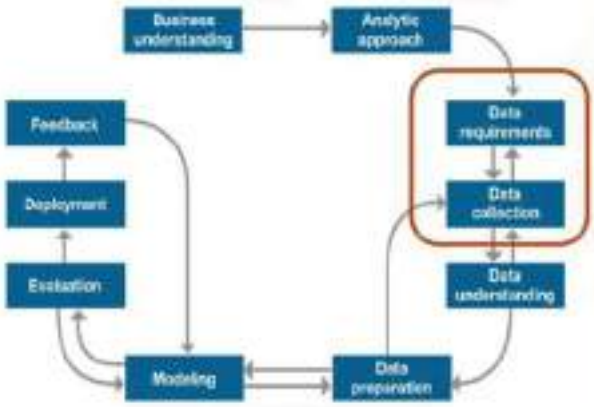
Data collection – Ingredients?

Data collection – Ingredients?

IBM Developer

SKILLS NETWORK

From Requirements to Collection



Data Requirements

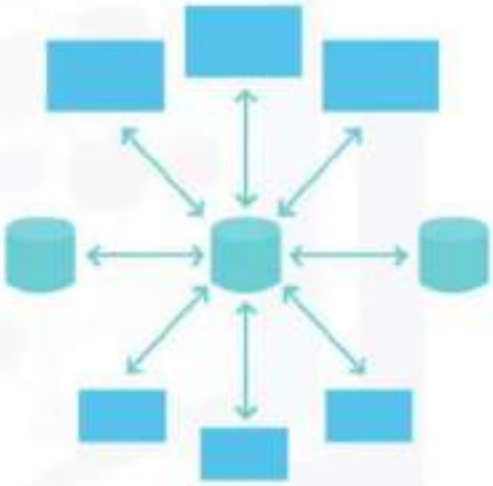
- *What are data requirements?*

Data Collection

- *What occurs during data collection?*

Case Study – Gathering available data

- Available data sources
  - Corporate data warehouse (single source of medical & claims, eligibility, provider and member information)
  - In-patient record system
  - Claim payment system
  - Disease management program information



Case Study – Deferring Inaccessible data

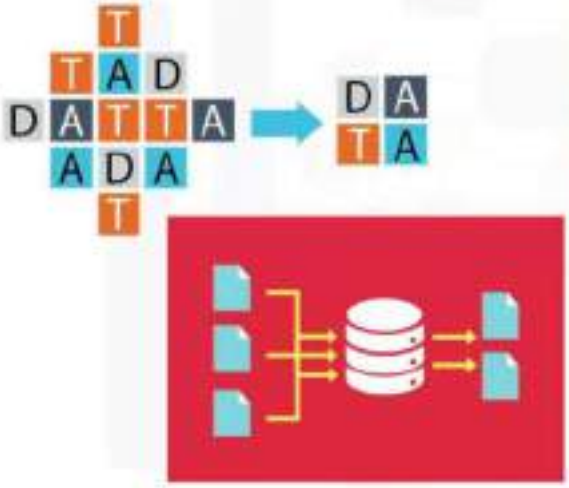
- Data wanted but not available
  - Pharmaceutical records
  - Decided to defer

Attempt to acquire the unavailable data can even be done after getting some intermediate results from the predictive modeling.

DATA NOT AVAILABLE

Case Study – Merging data

- Eliminate redundant data



Data Science Methodology

Data Understanding

IBM Developer

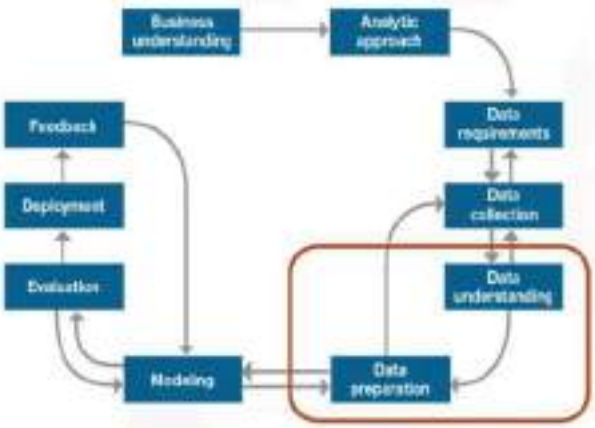
SKILLS NETWORK

Understanding the data

IBM Developer

SKILLS NETWORK

From Understanding to Preparation



Data understanding

- *What does it mean to "prepare" or "clean" data?*

Data preparation

- *What are ways in which data is prepared?*

Is the data that you collected representative of the problem to be solved?

Case Study – Understanding the data

- Descriptive statistics
  - Univariate statistics
  - Pairwise correlations
  - Histogram

$$f(a) + \sum_{k=1}^{n} \frac{1}{k!} \frac{d^k}{dt^k}\bigg|_{t=0} f(u(t)) + \int_0^1 \frac{(1-t)^n}{n!} \frac{d^{n+1}}{dt^{n+1}} f(u(t))\,dt.$$

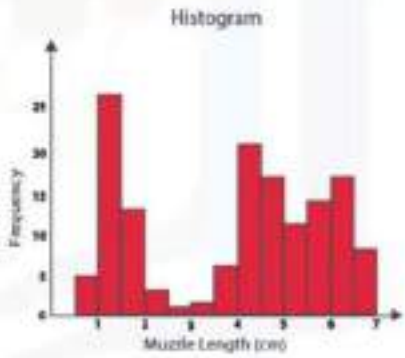
$F_{X,Y}(x,y)$ satisfies

$$F_{X,Y}(x,y) = F_X(x)F_Y(y),$$

or equivalently, their joint density $f_{X,Y}(x,y)$ satisfies

$$f_{X,Y}(x,y) = f_X(x)f_Y(y).$$

Histograms are a good way to understand how values or a variable are distributed, and what sorts of data preparation may be needed to make the variable more useful in a model.



Case study – Looking at data quality

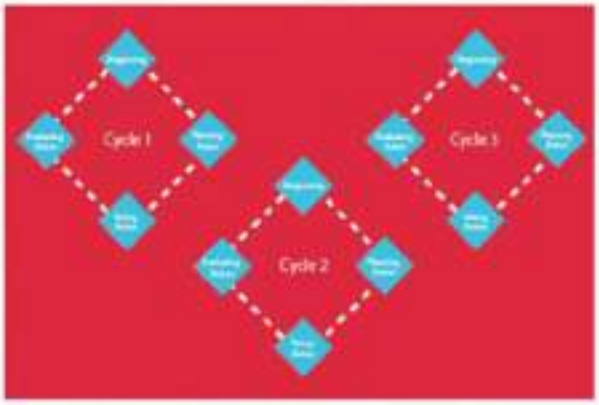
- Data quality
  - Missing values
  - Invalid or misleading values

Sometimes a missing value might mean "no", or "0" (zero), or at other times it simply means "we don't know".

What would 999 mean for the age of a patient?

Case study – This is an iterative process

- Iterative data collection and understanding
  - Refined definition of "CHF admission"



The more one works with the problem and the data, the more one learns and therefore the more refinement that can be done within the model.

Data Science Methodology

Data Preparation - Concepts

IBM Developer

SKILLS NETWORK

Data preparation – Cleansing data

Washing freshly picked vegetables to remove unwanted elements, such as dirt or imperfections.

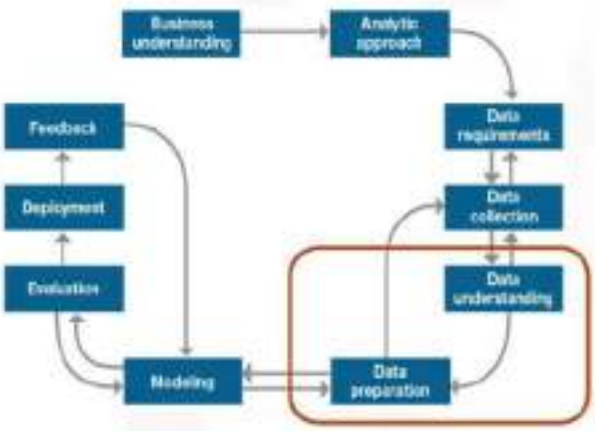
Data preparation is the most time-consuming phase of a data science project

IBM Developer

SKILLS NETWORK

Data preparation – Transforming data

From Understanding to Preparation



Data understanding

- *What does it mean to "prepare" or "clean" data?*

Data preparation

- *What are ways in which data is prepared?*

Examples of data cleansing

	A	B	C	D	E
1	Name	Date	Age	Location	Country
2	John Doe	2012 02 20	32	ON	CAN
3	May Lag	2013 02 33	2	ON	CA
4	Henry Don	30-Sep-12	35	Ontario	CANADA
5	Kelly, Tom	2015 02 20	65	ON	CA
6	John Kell	2016 02 20		AB	CA
7	Henry Don	30-Sep-12	35	Ontario	CANADA
8					

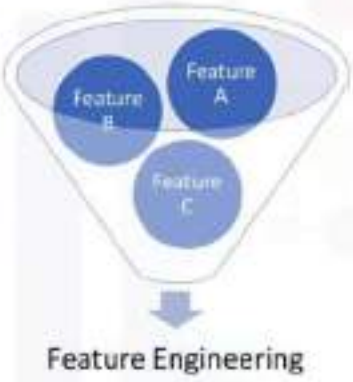
Invalid Values

Missing Data

Remove Duplicates

Formatting

Using domain knowledge



Feature engineering is the process of using domain knowledge of the data to create features that make the machine learning algorithms work.

Feature engineering is critical when machine learning tools are being applied to analyze the data.

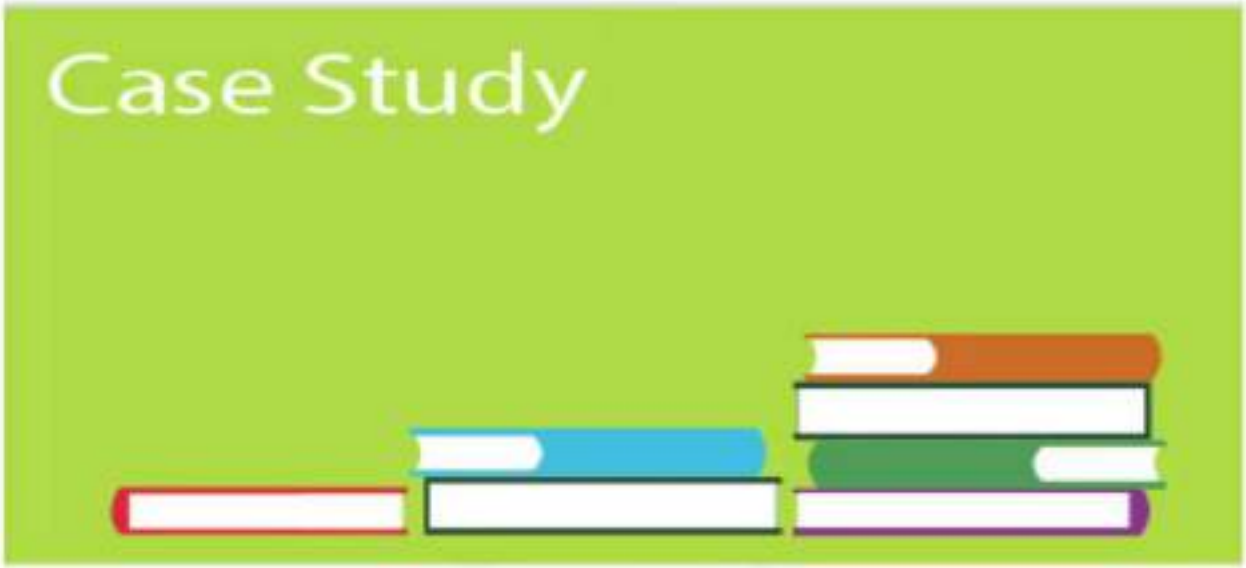
Data Science Methodology

Data Preparation – Case Study

IBM Developer

SKILLS NETWORK

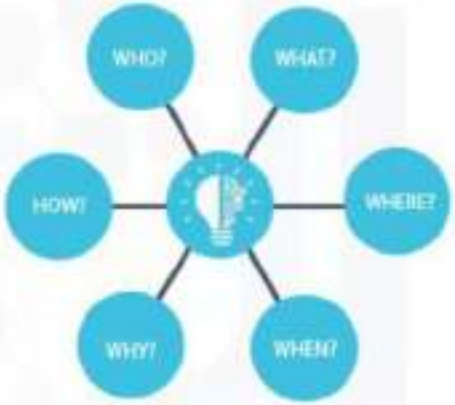
Case Study – Applying the concepts



IBM Developer

SKILLS NETWORK

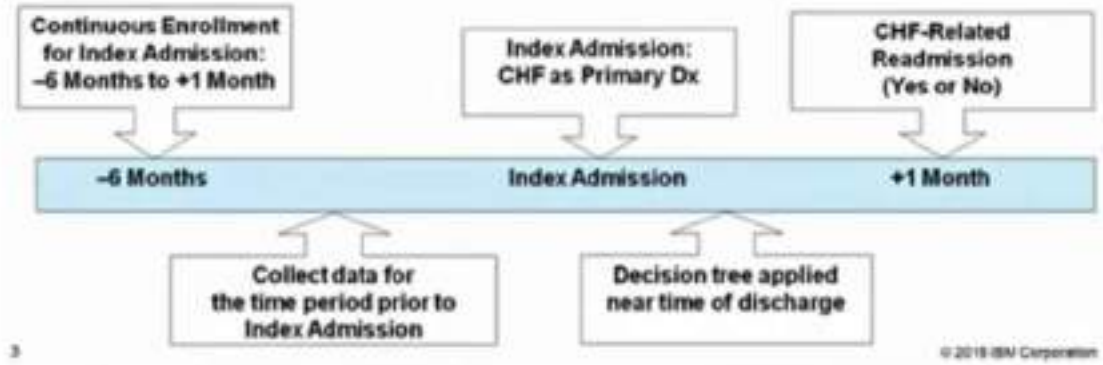
Case Study – Data preparation



Case Study – Defining readmission

Case Study – Defining CHF admission

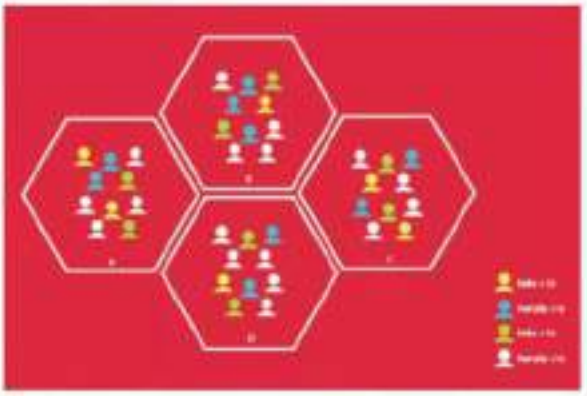
Define "CFF admission" and "CHF readmission"



Case Study – Aggregating records

Transactional records

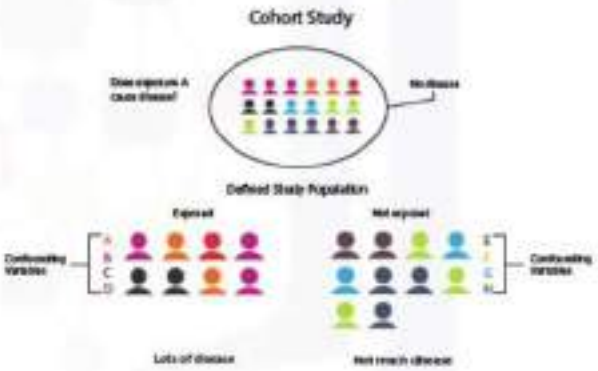
- Claims: professional provider, facility, pharmaceutical
- Inpatient & outpatient records: diagnoses, procedures, prescriptions, etc.
- Possibly thousands per patient, depending on clinical history



Case Study – Aggregating to patient level

Aggregate to patient level

- Roll up to 1 record per patient
- Create new columns representing the transaction
  - Outpatients visits/ Inpatient episodes: frequency, recency, diagnoses/length of stay, procedures, prescriptions
  - Comorbidities with CHF



Case Study – More or less data needed?

Literature review of important factors for CHF readmission

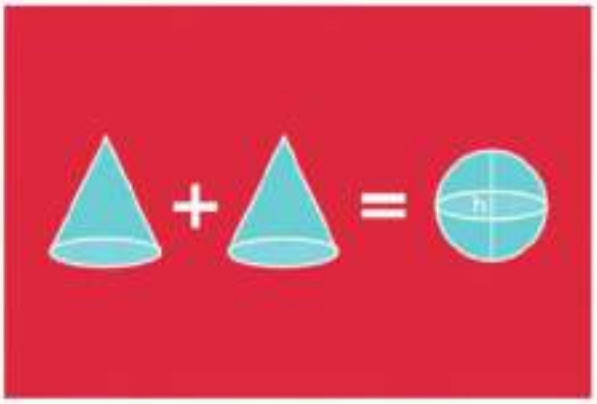
MORE LIKELY TO BE READMITTED

- Medicare / Medicaid insurance holders
- Comorbid conditions including:
  - Ischemic heart disease
  - Idiopathic Cardiomyopathy
  - Prior Cardiac surgery
  - Peripheral vascular disease
  - Diabetes mellitus
  - Anemia

LESS LIKELY TO BE READMITTED

- Patients treated at rural hospitals
- Patients discharged to skilled nursing facilities
- Patients receiving echocardiograms or cardiac catheterization

- Loop back to data collection stage and add additional data, if needed



Case Study – Completing the data set

Merge all data into one table

- One record per patient
- List of variables used in modeling
  - Target: CHF readmission with 30 days (Yes/No), following discharge from CHF hospitalization

Case Study – Creating new variables

Merge all data into one table

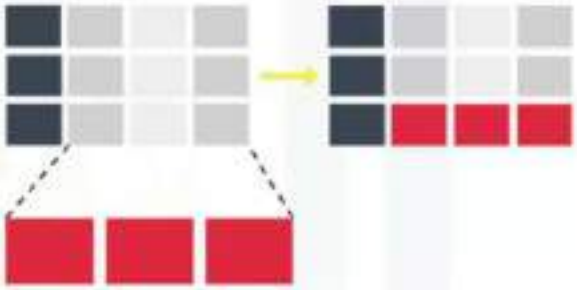
- One record per patient
- List of variables used in modeling

- Target
  - CHF readmission with 30 days (Yes/No), following discharge from CHF hospitalization
- Measures

Gender	Length of stay	CHF Diagnosis importance (primary, secondary, tertiary)
Age	Prior admissions	
Primary DRG	Line of business	

- Diagnosis flags (Y/N)

CHF	Atrial fibrillation	Pneumonia
Diabetes	Renal failure	Hypertension



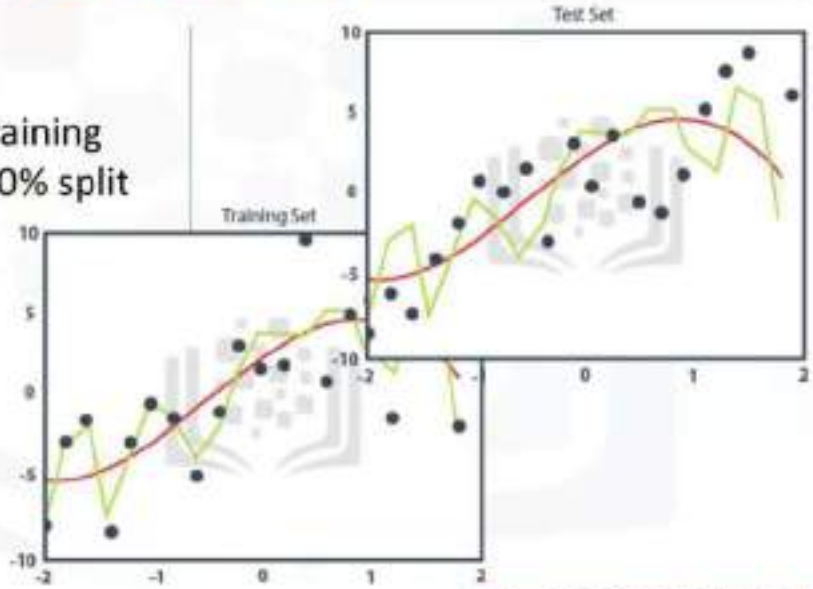
Case Study – Using training sets

Cohort: 2,343 patients

Randomly divided into training and testing sets: 70% / 30% split

Training: 1,640 patients

Testing: 703 patients

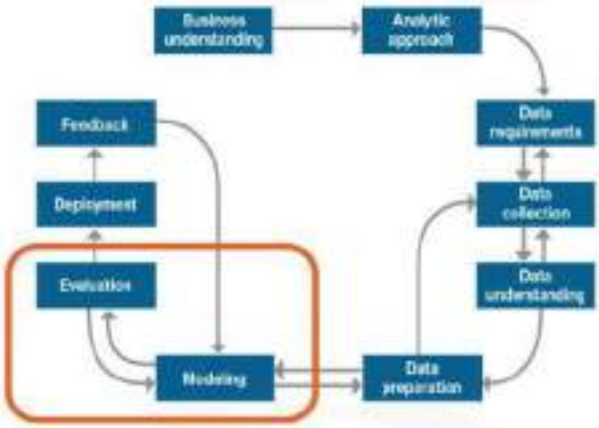


Data Science Methodology

Modeling - Concepts

Modeling – Sampling the food (the data)…

From Modeling to Evaluation



Modeling

- *In what way can the data be visualized to get to the answer that is required?*

Evaluation

- *Does the model used really answer the initial question or does it need to be adjusted?*

Data Modeling – Using Predictive or Descriptive?

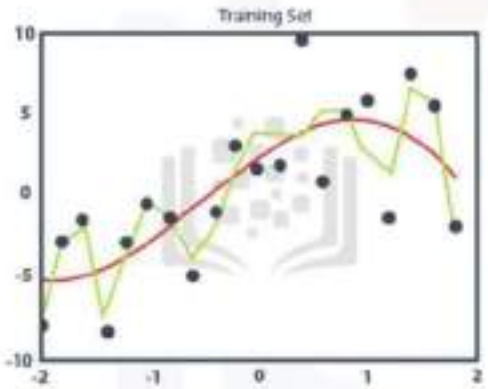
Descriptive Analytics

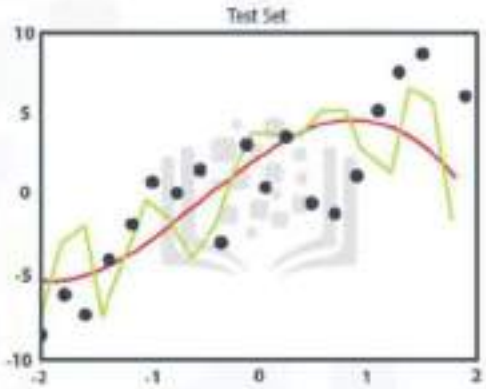
Predictive Analytics

IBM Developer

SKILLS NETWORK

Data Modeling – Using training / test sets





Does the model need to be calibrated?

IBM Developer

SKILLS NETWORK

Understanding the question

1. Understand the question at hand
2. Select an analytic approach or method to solve the problem
3. Obtain, understand, prepare, and model the data

Was the question answered?

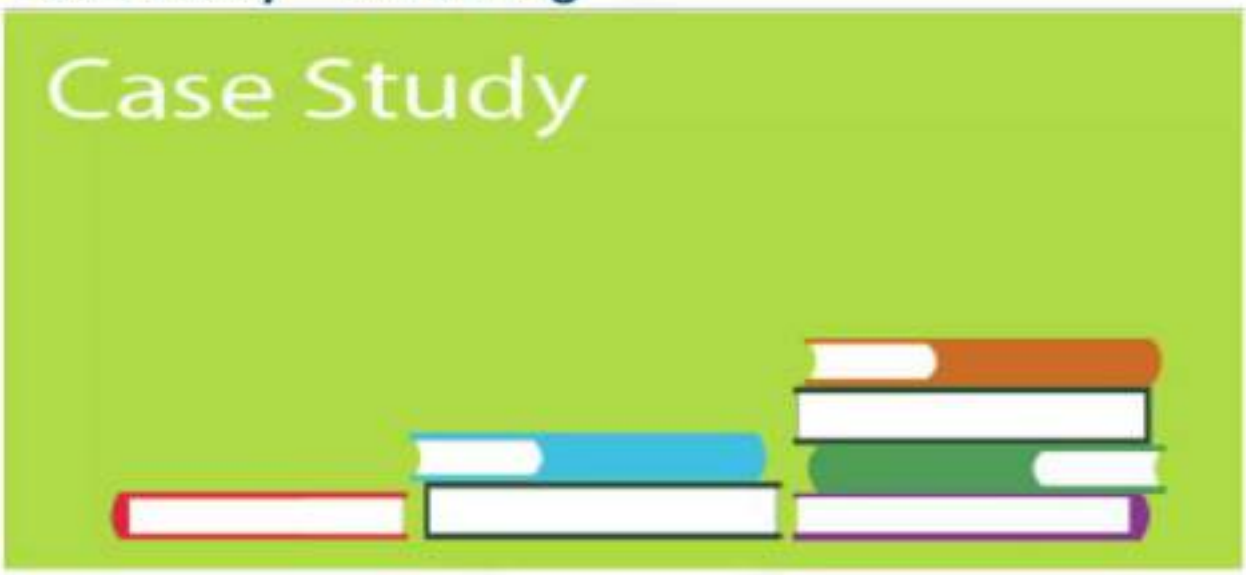
Data Science Methodology

Modeling – Case Study

IBM Developer

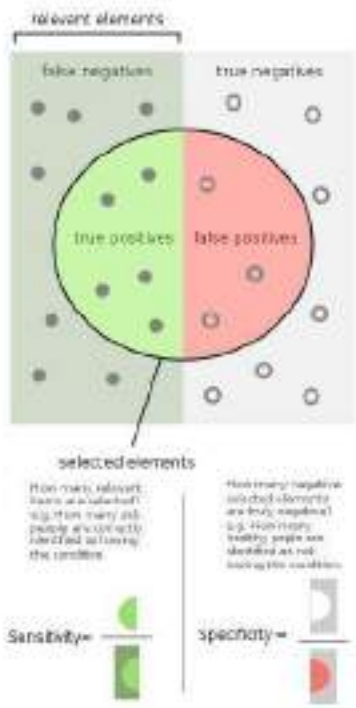
SKILLS NETWORK

Case Study – Modeling



IBM Developer

SKILLS NETWORK



$$\text{sensitivity} = \frac{\text{number of true positives}}{\text{number of true positives} + \text{number of false negatives}}$$

$$= \frac{\text{number of true positives}}{\text{total number of sick individuals in population}}$$

$$= \text{probability of a positive test given that the patient has the disease}$$

$$\text{specificity} = \frac{\text{number of true negatives}}{\text{number of true negatives} + \text{number of false positives}}$$

$$= \frac{\text{number of true negatives}}{\text{total number of well individuals in population}}$$

$$= \text{probability of a negative test given that the patient is well}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP}$$

https://en.wikipedia.org/wiki/Sensitivity_and_specificity

Type I vs. Type II Errors

- When a true, non-readmission is misclassified, and action is taken to reduce that patient's risk, the cost of that error is the wasted intervention.
  - A statistician calls this a type I error, or a false-positive.
- When a true readmission is misclassified, and no action is taken to reduce that risk, then the cost of that error is the readmission and all its attended costs, plus the trauma to the patient.
  - This is a type II error, or a false-negative.
- So we can see that the costs of the two different kinds of misclassification errors can be quite different.

Case Study – Analyzing the 1st model

Initial decision tree classification model

- Low accuracy on "Yes" outcome

Model	Relative Cost Y:N	Overall Accuracy (% correct Y & N)	Sensitivity (Y accuracy)	Specificity (N accuracy)
1	1:1	85%	45%	97%
2	9:1	49%	97%	35%
3	4:1	81%	68%	85%

Case Study – Analyzing the 2nd model

Second model

- High accuracy on "Yes" but poor on "No"

Model	Relative Cost Y:N	Overall Accuracy (% correct Y & N)	Sensitivity (Y accuracy)	Specificity (N accuracy)
1	1:1	85%	45%	97%
2	9:1	49%	97%	35%
3	4:1	81%	68%	85%

Case Study – Analyzing the 3rd model

Third model

- Better balance on "Yes" and "No" accuracy

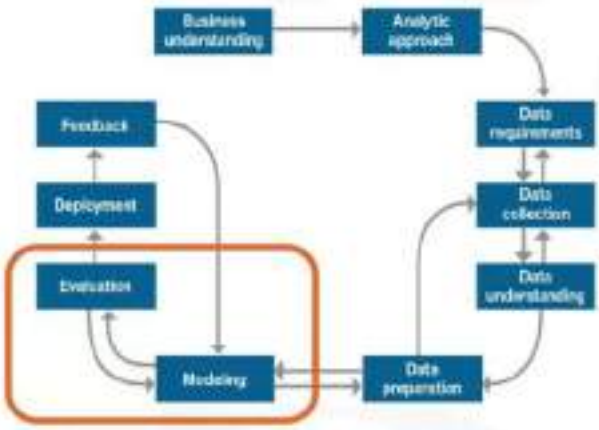
Model	Relative Cost Y:N	Overall Accuracy (% correct Y & N)	Sensitivity (Y accuracy)	Specificity (N accuracy)
1	1:1	85%	45%	97%
2	9:1	49%	97%	35%
3	4:1	81%	68%	85%

Data Science Methodology

Evaluation

Evaluating the model

From Modeling to Evaluation



Modeling

- *In what way can the data be visualized to get to the answer that is required?*

Evaluation

- *Does the model used really answer the initial question or does it need to be adjusted?*

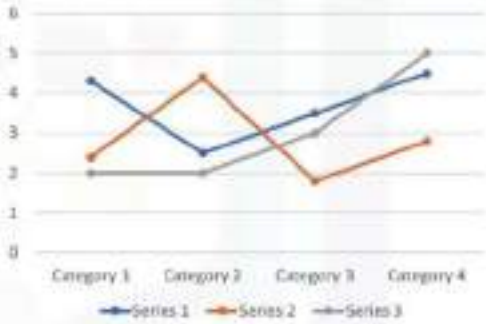
When and how to adjust the model?

Diagnostic measures

Predictive Model

Descriptive Model

Statistical Significance



Case Study – Applying the concepts

Case Study – Misclassification costs

Misclassification cost tuning

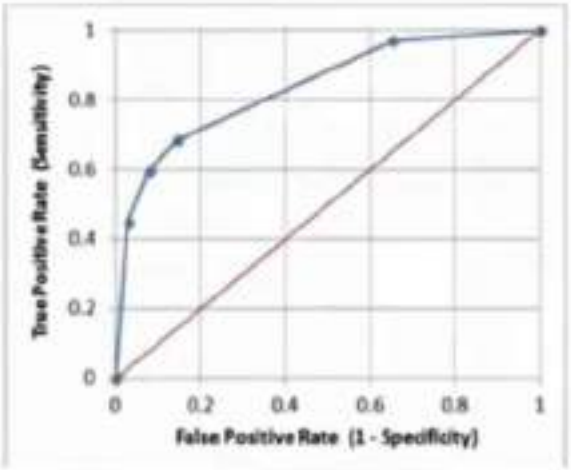
- Tune the relative misclassification costs
- Balance true-positive rate and false-positive rate for best model

Model	Relative Cost Y:N	True Positive Rate (Sensitivity)	Specificity (accuracy on N)	False Positive Rate (1 – Specificity)
1	1:1	0.45	0.97	0.03
2	1.5:1	0.60	0.92	0.08
3	4:1	0.68	0.85	0.15
4	9:1	0.97	0.35	0.65

Case Study – Using the ROC curve

Diagnostic tool for classification model evaluation

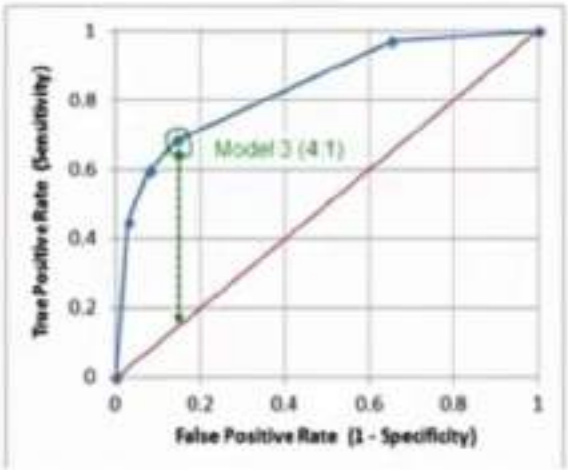
- Classification model performance
- True-Positive Rate vs False-Positive Rate
- Optimal model at maximum separation



Cost Study – Using the ROC curve

Diagnostic tool for classification model evaluation

- Classification model performance
- True-Positive Rate vs False-Positive Rate
- Optimal model at maximum separation

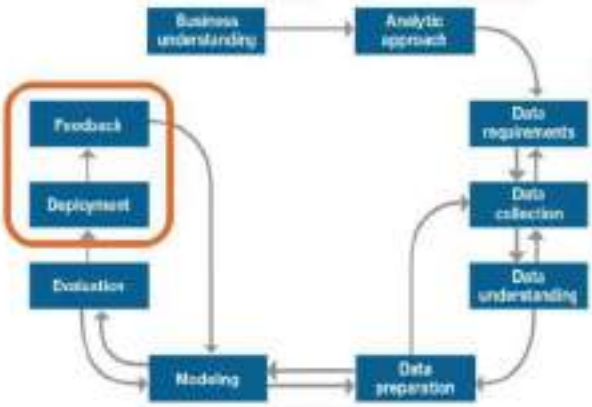


Data Science Methodology

Deployment

Deployment – Are stakeholders familiar with the new tool?

From Deployment to Feedback



Case Study – Understand the results

Assimilate knowledge for business

- Practical understanding of the meaning of model results
- Implications of model results for designing intervention actions

Case Study – Gathering application requirements

Application requirements

- Automated, near-real-time risk assessments of CHF inpatients
- Easy to use
- Automated data preparation and scoring
- Up-to-date risk assessment to help clinicians target high-risk patients

Case Study – Additional requirements?

Additional requirements

- Training for clinical staff
- Tracking / monitoring processes

Example 1 – Solution deployment

Hospitalization risk for juvenile diabetes patients

Example 2 – Solution deployment

Risk summary report by decision tree model note

Example 3 – Solution deployment

Individual patient risk report

Data Science Methodology

Feedback

IBM Developer

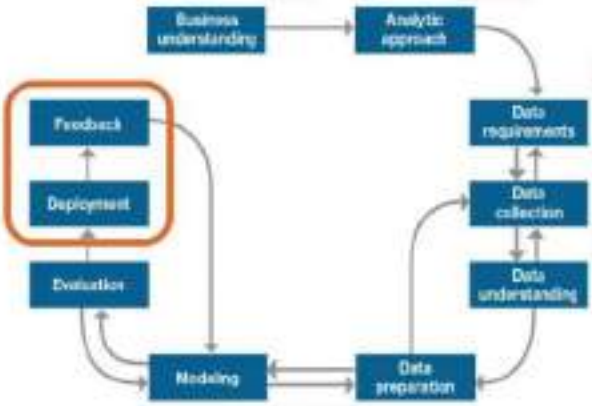
SKILLS NETWORK

Feedback – Problem solved? Question answered?

IBM Developer

SKILLS NETWORK

Feedback – Problem solved? Question answered?



From Deployment to Feedback

Once the model is evaluated and the data scientist is confident it will work, it is deployed and put to the ultimate test

- Actual real-time use in the field

Case Study – Assessing model performance

Define review process

- To measure results of applying the risk model to the CHF patient population
- Track patients who received intervention
  - Actual readmission outcomes
- Measure effectiveness of intervention
  - Compare readmission rates before & after model implementation

Case Study – Refinement

Refine model

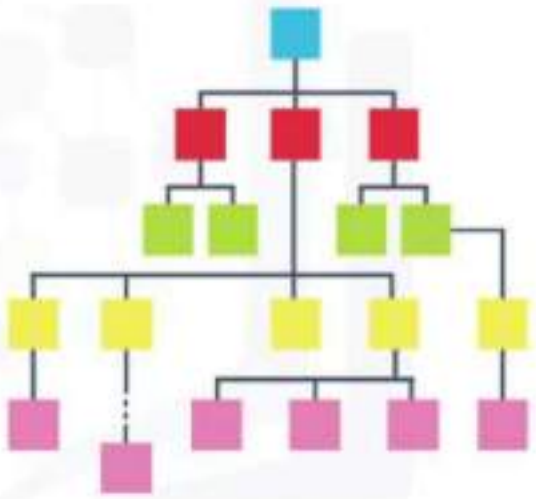
- Initial review after the first year of implementation
- Based on feedback data and knowledge gained
- Participation in intervention program
- Possibly incorporate detailed pharmaceutical data originally deferred
- Other possible refinements as yet unknown

Case Study – Redeployment

Review and refine intervention actions

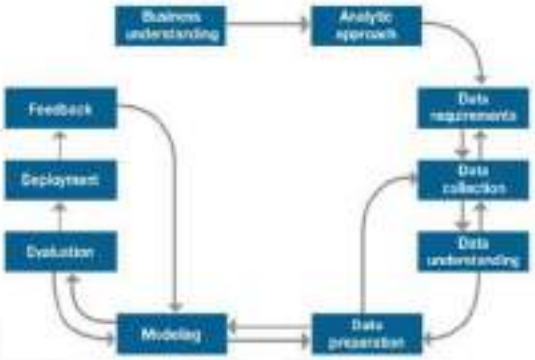
Redeploy

- Continue modeling, deployment, feedback, and refinement throughout the life of the intervention program



We've come to the end of the story

The methodology in a nutshell...



The **Data Science Methodology** aims to answer the following 10 questions in this prescribed sequence:

From problem to approach:

1. *What is the problem that you are trying to solve?*
2. *How can you use data to answer the question?*

Working with the data:

3. *What data do you need to answer the question?*
4. *Where is the data coming from (identify all sources) and how will you get it?*
5. *Is the data that you collected representative of the problem to be solved?*
6. *What additional work is required to manipulate and work with the data?*

Deriving the answer:

7. *In what way can the data be visualized to get to the answer that is required?*
8. *Does the model used really answer the initial question or does it need to be adjusted?*
9. *Can you put the model into practice?*
10. *Can you get constructive feedback into answering the question?*

Python for Data Analysis

Short Intro

BBM467 – Data Intensive Applications

Dr. Fuat Akal
akal@hacettepe.edu.tr

Lecture Overview

- Python Libraries to Analyse Data
  - Pandas
  - Numpy
  - Matplotlib

Python Libraries to Analyse Data

- Pandas
  - Provides data structures and operations for data (e.g. tables and time series) manipulation and analysis.
- Numpy
  - Provides means to work with multidimensional arrays.
- Matplotlib
  - A plotting library used to create high-quality graphs, charts, and figures.

Pandas

- A library that contains high-performance, easy-to-use data structures and data analysis tools.
- Some important aspects of Pandas
  - A fast and efficient DataFrame object for data manipulation with integrated indexing.
  - Tools for reading and writing data in different formats, e.g. csv, Excel, SQL Database.
  - Slicing, indexing, subsetting, merging and joining of huge datasets.
- Typically imported as `import pandas as pd` in Python programs

Create DataFrames using Dictionaries

```
import pandas as pd
data = { 'name': ['Fuat', 'Aykut', 'Erkut'],
         'midterm': [60, 85, 100],
         'final': [69, 90, 100],
         'attendance': [6, 10, 10]
}
df_bbm101 = pd.DataFrame(data)

print(df_bbm101.head()) # Prints top 5 rows
```

	name	midterm	final	attendance
0	Fuat	60	69	7
1	Aykut	85	90	10
2	Erkut	100	100	10

Same Thing, in Another Way

```
names = ['Fuat', 'Aykut', 'Erkut']
midterms = [60, 85, 100]
finals = [69, 90, 100]
attendances = [6, 10, 10]

list_labels = ['name', 'midterm', 'final', 'attendance']
list_cols = [names, midterms, finals, attendances]

zipped = list(zip(list_labels, list_cols))

print(zipped)     # [('name', ['Fuat', 'Aykut', 'Erkut']),
                  # ('midterm', [60, 85, 100]),
                  # ('final', [69, 90, 100]),
                  # ('attendance', [6, 10, 10])]

data = dict(zipped)

df_bbm101 = pd.DataFrame(data)
```

Broadcasting

```
df_bbm101['total'] = 0
# Adds new column to df and
# broadcasts 0 to entire column

print(df_bbm101.head())
```

	name	midterm	final	attendance	total
0	Fuat	60	69	6	0
1	Aykut	85	90	10	0
2	Erkut	100	100	10	0

Compute Columns

```
df_bbm101['total'] = df_bbm101['midterm']*0.3 + \
                     df_bbm101['final']*0.6 + \
                     df_bbm101['attendance']*0.1

df_bbm101.loc[(df_bbm101['total'] >= 60) &
         (df_bbm101['total'] < 70), 'grade'] = 'D'
…            # Code to compute Bs and Cs comes here
df_bbm101.loc[df_bbm101['total'] >= 90, 'grade'] = 'A'

print(df_bbm101.head())
```

	name	midterm	final	attendance	total	grade
0	Fuat	60	69	6	60.0	D
1	Aykut	85	90	10	80.5	B
2	Erkut	100	100	10	91.0	A

Beware that Fuat would not make it if he missed just one more lecture ;-)

Subsetting/Slicing Data

```
print(df_bbm101[['name', 'grade']])

print(df_bbm101.iloc[:, [0, 5]])

print(df_bbm101.iloc[:, [True, False, False, False,
                                False, True]])

# They all return the same thing
# name and grade columns of the df
# Same principle can be applied to rows as well
```

	name	grade
0	Fuat	D
1	Aykut	B
2	Erkut	A

DataFrames from CSV Files

```
name,midterm,final,attendance,total,grade
Fuat,60,69,6,60.0,D
Aykut,85,90,10,80.5,B
Erkut,100,100,10,91.0,A
```

file name: bbm101.csv

```
df_bbm101 = pd.read_csv('bbm101.csv')
print(df_bbm101.head())
```

	name	midterm	final	attendance	total	grade
0	Fuat	60	69	6	60.0	D
1	Aykut	85	90	10	80.5	B
2	Erkut	100	100	10	91.0	A

Indexing DataFrames

```
df_bbm101 = pd.read_csv('bbm101.csv', index_col ='name')
print(df_bbm101.head())
```

```
        midterm  final  attendance  total grade
name
 Fuat        60     69           6   60.0     D
Aykut        85     90          10   80.5     B
Erkut       100    100          10   91.0     A
```

```
print(df_bbm101.loc['Fuat'])
```

```
midterm       60
final         69
attendance     6
total         60
grade          D
Name: Fuat, dtype: object
```

```
print(df_bbm101.
   loc[['Aykut', 'Erkut']])
```

```
        midterm  final  attendance  total grade
name
Aykut        85     90          10   80.5     B
Erkut       100    100          10   91.0     A
```


Numpy

- A library for the Python programming language, adding support for large **multi-dimensional arrays and matrices**,
  - along with a large collection of high-level mathematical functions to operate on these arrays.
- A numpy array is a grid of values, **all of the same type**, and is indexed by a tuple of nonnegative integers.
- The number of dimensions is the **rank** of the array.
- The **shape** of an array is a tuple of integers giving the size of the array along each dimension.
- Typically imported as `import numpy as np` in Python programs

Creating Numpy Arrays

```
import numpy as np

a = np.array([1,2,3])      # Create a rank 1 array
print(type(a))             # <class 'numpy.ndarray'>
print(a.shape)             # (3,)
print(a)                   # [1 2 3]
print(a[0], a[1], a[2])    # 1 2 3


b = np.array([[1,2,3],[4,5,6]])    # Create a rank 2 array
print(b.shape)                     # (2, 3)
print(b)                           # [[1 2 3]
                                   #  [4 5 6]]
print(b[0, 0], b[0, 1], b[1, 0])   # 1 2 4
```


Miscellaneous Ways to Create Arrays

```
a = np.zeros((2,2))    # Create an array of all zeros
print(a)               # [[ 0.  0.]
                       #  [ 0.  0.]]

b = np.ones((1,2))     # Create an array of all ones
print(b)               # [[ 1.  1.]]

c = np.full((2,2), 7)  # Create a constant array
print(c)               # [[ 7.  7.]
                       #  [ 7.  7.]]

d = np.eye(2)          # Create a 2x2 identity matrix
print(d)               # [[ 1.  0.]
                       #  [ 0.  1.]]

e = np.random.random((2,2))  # Create an array filled with
                             # random values
print(e)                     # Might print
                             # [[ 0.91940167  0.08143941]
                             #  [ 0.68744134  0.87236687]]
```

Indexing Arrays

- Slicing
- Integer Indexing
- Boolean (or, Mask) Indexing

Slicing

- Similar to slicing Python lists.
- Since arrays may be multidimensional, you must specify a slice for each dimension of the array.
- Slices are views (not copies) of the original data.

Slicing Examples

```
a = np.array([[1, 2, 3, 4],     # Create a rank 2 array
              [5, 6, 7, 8],     # with shape (3, 4)
              [9, 10, 11, 12]])

print(a)                        # [[ 1  2  3  4]
                                #  [ 5  6  7  8]
                                #  [ 9 10 11 12]]

b = a[:2, 1:3]
print(b)                        # [[ 2  3 ]
                                #  [ 6  7 ]

print(a[1, :])                  # [5 6 7 8]

print(a[:, :-2])                # [[ 1  2]
                                #  [ 5  6]
                                #  [ 9 10]]
```


Integer Indexing

- NumPy arrays may be indexed with other arrays.
- Index arrays must be of integer type.
- Each value in the array indicates which value in the array to use in place of the index.
- Returns a copy of the original data.

Integer Indexing Examples

```
a = np.array([1, 2, 3, 4, 5, 6])
print(a)                                # [1 2 3 4 5 6]
print(a[[1, 3, 5]])                     # [2 4 6]


a = np.array([[1, 2], [3, 4], [5, 6]])
print(a)                                # [[ 1  2 ]
                                        #  [ 3  4 ]
                                        #  [ 5  6 ]]

# The returned array will have shape (3,)
print(a[[0, 1, 2], [0, 1, 0]])                # [1 4 5]
print(np.array([a[0, 0], a[1, 1], a[2, 0]]))  # [1 4 5]

# The same element from the source array can be reused
print(a[[0, 0], [1, 1]])                      # [2 2]
print(np.array([a[0, 1], a[0, 1]]))           # [2 2]
```


Boolean (or, Mask) Indexing

- Boolean array indexing lets you pick out arbitrary elements of an array.
- Frequently used to select the elements of an array that satisfy some condition.
  - Thus, called the mask indexing.

Boolean (or, Mask) Indexing Examples

```
a = np.array([1, 2, 3, 4, 5, 6])

bool_idx = (a > 2)
# Find the elements of a that are bigger than 2;
# this returns a numpy array of Booleans of the same
# shape as a, where each slot of bool_idx tells
# whether that element of a is > 2.

print(bool_idx)            # [False False  True
                           #            True  True  True]

# We use boolean array indexing to construct a rank 1 array
# consisting of the elements of a corresponding to the True
# values of bool_idx
print(a[bool_idx])         # [3 4 5 6]

# We can do all of the above in a single concise statement:
print(a[a > 2])            # [3 4 5 6]
```


Array Math

- Basic mathematical functions operate elementwise on arrays.

```
x = np.array([[1, 2], [3, 4]])
y = np.array([[5, 6], [7, 8]])

# Elementwise sum
print(x + y)
print(np.add(x, y))
# [[ 6  8]
#  [10 12]]

# Elementwise product
print(x * y)
print(np.multiply(x, y))
# [[ 5 12]
#  [21 32]]
```

Same principle holds for "np.divide, /" and "np.subtract, -"

Array Math (Cont'd)

```
x = np.array([[1, 2], [3, 4]])
y = np.array([[5, 6], [7, 8]])

v = np.array([9, 10]
w = np.array([11, 12])

# Inner product of vectors;
# both produce 219
print(v.dot(w))
print(np.dot(v, w))

# Matrix / vector product;
# both produce the rank 1
# array [29 67]
print(x.dot(v))
print(np.dot(x, v))

# Matrix / matrix product;
# both produce a rank 2 array
# [[19 22]
#  [43 50]]
print(x.dot(y))
print(np.dot(x, y))

# Transpose of x
# [[1 3]
#  [2 4]]
print(x.T)
```


Matplotlib

- Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments.
- Typically imported as `import matplotlib.pyplot as plt` in Python programs.
- Pyplot is a module of Matplotlib which provides simple functions to add plot elements like lines, images, text, etc.
- There are many plot types. Some of are more frequently used.

Why Build Visuals?

- For exploratory data analysis
- Communicate data clearly
- Share unbiased representation of data
- A picture is worth a thousand words ☺

Make a Simple Plot

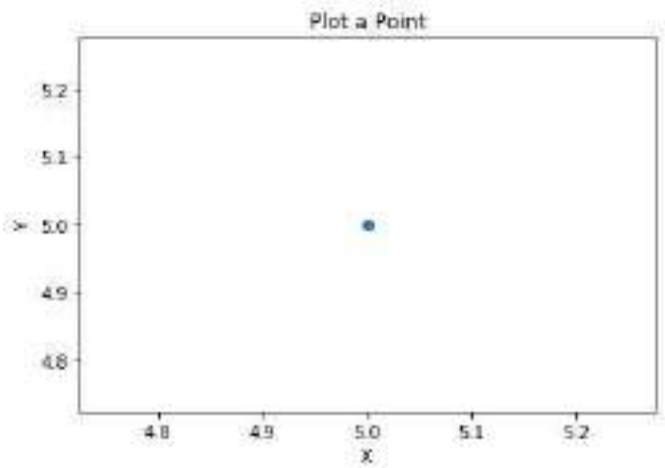
```
import matplotlib.pyplot as plt

plt.plot(5, 5, 'o')

plt.title("Plot a Point")

plt.xlabel("X")
plt.ylabel("Y")

plt.show()
```



Plot a Simple Line

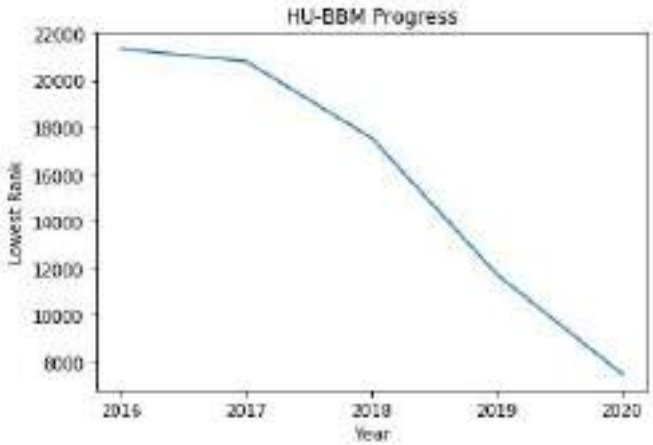
```
import matplotlib.pyplot as plt

year = ['2016', '2017', '2018', '2019', '2020']
lowest_rank = [21358, 20816, 17555, 11743, 7500]

plt.plot(year, lowest_rank)

plt.title(”HU-BBM Progress")
plt.xlabel('Year')
plt.ylabel('Lowest Rank')

plt.show()
```



The Problem

Why Data Analysis?

- Data is everywhere.
- Data analysis/data science helps us answer questions from data.
- **Data analysis** plays an important role in:
  - Discovering useful information
  - Answering questions
  - Predicting future or the unknown

Tom wants to sell his car

Tom

How much money should he sell his car for?

The price he sets should not be too high, but not too low either.

COGNITIVE CLASS.ai

Estimate used car prices

How can we help Tom determine the best price for his car?

- Is there data on the prices of other cars and their characteristics?
- What features of cars affect their prices?
  - Color? Brand? Horsepower? Something else?
- Asking the right questions in terms of data

Tom

COGNITIVE CLASS.ai

Data on used car prices

```
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,13495
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,16500
1,?,alfa-romero,gas,std,two,hatchback,rwd,front,94.50,171.20,65.50,52.40,2823,ohcv,six,152,mpfi,2.68,3.47,9.00,154,5000,19,26,16500
2,164,audi,gas,std,four,sedan,fwd,front,99.80,176.60,66.20,54.30,2337,ohc,four,109,mpfi,3.19,3.40,10.00,102,5500,24,30,13950
2,164,audi,gas,std,four,sedan,4wd,front,99.40,176.60,66.40,54.30,2824,ohc,five,136,mpfi,3.19,3.40,8.00,115,5500,18,22,17450
2,?,audi,gas,std,two,sedan,fwd,front,99.80,177.30,66.30,53.10,2507,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,15250
1,158,audi,gas,std,four,sedan,fwd,front,105.80,192.70,71.40,55.70,2844,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,17710
1,?,audi,gas,std,four,wagon,fwd,front,105.80,192.70,71.40,55.70,2954,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,18920
1,158,audi,gas,turbo,four,sedan,fwd,front,105.80,192.70,71.40,55.90,3086,ohc,five,131,mpfi,3.13,3.40,8.30,140,5500,17,20,23875
0,?,audi,gas,turbo,two,hatchback,4wd,front,99.50,178.20,67.90,52.00,3053,ohc,five,131,mpfi,3.13,3.40,7.00,160,5500,16,22,?
2,192,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16430
0,192,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16925
0,188,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2710,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,20970
0,188,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2765,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,21105
1,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3055,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,20,25,24565
0,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3230,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,30760
0,?,bmw,gas,std,two,sedan,rwd,front,103.50,193.80,67.90,53.70,3380,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,41315
0,?,bmw,gas,std,four,sedan,rwd,front,110.00,197.00,70.90,56.30,3505,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,15,20,36880
2,121,chevrolet,gas,std,two,hatchback,fwd,front,88.40,141.10,60.30,53.20,1488,1,three,61,2bbl,2.91,3.03,9.50,48,5100,47,53,5151
```

https://archive.ics.uci.edu/ml/machine-learning-databases/autos/

COGNITIVE CLASS.ai

© 2017 IBM Corporation

Understanding the Data

COGNITIVE CLASS.ai

Dataset – Used Automobiles (CSV)

```
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,13495
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,16500
1,?,alfa-romero,gas,std,two,hatchback,rwd,front,94.50,171.20,65.50,52.40,2823,ohcv,six,152,mpfi,2.68,3.47,9.00,154,5000,19,26,16500
2,164,audi,gas,std,four,sedan,fwd,front,99.80,176.60,66.20,54.30,2337,ohc,four,109,mpfi,3.19,3.40,10.00,102,5500,24,30,13950
2,164,audi,gas,std,four,sedan,4wd,front,99.40,176.60,66.40,54.30,2824,ohc,five,136,mpfi,3.19,3.40,8.00,115,5500,18,22,17450
2,?,audi,gas,std,two,sedan,fwd,front,99.80,177.30,66.30,53.10,2507,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,15250
1,158,audi,gas,std,four,sedan,fwd,front,105.80,192.70,71.40,55.70,2844,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,17710
1,?,audi,gas,std,four,wagon,fwd,front,105.80,192.70,71.40,55.70,2954,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,18920
1,158,audi,gas,turbo,four,sedan,fwd,front,105.80,192.70,71.40,55.90,3086,ohc,five,131,mpfi,3.13,3.40,8.30,140,5500,17,20,23875
0,?,audi,gas,turbo,two,hatchback,4wd,front,99.50,178.2?,67.90,52.00,3053,ohc,five,131,mpfi,3.13,3.40,7.00,160,5500,16,22,?
2,192,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16430
0,192,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16925
0,188,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2710,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,20970
0,188,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2765,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,21105
1,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3055,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,20,25,24565
0,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3230,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,30760
0,?,bmw,gas,std,two,sedan,rwd,front,103.50,193.80,67.90,53.70,3380,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,41315
0,?,bmw,gas,std,four,sedan,rwd,front,110.00,197.00,70.90,56.30,3505,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,15,20,36880
2,121,chevrolet,gas,std,two,hatchback,fwd,front,88.40,141.10,60.30,53.20,1488,l,three,61,2bbl,2.91,3.03,9.50,48,5100,47,53,5151
```

https://archive.ics.uci.edu/ml/machine-learning-databases/autos/

Each of the attributes in the dataset

No.	Attribute name	attribute range	No.	Attribute name	attribute range
1	symboling	-3, -2, -1, 0, 1, 2, 3.	14	curb-weight	continuous from 1488 to 4066.
2	normalized-losses	continuous from 65 to 256.	15	engine-type	dohc, dohcv, l, ohc, ohcf, ohcv, rotor.
3	make	audi, bmw, etc.	16	num-of-cylinders	eight, five, four, six, three, twelve, two.
4	fuel-type	diesel, gas.	17	engine-size	continuous from 61 to 326.
5	aspiration	std, turbo.	18	fuel-system	1bbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi.
6	num-of-doors	four, two.	19	bore	continuous from 2.54 to 3.94.
7	body-style	hardtop, wagon, etc.	20	stroke	continuous from 2.07 to 4.17.
8	drive-wheels	4wd, fwd, rwd.	21	compression-ratio	continuous from 7 to 23.
9	engine-location	front, rear.	22	horsepower	continuous from 48 to 288.
10	wheel-base	continuous from 86.6 120.9.	23	peak-rpm	continuous from 4150 to 6600.
11	length	continuous from 141.1 to 208.1.	24	city-mpg	continuous from 13 to 49.
12	width	continuous from 60.3 to 72.3.	25	highway-mpg	continuous from 16 to 54.
13	height	continuous from 47.8 to 59.8.	26	price	continuous from 5118 to 45400.

Attributes description: https://archive.ics.uci.edu/ml/machine-learning-databases/autos/imports-85.names

Each of the attributes in the dataset

No.	Attribute name	attribute range	No.	Attribute name	attribute range
1	symboling	-3, -2, -1, 0, 1, 2, 3.	14	curb-weight	continuous from 1488 to 4066.
2	normalized-losses	continuous from 65 to 256.	15	engine-type	dohc, dohcv, l, ohc, ohcf, ohcv, rotor.
3	make	audi, bmw, etc.	16	num-of-cylinders	eight, five, four, six, three, twelve, two.
4	fuel-type	diesel, gas.	17	engine-size	continuous from 61 to 326.
5	aspiration	std, turbo.	18	fuel-system	1bbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi.
6	num-of-doors	four, two.	19	bore	continuous from 2.54 to 3.94.
7	body-style	hardtop, wagon, etc.	20	stroke	continuous from 2.07 to 4.17.
8	drive-wheels	4wd, fwd, rwd.	21	compression-ratio	continuous from 7 to 23.
9	engine-location	front, rear.	22	horsepower	continuous from 48 to 288.
10	wheel-base	continuous from 86.6 120.9.	23	peak-rpm	continuous from 4150 to 6600.
11	length	continuous from 141.1 to 208.1.	24	city-mpg	continuous from 13 to 49.
12	width	continuous from 60.3 to 72.3.	25	highway-mpg	continuous from 16 to 54.
13	height	continuous from 47.8 to 59.8.	26	price	continuous from 5118 to 45400.

Target (Label)

COGNITIVE CLASS.ai

Attributes description: https://archive.ics.uci.edu/ml/machine-learning-databases/autos/imports-85.names

Python packages for data science

COGNITIVE CLASS.ai

Scientifics Computing Libraries in Python

1. **Scientifics Computing** Libraries

Pandas
(Data structures & tools)

NumPy
(Arrays & matrices)

SciPy
(Integrals, solving differential equations, optimization)

Visualization Libraries in Python

2. **Visualization** Libraries

Matplotlib
(plots & graphs, most popular)

Seaborn
(plots : heat maps, time series, violin plots)

Algorithmic Libraries in Python

3. **Algorithmic** libraries

Scikit-learn
(Machine Learning : regression, classification,...)

Statsmodels
(Explore data, estimate statistical models, and perform statistical tests.)

Importing and Exporting Data in Python

Importing Data

- Process of loading and reading data into Python from various resources.
- **Two important properties:**
  - Format
    - various formats: .csv, .json, .xlsx, .hdf ....
  - File Path of dataset
    - Computer: /Desktop/mydata.csv
    - Internet: https://archive.ics.uci.edu/autos/imports-85.data

Getting Data

```
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,13495
3,?,alfa-romero,gas,std,two,convertible,rwd,front,88.60,168.80,64.10,48.80,2548,dohc,four,130,mpfi,3.47,2.68,9.00,111,5000,21,27,16500
1,?,alfa-romero,gas,std,two,hatchback,rwd,front,94.50,171.20,65.50,52.40,2823,ohcv,six,152,mpfi,2.68,3.47,9.00,154,5000,19,26,16500
2,164,audi,gas,std,four,sedan,fwd,front,99.80,176.80,66.20,54.30,2337,ohc,four,109,mpfi,3.19,3.40,10.00,102,5500,24,30,13950
2,164,audi,gas,std,four,sedan,4wd,front,99.40,176.60,66.40,54.30,2824,ohc,five,136,mpfi,3.19,3.40,8.00,115,5500,18,22,17450
2,?,audi,gas,std,two,sedan,fwd,front,99.80,177.30,66.30,53.10,2507,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,15250
1,158,audi,gas,std,four,sedan,fwd,front,105.80,192.70,71.40,55.70,2844,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,17710
1,?,audi,gas,std,four,wagon,fwd,front,105.80,192.70,71.40,55.70,2954,ohc,five,136,mpfi,3.19,3.40,8.50,110,5500,19,25,18920
1,158,audi,gas,turbo,four,sedan,fwd,front,105.80,192.70,71.40,55.90,3086,ohc,five,131,mpfi,3.13,3.40,8.30,140,5500,17,20,23875
0,?,audi,gas,turbo,two,hatchback,4wd,front,99.50,178.20,67.90,52.00,3053,ohc,five,131,mpfi,3.13,3.40,7.00,160,5500,16,22,?
2,192,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16430
0,192,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2395,ohc,four,108,mpfi,3.50,2.80,8.80,101,5800,23,29,16925
0,188,bmw,gas,std,two,sedan,rwd,front,101.20,176.80,64.80,54.30,2710,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,20970
0,188,bmw,gas,std,four,sedan,rwd,front,101.20,176.80,64.80,54.30,2765,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,21,28,21105
1,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3055,ohc,six,164,mpfi,3.31,3.19,9.00,121,4250,20,25,24565
0,?,bmw,gas,std,four,sedan,rwd,front,103.50,189.00,66.90,55.70,3230,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,30760
0,?,bmw,gas,std,two,sedan,rwd,front,103.50,193.80,67.90,53.70,3380,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,16,22,41315
0,?,bmw,gas,std,four,sedan,rwd,front,110.00,197.00,70.90,56.30,3505,ohc,six,209,mpfi,3.62,3.39,8.00,182,5400,15,20,36880
2,121,chevrolet,gas,std,two,hatchback,fwd,front,88.40,141.10,60.30,53.20,1488,l,three,61,2bbl,2.91,3.03,9.50,48,5100,47,53,5151
1,98,chevrolet,gas,std,two,hatchback,fwd,front,94.50,155.90,63.60,52.00,1874,ohc,four,90,2bbl,3.03,3.11,9.60,70,5400,38,43,6295
0,81,chevrolet,gas,std,four,sedan,fwd,front,94.50,158.80,63.60,52.00,1909,ohc,four,90,2bbl,3.03,3.11,9.60,70,5400,38,43,6575
1,118,dodge,gas,std,two,hatchback,fwd,front,93.70,157.30,63.80,50.80,1876,ohc,four,90,2bbl,2.97,3.23,9.41,68,5500,37,41,5572
1,118,dodge,gas,std,two,hatchback,fwd,front,93.70,157.30,63.80,50.80,1876,ohc,four,90,2bbl,2.97,3.23,9.40,68,5500,31,38,6377
1,118,dodge,gas,turbo,two,hatchback,fwd,front,93.70,157.30,63.80,50.80,2128,ohc,four,98,mpfi,3.03,3.39,7.60,102,5500,24,30,7957
1,148,dodge,gas,std,four,hatchback,fwd,front,93.70,157.30,63.80,50.60,1967,ohc,four,90,2bbl,2.97,3.23,9.40,68,5500,31,38,6229
1,148,dodge,gas,std,four,sedan,fwd,front,93.70,157.30,63.80,50.60,1989,ohc,four,90,2bbl,2.97,3.23,9.40,68,5500,31,38,6692
1,148,dodge,gas,std,four,sedan,fwd,front,93.70,157.30,63.80,50.60,1989,ohc,four,90,2bbl,2.97,3.23,9.40,68,5500,31,38,7609
1,148,dodge,gas,turbo,?,sedan,fwd,front,93.70,157.30,63.80,50.60,2191,ohc,four,98,mpfi,3.03,3.39,7.60,102,5500,24,30,8558
```

data source : https://archive.ics.uci.edu/ml/machine-learning-databases/autos/imports-85.data

Importing a CSV into Python

```
import pandas as pd

url = "https://archive.ics.uci.edu/ml/machine-learningdatabases/autos/imports-85.data"

df = pd.read_csv(url)
```

COGNITIVE CLASS.ai

Importing a CSV without a header

```
import pandas as pd

url = "https://archive.ics.uci.edu/ml/machine-learning-databases/autos/imports-85.data"

df = pd.read_csv(url, header = None)
```

COGNITIVE CLASS.ai

Printing the dataframe in Python

- `df` prints the entire dataframe (not recommended for large datasets)
- `df.head(n)` to show the first *n* rows of data frame.
- `df.tail(n)` shows the bottom *n* rows of data frame.

```
df.head()
```

	0	1	2	3	4	5	6	7	8	9	...	16	17	18	19	20	21	22	23	24	25
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111	5000	21	27	13495
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111	5000	21	27	16500
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	154	5000	19	26	16500
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	102	5500	24	30	13950
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	115	5500	18	22	17450

COGNITIVE CLASS.ai

Printing the dataframe in Python

- `df` prints the entire dataframe (not recommended for large datasets)
- `df.head(n)` to show the first *n* rows of data frame.
- `df.tail(n)` shows the bottom *n* rows of data frame.

```
df.head()
```

Header

n=5

	0	1	2	3	4	5	6	7	8	9	...	16	17	18	19	20	21	22	23	24	25
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111	5000	21	27	13495
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	111	5000	21	27	16500
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	154	5000	19	26	16500
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	102	5500	24	30	13950
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	115	5500	18	22	17450

COGNITIVE CLASS.ai

Adding headers

- Replace default header (by `df.columns = headers`)

```
headers = ["symboling","normalized-losses","make","fuel-type","aspiration", "num-of-doors","body-style",
"drive-wheels","engine-location","wheel-base", "length","width","height","curb-weight","engine-type",
"num-of-cylinders", "engine-size","fuel-system","bore","stroke","compression-ratio","horsepower", "peak-
rpm","city-mpg","highway-mpg","price"]
```

```
df.columns=headers
```

```
df.head(5)
```

COGNITIVE CLASS.ai

Adding headers

- Replace default header (by `df.columns = headers`)

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke	compression-ratio	h
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	1
1	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2.68	9.0	1
2	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3.47	9.0	1
3	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3.40	10.0	1
4	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3.40	8.0	1

COGNITIVE CLASS.ai

Exporting a Pandas dataframe to CSV

- Preserve progress anytime by saving modified dataset using

```
path="C:\Windows\…\automobile.csv"
df.to_csv(path)
```

COGNITIVE CLASS.ai

Exporting to different formats in Python

Data Format	Read	Save
csv	pd.read_csv()	df.to_csv()
json	pd.read_json()	df.to_json()
Excel	pd.read_excel()	df.to_excel()
sql	pd.read_sql()	df.to_sql()

COGNITIVE CLASS.ai

Getting started analyzing data in Python

COGNITIVE CLASS.ai

Basic insights from the data

- Understand your data before you begin any analysis
- Should check:
  - Data Types
  - Data Distribution
- Locate potential issues with the data

COGNITIVE CLASS.ai

Basic Insights of Dataset - Data Types

Pandas Type	Native Python Type	Description
object	string	numbers and strings
int64	int	Numeric characters
float64	float	Numeric characters with decimals
datetime64, timedelta[ns]	N/A (but see the datetime module in Python's standard library)	time data.

Why check data types?

- potential info and type mismatch
- compatibility with python methods

COGNITIVE CLASS.ai

Basic Insights of Dataset - Data Types

- In pandas, we use `dataframe.dtypes` to check data types

```
df.dtypes
```

```
symboling              int64
normalized-losses     object
make                  object
fuel-type             object
aspiration            object
num-of-doors          object
body-style            object
drive-wheels          object
engine-location       object
wheel-base           float64
length               float64
width                float64
height               float64
curb-weight            int64
engine-type           object
num-of-cylinders      object
engine-size            int64
fuel-system           object
bore                  object
stroke                object
compression-ratio    float64
horsepower            object
peak-rpm              object
city-mpg               int64
highway-mpg            int64
price                 object
dtype: object
```

COGNITIVE CLASS.ai

Basic Insights of Dataset - Data Types

- In pandas, we use `dataframe.dtypes` to check data types

```
df.dtypes
```

```
symboling                int64
normalized-losses       object
make                    object
fuel-type               object
aspiration              object
num-of-doors            object
body-style              object
drive-wheels            object
engine-location         object
wheel-base             float64
length                 float64
width                  float64
height                 float64
curb-weight              int64
engine-type             object
num-of-cylinders        object
engine-size              int64
fuel-system             object
bore                    object
stroke                  object
compression-ratio      float64
horsepower              object
peak-rpm                object
city-mpg                 int64
highway-mpg              int64
price                   object
dtype: object
```

COGNITIVE CLASS.ai

dataframe.describe()

- Returns a statistical summary

```
df.describe()
```

	symboling	wheel-base	length	width	height	curb-weight	engine-size	compression-ratio	city-mpg	highway-mpg
count	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000	205.000000
mean	0.834146	98.756585	174.049268	65.907805	53.724878	2555.565854	126.907317	10.142537	25.219512	30.751220
std	1.245307	6.021776	12.337289	2.145204	2.443522	520.680204	41.642693	3.972040	6.542142	6.886443
min	-2.000000	86.600000	141.100000	60.300000	47.800000	1488.000000	61.000000	7.000000	13.000000	16.000000
25%	0.000000	94.500000	166.300000	64.100000	52.000000	2145.000000	97.000000	8.600000	19.000000	25.000000
50%	1.000000	97.000000	173.200000	65.500000	54.100000	2414.000000	120.000000	9.000000	24.000000	30.000000
75%	2.000000	102.400000	183.100000	66.900000	55.500000	2935.000000	141.000000	9.400000	30.000000	34.000000
max	3.000000	120.900000	208.100000	72.300000	59.800000	4066.000000	326.000000	23.000000	49.000000	54.000000

COGNITIVE CLASS.ai

dataframe.describe(include="all")

- Provides full summary statistics

```
df.describe(include="all")
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke
count	205.000000	200	205	205	205	205	205	205	205	205.000000	...	205.000000	205	205	205
unique	NaN	52	22	2	2	3	5	3	2	NaN	...	NaN	8	38	37
top	NaN	7	toyota	gas	std	four	sedan	fwd	front	NaN	...	NaN	mpfi	3.62	3.40
freq	NaN	41	32	185	168	114	96	120	202	NaN	...	NaN	94	23	20
mean	0.834146	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	98.756585	...	126.907317	NaN	NaN	NaN
std	1.245307	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6.021776	...	41.642693	NaN	NaN	NaN
min	-2.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	86.600000	...	61.000000	NaN	NaN	NaN
25%	0.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	94.500000	...	97.000000	NaN	NaN	NaN
50%	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	97.000000	...	120.000000	NaN	NaN	NaN
75%	2.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	102.400000	...	141.000000	NaN	NaN	NaN
max	3.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	120.900000	...	326.000000	NaN	NaN	NaN

COGNITIVE CLASS

dataframe.describe(include="all")

- Provides full summary statistics

```
df.describe(include="all")
```

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location	wheel-base	...	engine-size	fuel-system	bore	stroke
count	205.000000	200	205	205	205	205	205	205	205	205.000000	...	205.000000	205	205	205
unique	NaN	52	22	2	2	3	5	3	2	NaN	...	NaN	8	38	37
top	NaN	7	toyota	gas	std	four	sedan	fwd	front	NaN	...	NaN	mpfi	3.62	3.40
freq	NaN	41	32	185	168	114	96	120	202	NaN	...	NaN	94	23	20
mean	0.834146	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	98.756585	...	126.907317	NaN	NaN	NaN
std	1.245307	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6.021776	...	41.642693	NaN	NaN	NaN
min	-2.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	86.600000	...	61.000000	NaN	NaN	NaN
25%	0.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	94.500000	...	97.000000	NaN	NaN	NaN
50%	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	97.000000	...	120.000000	NaN	NaN	NaN
75%	2.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	102.400000	...	141.000000	NaN	NaN	NaN
max	3.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	120.900000	...	326.000000	NaN	NaN	NaN

COGNITIVE CLASS

Basic Insights of Dataset - Info

`dataframe.info()` provides a concise summary of your DataFrame.

```
df.info()
```

Row Number

Data Pre-processing in Python

Data Pre-processing

Also known as:

Data Cleaning, Data Wrangling

The process of converting or mapping data from the initial "raw" form into another format, in order to prepare the data for further analysis.

COGNITIVE CLASS.ai

Learning Objectives

- Identify and handle missing values
- Data Formatting
- Data Normalization (centering /scaling)
- Data Binning
- Turning Categorical values to numeric variables

COGNITIVE CLASS.ai

Dealing with Missing Values in Python

Missing Values

- What is missing value?
- Missing values occur when no data value is stored for a variable (feature) in an observation.
- Could be represented as "?", "N/A", 0 or just a blank cell.

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engine-location
0	3	NaN ←	alfa-romero	gas	std	two	convertible	rwd	front

How to deal with missing data?

Check with the data collection source

Drop the missing values

- drop the variable
- drop the data entry

Replace the missing values

- replace it with an average (of similar datapoints)
- replace it by frequency
- replace it based on other functions

Leave it as missing data

How to drop missing values in Python

- Use `dataframes.dropna()`:

highway-mpg	price
...	...
20	23875
22	NaN
29	16430
...	...

How to drop missing values in Python

- Use `dataframes.dropna()`:

highway-mpg	price
...	...
20	23875
~~22~~	~~NaN~~
29	16430
...	...

→

highway-mpg	price
...	...
20	23875
29	16430
...	...

`axis=0` drops the entire row
`axis=1` drops the entire column

How to drop missing values in Python

- Use `dataframes.dropna()`:

highway-mpg	price
...	...
20	23875
~~22~~	~~NaN~~
29	16430
...	...

→

highway-mpg	price
...	...
20	23875
29	16430
...	...

`axis=0` drops the entire row
`axis=1` drops the entire column

```
df.dropna(subset=["price"], axis=0, inplace = True)
```

How to drop missing values in Python

- Use `dataframes.dropna()`:

highway-mpg	price
...	...
20	23875
~~22~~	~~NaN~~
29	16430
...	...

highway-mpg	price
...	...
20	23875
29	16430
...	...

axis=0 drops the entire row
axis=1 drops the entire column

```
df.dropna(subset=["price"], axis=0, inplace = True)
df = df.dropna(subset=["price"], axis=0)
```

COGNITIVE CLASS.ai

Don't Forget

```
df.dropna(subset=["price"], axis=0)
```

```
df.dropna(subset=["price"], axis=0, inplace = True)
```

http://pandas.pydata.org/

COGNITIVE CLASS.ai

© 2017 IBM Corporation

How to replace missing values in Python

Use `dataframe.replace(missing_value, new_value):`

normalized-losses	make
...	...
164	audi
164	audi
NaN	audi
158	audi
...	...

→

normalized-losses	make
...	...
164	audi
164	audi
162	audi
158	audi
...	...

```
mean = df["normalized-losses"].mean()
df["normalized-losses"].replace(np.nan, mean)
```


How to deal with missing data?

Check with the data collection source

Drop the missing values

- drop the variable
- drop the data entry

Replace the missing values

- replace it with an average (of similar datapoints)
- replace it by frequency
- replace it based on other functions

Leave it as missing data

Data Formatting in Python

Data Formatting

- Data are usually collected from different places and stored in different formats.
- Bringing data into a common standard of expression allows users to make meaningful comparison.

Non-formatted:
- confusing
- hard to aggregate
- hard to compare

City
NY
New York
N.Y
N.Y

City
New York
New York
New York
New York

Formatted:
- more clear
- easy to aggregate
- easy to compare

Applying calculations to an entire column

- Convert "mpg" to "L/100km" in Car dataset.

city-mpg
21
21
19
...

→

city-L/100km
11.2
11.2
12.4
...

```
df["city-mpg"]= 235/df["city-mpg"]

df.rename(columns={"city_mpg": "city-L/100km"}, inplace=True)
```


Incorrect data types

- Sometimes the wrong data type is assigned to a feature.

```
df["price"].tail(5)

200    16845
201    19045
202    21485
203    22470
204    22625
Name: price, dtype: object
```

Data Types in Python and Pandas

- There are many data types in pandas
- Objects : "A", "Hello"..
- Int64 : 1,3,5
- Float64 : 2.123, 632.31,0.12

COGNITIVE CLASS.ai

Correcting data types

To *identify* data types:

- Use **`dataframe.dtypes()`** to identify data type.

To *convert* data types:

- Use **`dataframe.astype()`** to convert data type.

Example: convert data type to integer in column "price"

```
df["price"] = df["price"].astype("int")
```

COGNITIVE CLASS.ai

Data Normalization in Python

COGNITIVE CLASS.ai

Data Normalization

- Uniform the features value with different range.

length	width	height
168.8	64.1	48.8
168.8	64.1	48.8
171.2	65.5	52.4
176.6	66.2	54.3
176.6	66.4	54.3
177.3	66.3	53.1
192.7	71.4	55.7
192.7	71.4	55.7
192.7	71.4	55.9

scale	[150,250]	[50,100]	[50,100]
impact	large	small	small

COGNITIVE CLASS.ai

Data Normalization

age	income
20	100000
30	20000
40	500000

Not-normalized

- "age" and "income" are in different range.
- hard to compare
- "income" will influence the result more

COGNITIVE CLASS.ai

Data Normalization

age	income
20	100000
30	20000
40	500000

age	income
0.2	0.2
0.3	0.04
0.4	1

Not-normalized

- "age" and "income" are in different range.
- hard to compare
- "income" will influence the result more

Normalized

- similar value range.
- similar intrinsic influence on analytical model.

COGNITIVE CLASS.ai

Methods of normalizing data

Several approaches for normalization:

① $x_{new} = \frac{x_{old}}{x_{max}}$

Simple Feature scaling

② $x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}}$

Min-Max

③ $x_{new} = \frac{x_{old} - \mu}{\sigma}$

Z-score

Simple Feature Scaling in Python

With Pandas:

length	width	height
168.8	64.1	48.8
168.8	64.1	48.8
180.0	65.5	52.4
...	...	...

→

length	width	height
0.81	64.1	48.8
0.81	64.1	48.8
0.87	65.5	52.4
...	...	...

```
df["length"] = df["length"]/df["length"].max()
```

Min-max in Python

With Pandas:

length	width	height
168.8	64.1	48.8
168.8	64.1	48.8
180.0	65.5	52.4
...	...	...

→

length	width	height
0.41	64.1	48.8
0.41	64.1	48.8
0.58	65.5	52.4
...	...	...

```
df["length"] = (df["length"]-df["length"].min())/
               (df["length"].max()-df["length"].min())
```

COGNITIVE CLASS.ai

Z-score in Python

With Pandas:

length	width	height
168.8	64.1	48.8
168.8	64.1	48.8
180.0	65.5	52.4
...	...	...

→

length	width	height
-0.034	64.1	48.8
-0.034	64.1	48.8
0.039	65.5	52.4
...	...	...

```
df["length"] = (df["length"]-df["length"].mean())/df["length"].std()
```

COGNITIVE CLASS.ai

Binning

Binning

- Binning: Grouping of values into "bins"
- Converts numeric into categorical variables
- Group a set of numerical values into a set of "bins"
- "price" is a feature range from 5,000 to 45,500
  (in order to have a **better representation** of price)

price: 5000, 10000,12000,12000, 30000, 31000, 39000, 44000,44500

bins: low Mid High

Binning in Python pandas

price
13495
16500
18920
41315
5151
6295
...

→

price	price-binned
13495	Low
16500	Low
18920	Medium
41315	High
5151	Low
6295	Low
...	...

COGNITIVE CLASS.ai

Binning in Python pandas

price
13495
16500
18920
41315
5151
6295
...

→

price	price-binned
13495	Low
16500	Low
18920	Medium
41315	High
5151	Low
6295	Low
...	...

```
bins = np.linspace(min(df["price"]), max(df["price"]), 4)
group_names = ["Low", "Medium", "High"]
```

COGNITIVE CLASS.ai

Binning in Python pandas

price
13495
16500
18920
41315
5151
6295
...

price	price-binned
13495	Low
16500	Low
18920	Medium
41315	High
5151	Low
6295	Low
...	...

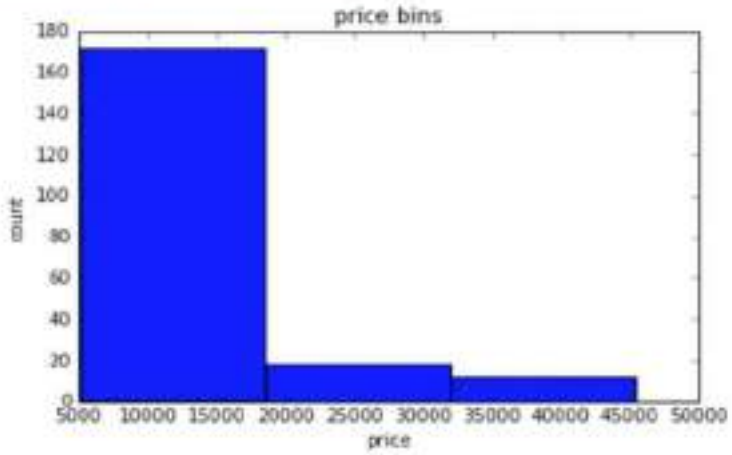
```
bins = np.linspace(min(df["price"]), max(df["price"]), 4)

group_names = ["Low", "Medium", "High"]

df["price-binned"] = pd.cut(df["price"], bins, labels=group_names, include_lowest=True )
```


Visualizing binned data

- E.g., Histograms



Turning categorical variables into quantitative variables in Python

COGNITIVE CLASS.ai

Categorical Variables

Problem:

- Most statistical models cannot take in the objects/strings as input

Car	Fuel	...
A	gas	...
B	diesel	...
C	gas	...
D	gas	...

COGNITIVE CLASS.ai

Categorical → Numeric

Solution:

- Add dummy variables for each unique category
- Assign 0 or 1 in each category

Car	Fuel	...	gas	diesel
A	gas	...	1	0
B	diesel	...	0	1
C	gas	...	1	0
D	gas	...	1	0

"One-hot encoding"

COGNITIVE CLASS.ai

Dummy variables in Python pandas

- Use pandas.get_dummies() method.
- Convert categorical variables to dummy variables (0 or 1)

fuel
gas
diesel
gas
gas

```
pd.get_dummies(df['fuel'])
```

COGNITIVE CLASS.ai

Dummy variables in Python pandas

- Use pandas.get_dummies() method.
- Convert categorical variables to dummy variables (0 or 1)

fuel
gas
diesel
gas
gas

→

gas	diesel
1	0
0	1
1	0
1	0

```
pd.get_dummies(df['fuel'])
```

COGNITIVE CLASS.ai

Exploratory Data Analysis

COGNITIVE CLASS.ai

Exploratory Data Analysis (EDA)

- Preliminary step in data analysis to:
  - Summarize main characteristics of the data
  - Gain better understanding of the data set
  - Uncover relationships between variables
  - Extract important variables

- Question:

  "What are the characteristics that have the most impact on the car price?"

Descriptive Statistics

Descriptive Statistics

- Describe basic features of data
- Giving short summaries about the sample and measures of the data

Descriptive Statistics- Describe()

- Summarize statistics using pandas **describe()** method

```
df.describe()
```

	Unnamed: 0	symboling	normalized-losses	wheel-base	length	width	height	curb-weight	engine-size	bore	stroke
count	201.000000	201.000000	164.000000	201.000000	201.000000	201.000000	201.000000	201.000000	201.000000	201.000000	201.000000
mean	100.000000	0.840796	122.000000	98.797015	174.200995	65.889055	53.766667	2555.666667	126.875622	3.319154	3.256766
std	58.167861	1.254802	35.442168	6.066366	12.322175	2.101471	2.447822	517.296727	41.546834	0.270130	0.316049
min	0.000000	-2.000000	65.000000	86.600000	141.100000	60.300000	47.800000	1488.000000	61.000000	2.540000	2.070000
25%	50.000000	0.000000	NaN	94.500000	166.800000	64.100000	52.000000	2169.000000	98.000000	3.150000	3.110000
50%	100.000000	1.000000	NaN	97.000000	173.200000	65.500000	54.100000	2414.000000	120.000000	3.310000	3.290000
75%	150.000000	2.000000	NaN	102.400000	183.500000	66.600000	55.500000	2926.000000	141.000000	3.580000	3.410000
max	200.000000	3.000000	256.000000	120.900000	208.100000	72.300000	59.800000	4066.000000	326.000000	3.940000	4.170000

Descriptive Statistics - Value_Counts()

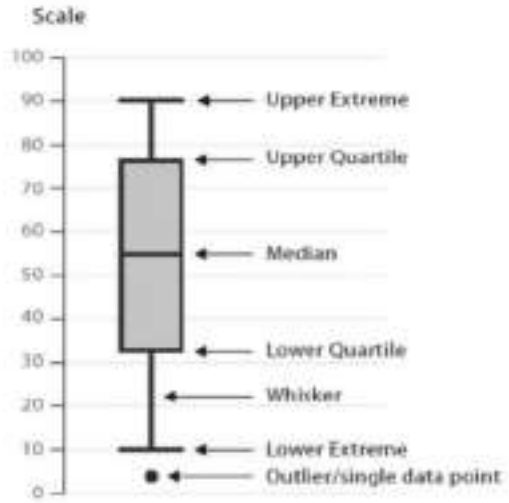
- summarize the categorical data is by using the **value_counts()** method

```
drive_wheels_counts=df["drive-wheels"].value_counts()
```

```
drive_wheels_counts.rename(colunms={'drive-wheels':'value_counts' inplace=True)
drive_wheels_counts.index.name= 'drive-wheels'
```

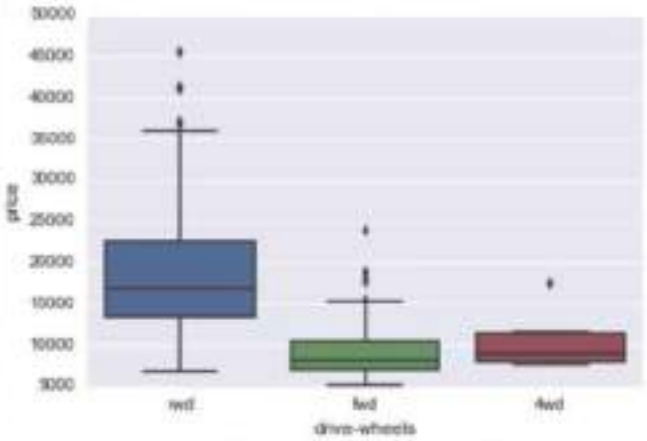
	vslue_counts
drive-wheels	
fwd	118
rwd	75
4wd	8

Descriptive Statistics - Box Plots



Box Plot - Example

```
sns.boxplot(x= "drive-wheels", y= "price",data=df)
```



Descriptive Statistics - Scatter Plot

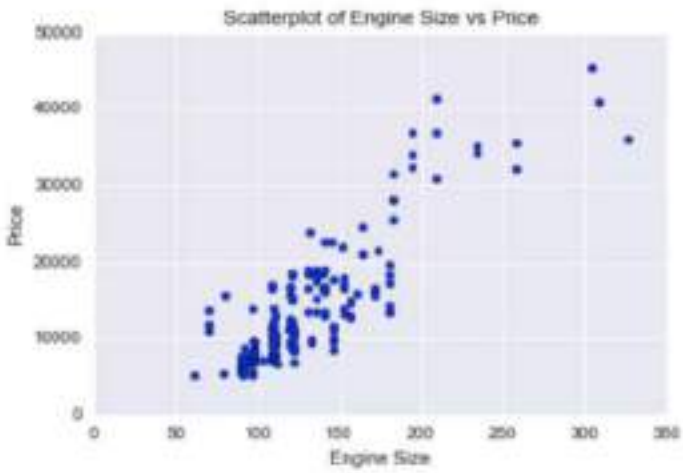
- Each observation represented as a point.
- Scatter plot Show the relationship between two variables.

1. Predictor/independent variables on x-axis.
2. Target/dependent variables on y-axis.

Scatterplot - Example

```
y=df["engine-size"]
x=df["price"]
plt.scatter(x,y)

plt.title("Scatterplot of Engine Size vs Price")
plt.xlabel("Engine Size")
plt.xlabel("Price")
```

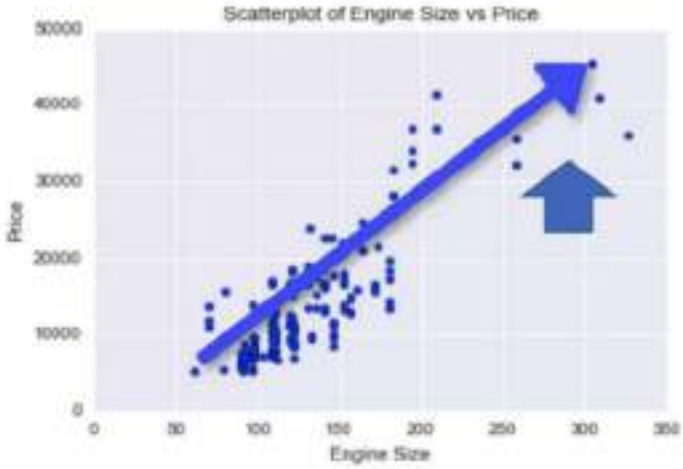


COGNITIVE CLASS.ai

Scatterplot - Example


```
y=df["engine-size"]
x=df["price"]
plt.scatter(x,y)

plt.title("Scatterplot of Engine Size vs Price")
plt.xlabel("Engine Size")
plt.xlabel("Price")
```

COGNITIVE CLASS.ai

GroupBy in Python

COGNITIVE CLASS.ai

Grouping data

- Use Panda **dataframe. Groupby()** method:
  - Can be applied on categorical variables
  - Group data into categories
  - Single or multiple variables

COGNITIVE CLASS.ai

Groupby()- Example

```
df_test = df['drive-wheels', 'body-style', 'price']
df_grp = df_test.groupby(['drive-wheels', 'body-style'], as_index=False).mean()
df_grp
```

	drive-wheels	body-style	price
0	4wd	hatchback	7603.000000
1	4wd	sedan	12647.333333
2	4wd	wagon	9095.750000
3	fwd	convertible	11595.000000
4	fwd	hardtop	8249.000000
5	fwd	hatchback	8396.387755
6	fwd	sedan	9811.800000
7	fwd	wagon	9997.333333
8	rwd	convertible	23949.600000
9	rwd	hardtop	24202.714286
10	rwd	hatchback	14337.777778
11	rwd	sedan	21711.833333
12	rwd	wagon	16994.222222

Pandas method - Pivot()

- One variable displayed along the columns and the other variable displayed along the rows.

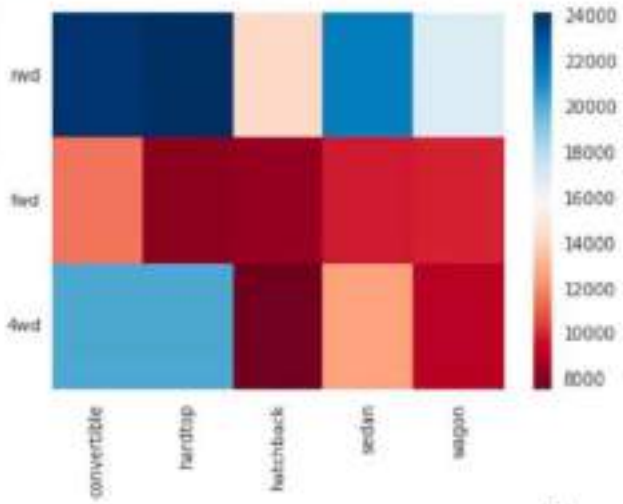
```
df_pivot = df_grp.pivot(index= 'drive-wheels', columns='body-style')
```

	price				
body-style	**convertible**	**hardtop**	**hatchback**	**sedan**	**wagon**
drive-wheels					
4wd	20239.229524	20239.229524	7603.000000	12647.333333	9095.750000
fwd	11595.000000	8249.000000	8396.387755	9811.800000	9997.333333
rwd	23949.600000	24202.714286	14337.777778	21711.833333	16994.222222

Heatmap

- Plot target variable over multiple variables

```
plt.pcolor(df_pivot, cmap='RdBBu')
plt.colorbar()
plt.show()
```



Correlation

Correlation

What is Correlation?

- Measures to what extent different variables are interdependent.

- For example:
  - Lung cancer → Smoking
  - Rain → Umbrella

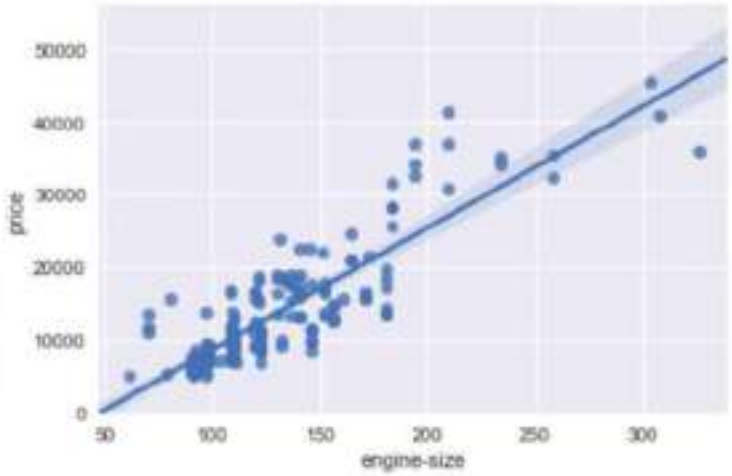
- Correlation doesn't imply causation.

COGNITIVE CLASS.ai

Correlation - Positive Linear Relationship

- Correlation between two features (engine-size and price).

```
sns.regplot(x= "engin-size",  y= "prices",data=df)
Plt.ylim(0,)
```

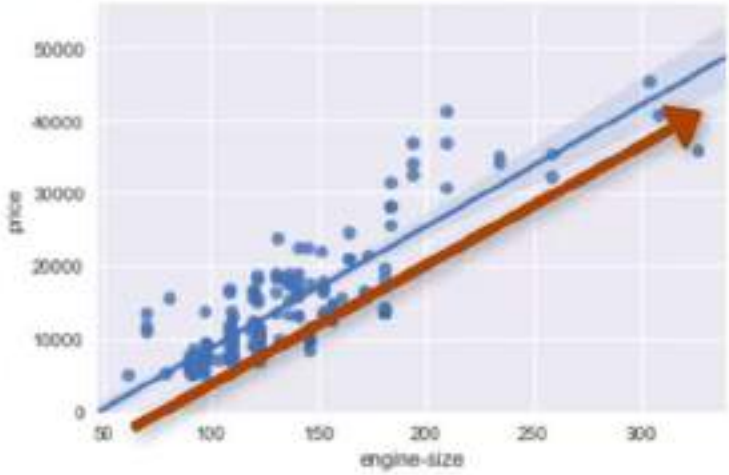


COGNITIVE CLASS.ai

Correlation - Positive Linear Relationship

- Correlation between two features (engine-size and price).

```
sns.regplot(x= "engin-size",  y= "prices",data=df)
Plt.ylim(0,)
```

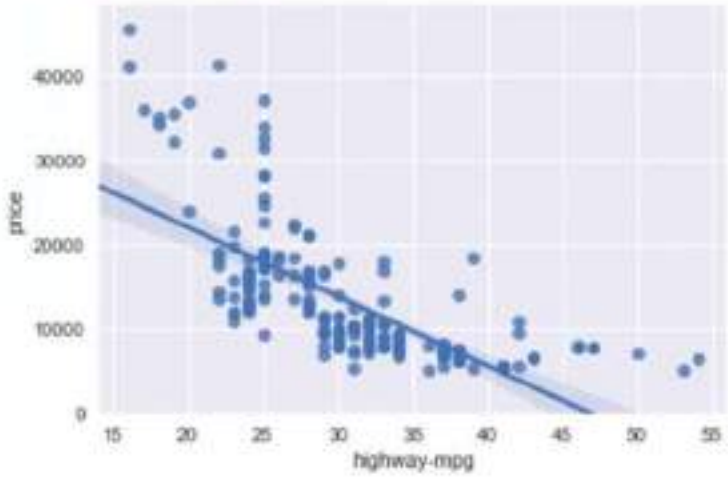


COGNITIVE CLASS.ai

Correlation - Negative Linear Relationship

- Correlation between two features (highway-mpg and price).

```
sns.regplot(x= "highway-mpg",  y= "prices",data=df)
plt.ylim(0,)
```

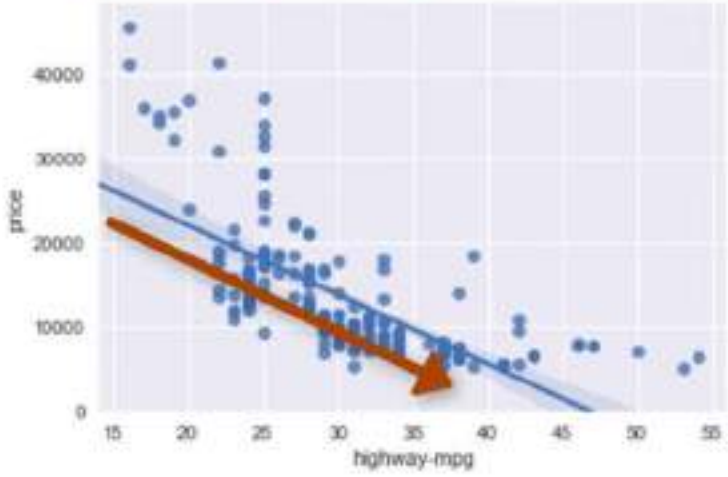


COGNITIVE CLASS.ai

Correlation - Negative Linear Relationship

- Correlation between two features (highway-mpg and price).

```
sns.regplot(x= "highway-mpg", y= "prices",data=df)
plt.ylim(0,)
```

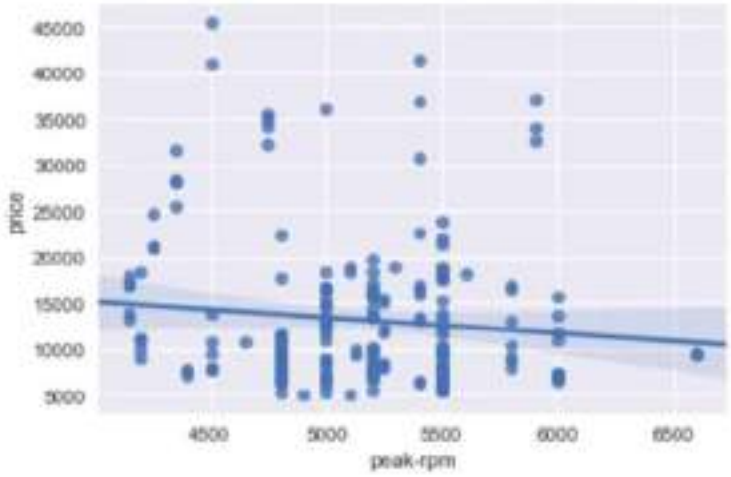


COGNITIVE CLASS.ai

Correlation - Negative Linear Relationship

- Weak correlation between two features (peak-rmp and price).

```
sns.regplot(x= "peak-rmp", y= "prices",data=df)
plt.ylim(0,)
```



COGNITIVE CLASS.ai

Next week…

Model Development

COGNITIVE CLASS.ai

Model Development

Model Development

- A model can be thought of as a mathematical equation used to predict a value given one or more other values
- Relating one or more independent variables to dependent variables.

independent variables or features

'highway-mpg'

$55\ mpg$

Model

dependent variables

'predicted price'

$5000

Model Development

- Usually the more **relevant data** you have the more accurate your model is

'highway-mpg'
'curb-weight'
'engine-size'
'city-mpg'

→ **Model** → 'price' $5400

Model Development

To understand why more data is important consider the following situation:

1. you have two almost identical cars
2. Pink cars sell for significantly less

'highway-mpg'
'curb-weight'
'engine-size'
'city-mpg'

→ **Model** → $Y = \$5400$

'highway-mpg'
'curb-weight'
'engine-size'
'city-mpg'

→ **Model** → $Y = \$5400$

Model Development

In addition to getting more data you can try different types of models.

In this course you will learn about:

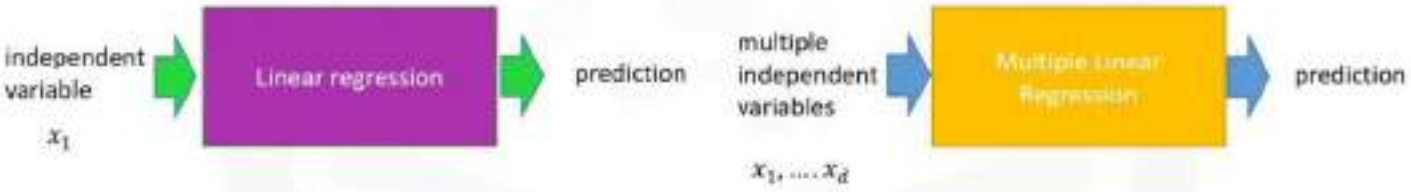
1. Simple Linear Regression
2. Multiple Linear Regression
3. Polynomial Regression

Linear Regression and Multiple Linear Regression

© 2017 IBM Corporation

Introduction

- Linear regression will refer to one independent variable to make a prediction
- Multiple Linear Regression will refer to multiple independent variables to make a prediction



COGNITIVE CLASS.ai

© 2017 IBM Corporation

Simple Linear Regression

1. The predictor (independent) variable - x
2. The target (dependent) variable - y

$$y = b_0 + b_1 x$$

- b_0: **the intercept**
- b_1 : **the slope**

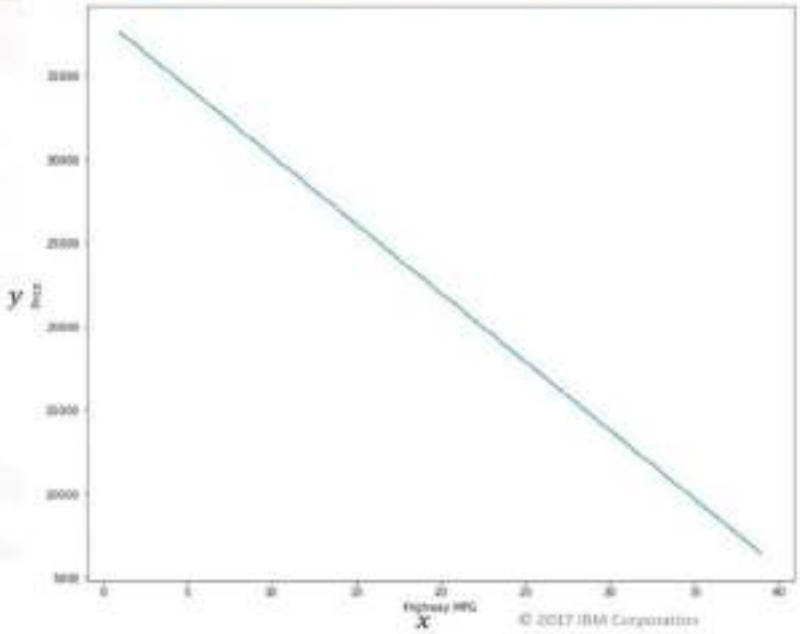
When we fit or train the model, we will come up with these parameters.

That requires lots of math.

COGNITIVE CLASS.ai

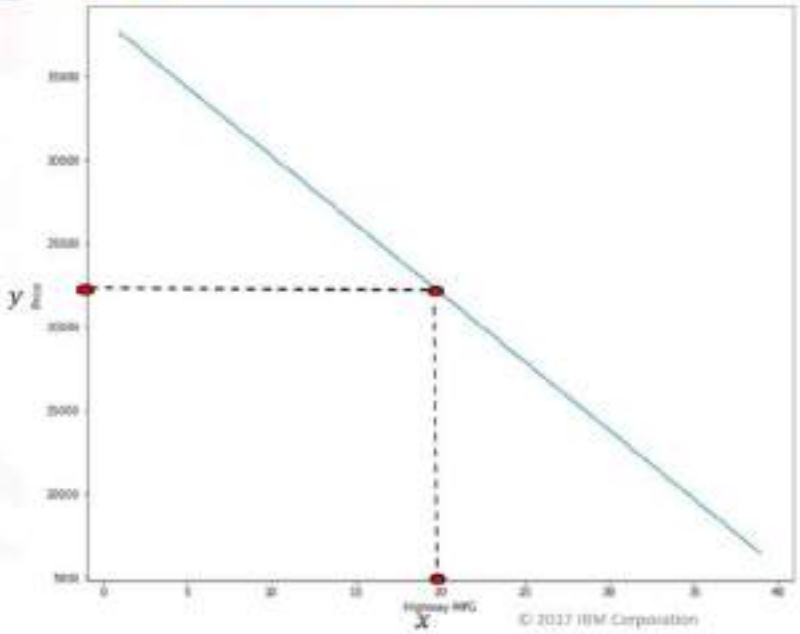
Simple Linear Regression: Prediction

$$y = 38423 - 821x$$

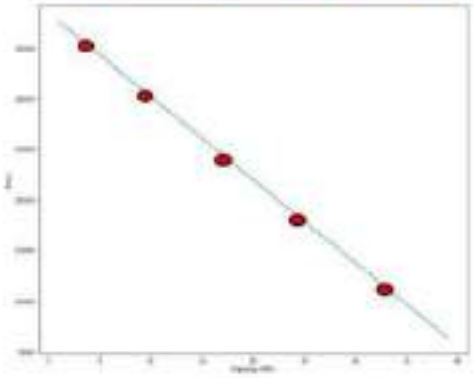


Simple Linear Regression: Prediction

$$y = 38423 - 821x$$
$$= 38423 - 821\,(20)$$
$$= 22\,003$$



Simple Linear Regression: Fit



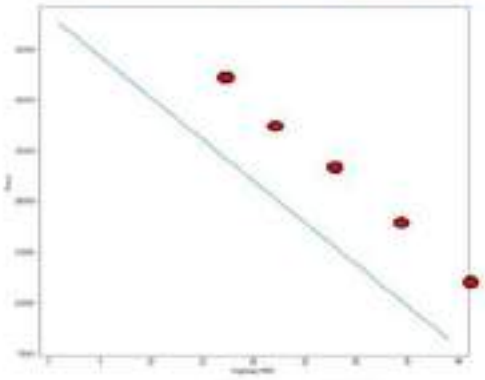
Fit

COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

Simple Linear Regression: Fit



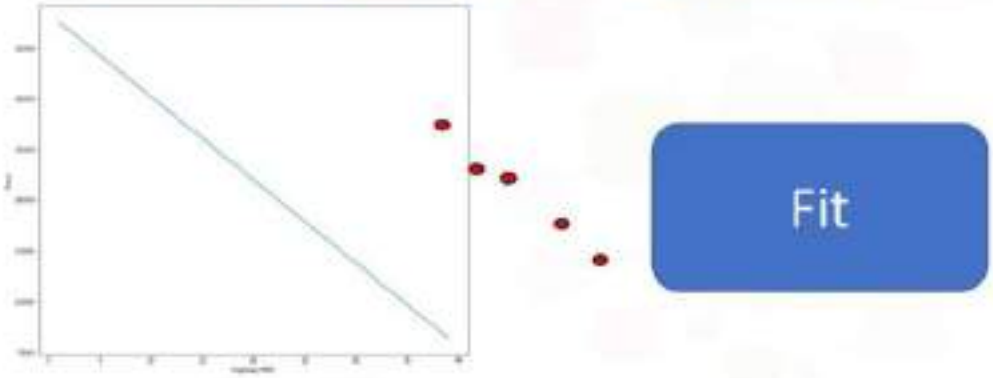
Fit

COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

Simple Linear Regression: Fit

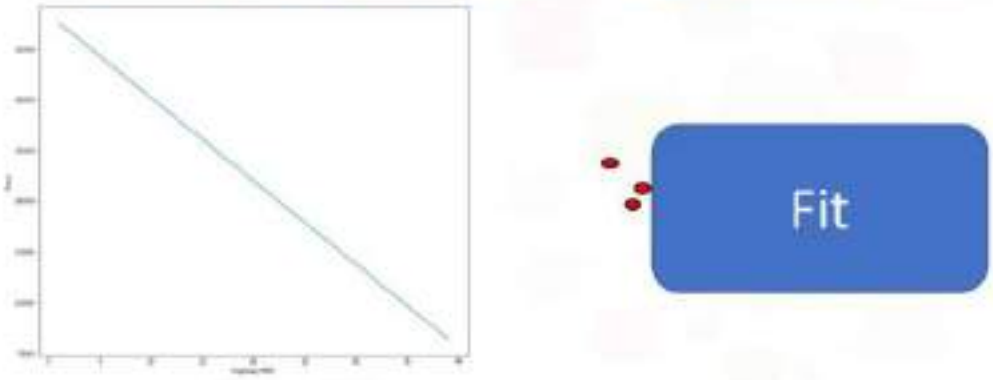


COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

Simple Linear Regression: Fit



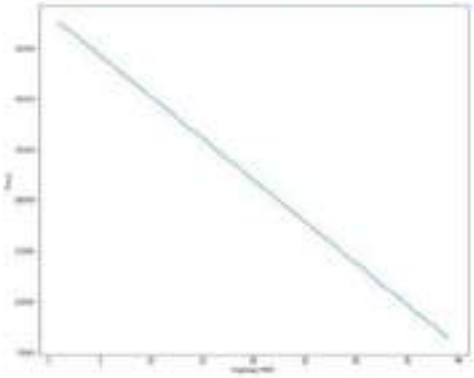
COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

5

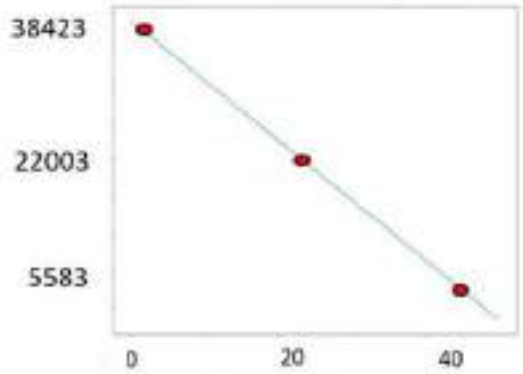
Simple Linear Regression: Fit



Fit

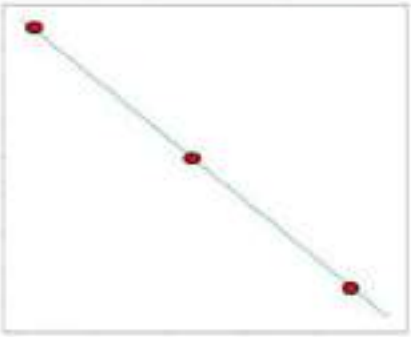
(b_0, b_1)

Simple Linear Regression: Fit



$$X = \begin{bmatrix} \\ \\ \end{bmatrix} \quad Y = \begin{bmatrix} \\ \\ \end{bmatrix}$$

Simple Linear Regression: Fit



$$X = \begin{bmatrix} 0 \\ 20 \\ 40 \end{bmatrix} \quad Y = \begin{bmatrix} 38423 \\ 22003 \\ 5583 \end{bmatrix}$$

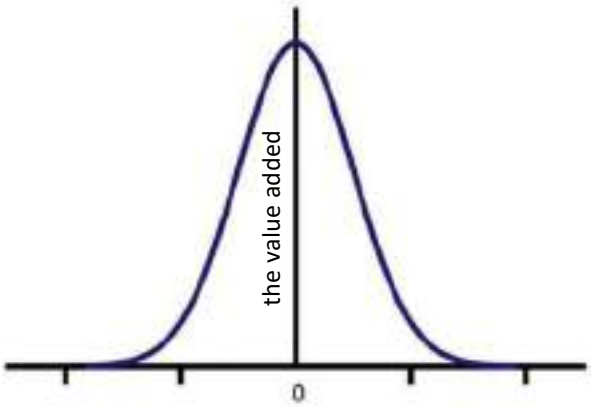
COGNITIVE CLASS.ai

x

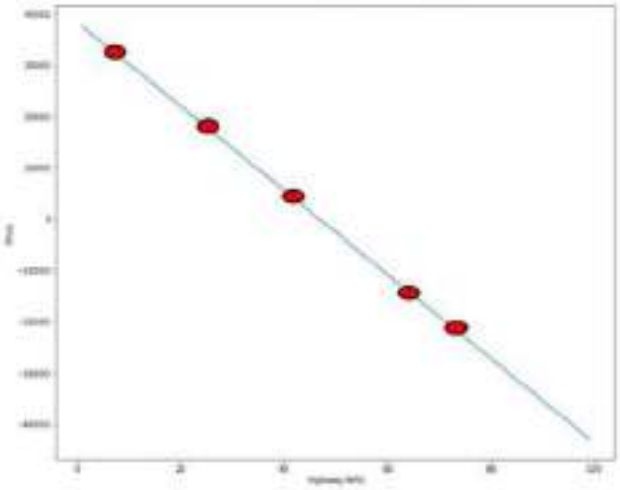
© 2017 IBM Corporation

Simple Linear Regression: Fit

distribution of the noise



the probability that the value will be added

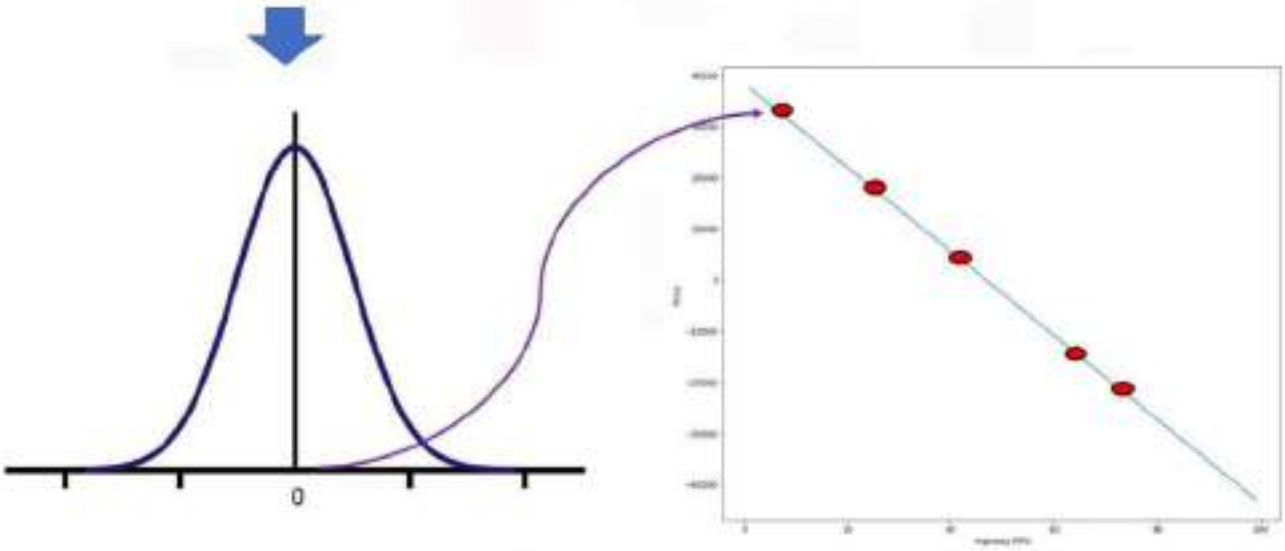


COGNITIVE CLASS.ai

x

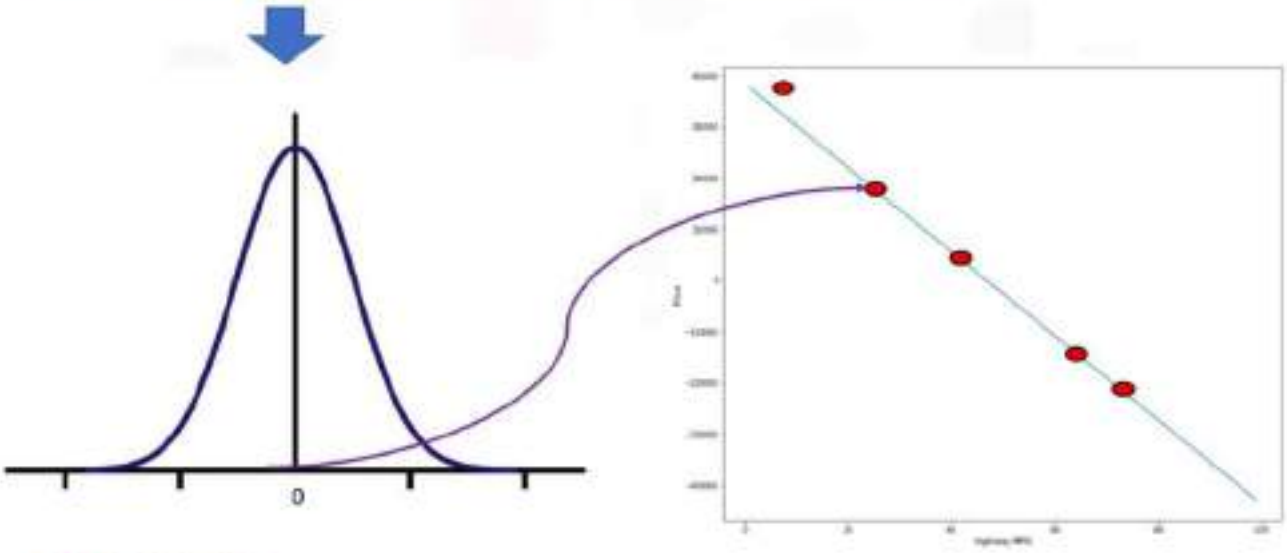
© 2017 IBM Corporation

Simple Linear Regression: Fit



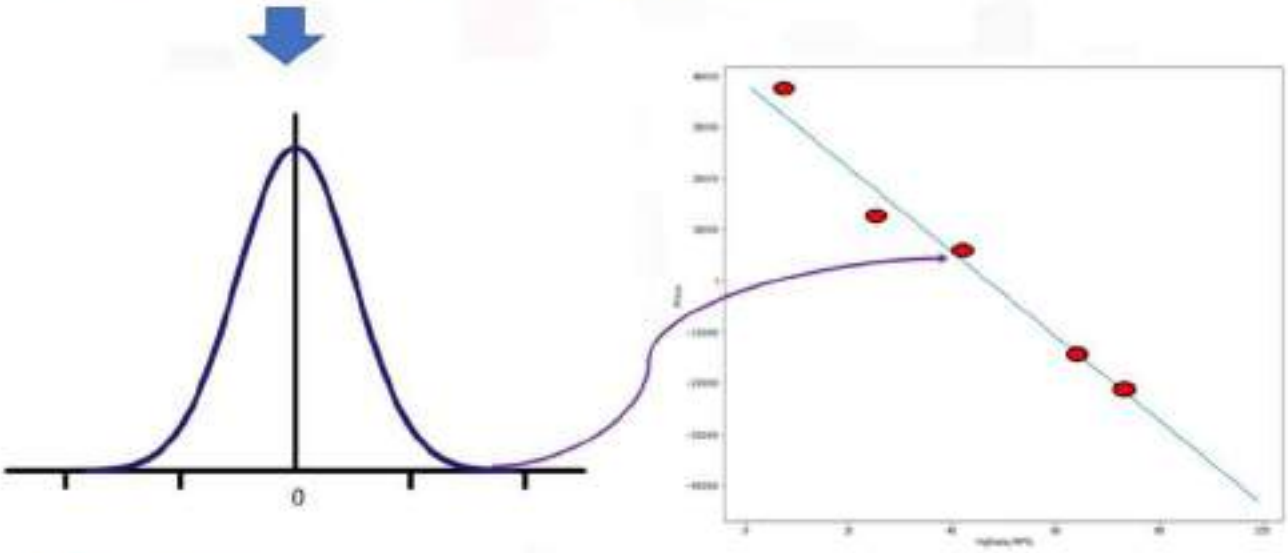
COGNITIVE CLASS.ai

Simple Linear Regression: Fit



COGNITIVE CLASS.ai

Simple Linear Regression: Fit

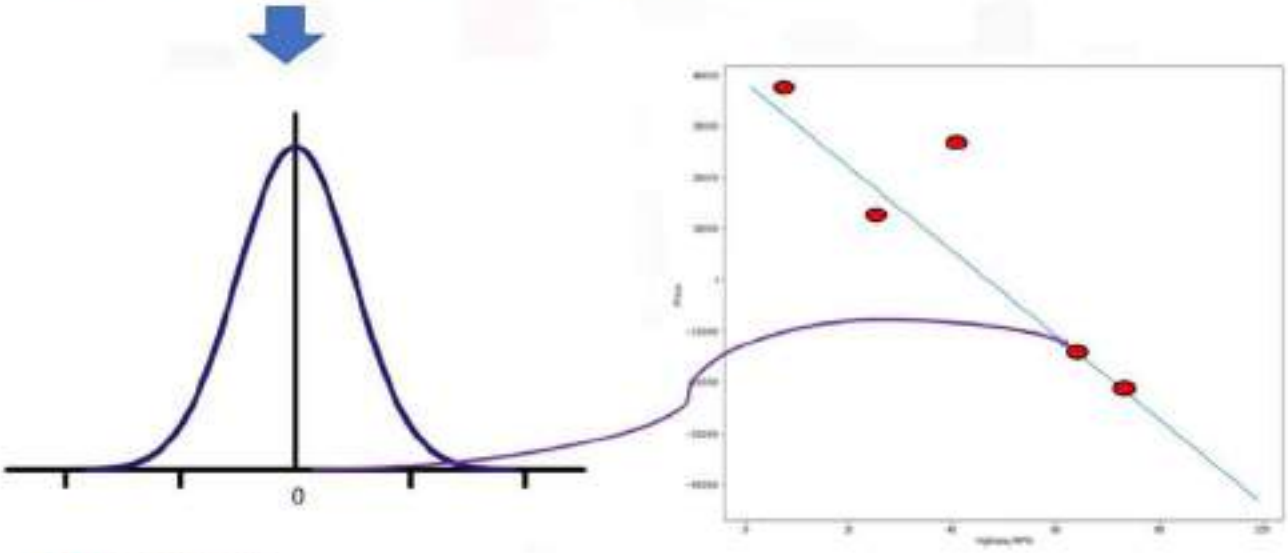


COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

Simple Linear Regression: Fit



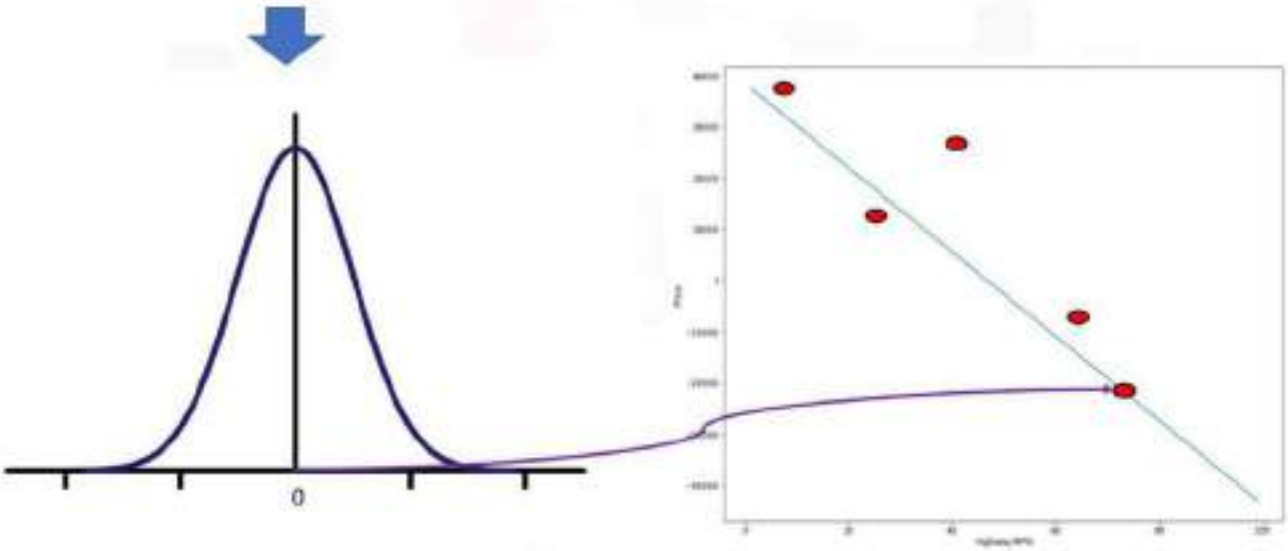
COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

7

Simple Linear Regression: Fit

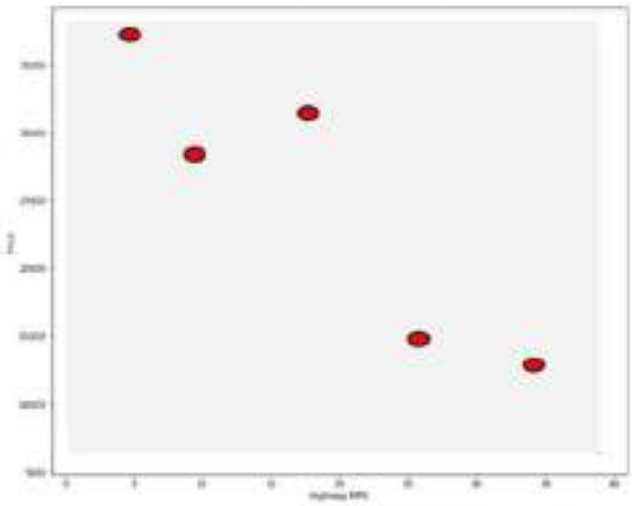


COGNITIVE CLASS.ai

x

© 2017 IBM Corporation

Simple Linear Regression



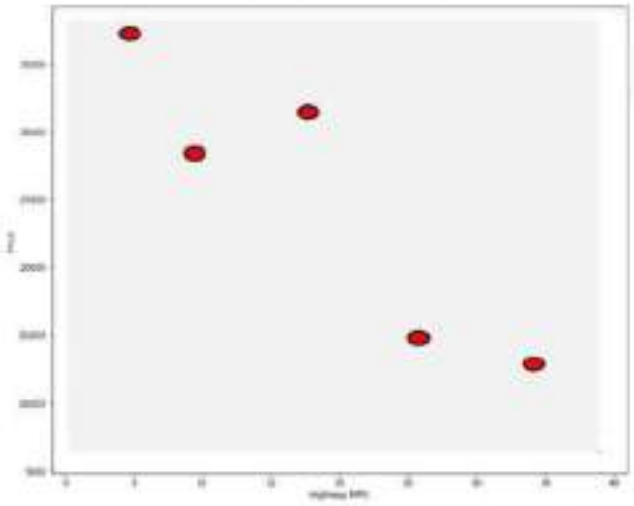
Fit

Predict

COGNITIVE CLASS.ai

© 2017 IBM Corporation

Simple Linear Regression

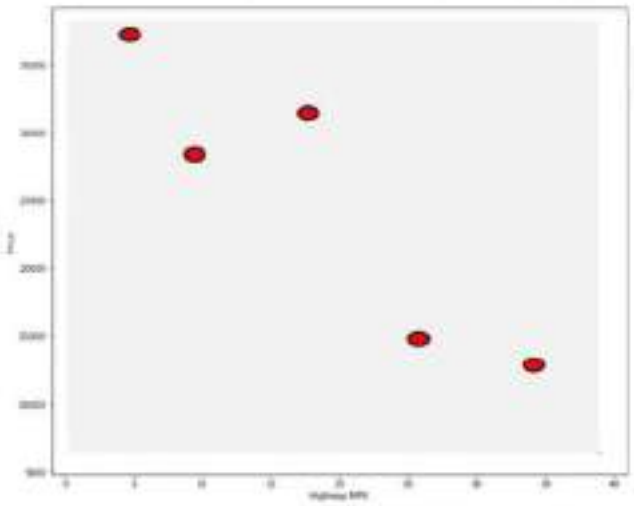


Fit

Predict

© 2017 IBM Corporation

Simple Linear Regression



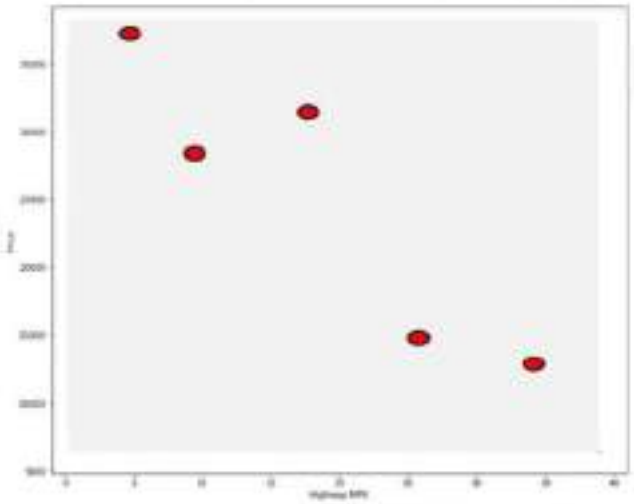
Fit

(b_0, b_1)

Predict

© 2017 IBM Corporation

Simple Linear Regression



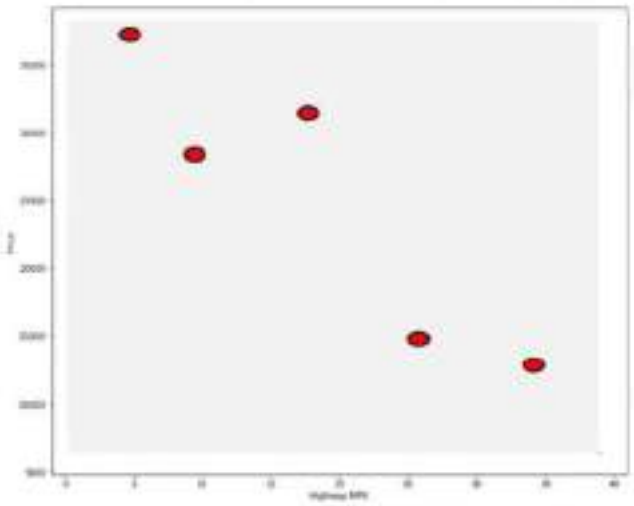
Fit

Predict

$$\hat{y} = b_0 + b_1\, x$$

© 2017 IBM Corporation

Simple Linear Regression



Fit

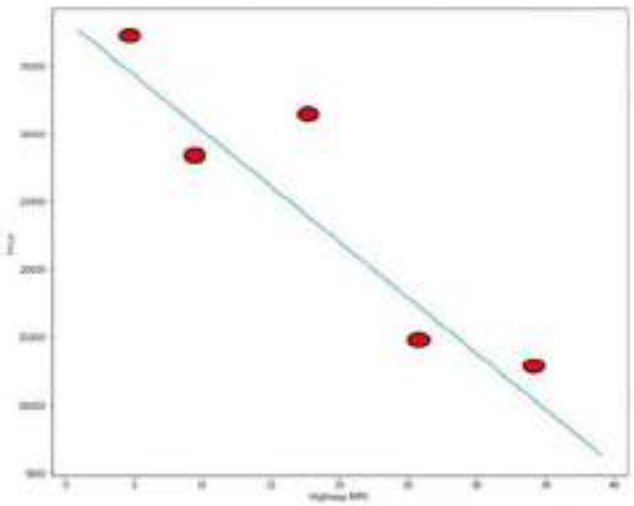
Predict

$$\hat{y} = b_0 + b_1\, x$$

© 2017 IBM Corporation

8

Simple Linear Regression



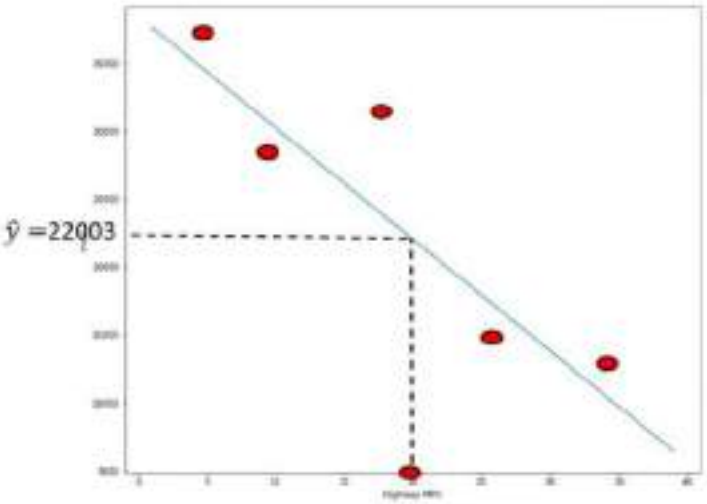
Fit

Predict

$$\hat{y} = b_0 + b_1\, x$$

© 2017 IBM Corporation

Simple Linear Regression



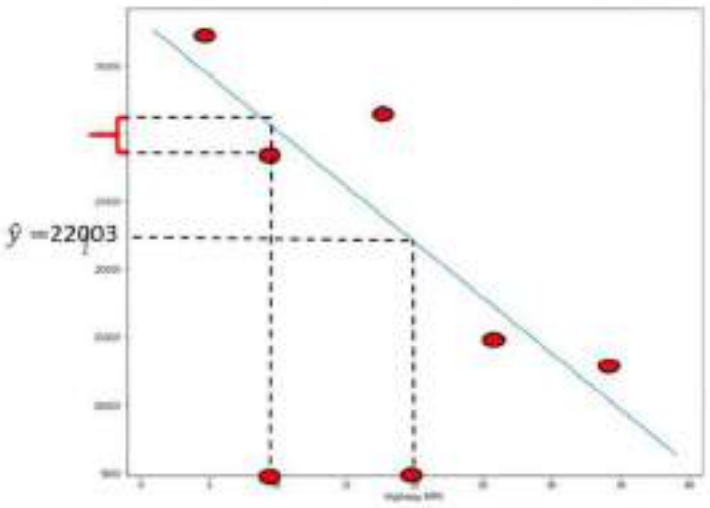
Fit

Predict

$$\hat{y} = b_0 + b_1\, x$$

© 2017 IBM Corporation

Simple Linear Regression



Fit

Predict

$$\hat{y} = b_0 + b_1\, x$$

© 2017 IBM Corporation

Fitting a Simple Linear Model Estimator

- X :Predictor variable
- Y: Target variable

1. Import linear_model from scikit-learn

```
from sklearn.linear_model import LinearRegression
```

2. Create a Linear Regression Object using the constructor :

```
lm=LinearRegression()
```

Fitting a Simple Linear Model

- We define the predictor variable and target variable

```
X = df[['highway-mpg']]
Y = df['price']
```

- Then use lm.fit (X, Y) to fit the model , i.e fine the parameters b_0 and b_1

```
lm.fit(X, Y)
```

- We can obtain a prediction

```
Yhat=lm.predict(X)
```

Yhat	X
2	5
:	
3	4

COGNITIVE CLASS.ai

SLR – Estimated Linear Model

- We can view the intercept (b_0): `lm.intercept_`
  38423.305858
- We can also view the slope (b_1): `lm.coef_`
  -821.73337832
- The Relationship between Price and Highway MPG is given by:
- **Price = 38423.31 - 821.73 * highway-mpg**

COGNITIVE CLASS.ai

SLR – Estimated Linear Model

- We can view the intercept (b_0): `lm.intercept_`
  `38423.305858`
- We can also view the slope (b_1): `lm.coef_`
  -821.73337832
- The Relationship between Price and Highway MPG is given by:
- **Price = 38423.31 - 821.73 * highway-mpg**

$$\hat{Y} = b_0 + b_1 x$$

Multiple Linear Regression (MLR)

This method is used to explain the relationship between:

- One continuous target (Y) variable
- Two or more predictor (X) variables

Multiple Linear Regression (MLR)

$$\hat{Y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4$$

- b_0 : **intercept (X=0)**
- b_1: the **coefficient** or **parameter** of x_1
- b_2: the **coefficient** of **parameter** x_2 and so on..

Fitting a Multiple Linear Model Estimator

1. We can extract the for 4 predictor variables and store them in the variable Z

```
Z = df[['horsepower', 'curb-weight', 'engine-size', 'highway-mpg']]
```

2. Then train the model as before:

```
lm.fit(Z, df['price'])
```

3. We can also obtain a prediction

```
Yhat=lm.predict(X)
```

Fitting a Multiple Linear Model Estimator

1. We can extract the for 4 predictor variables and store them in the variable Z

```
Z = df[['horsepower', 'curb-weight', 'engine-size', 'highway-mpg']]
```

2. Then train the model as before:

```
lm.fit(Z, df['price'])
```

3. We can also obtain a prediction

```
Yhat=lm.predict(X)
```

x_1	x_2	x_3	x_4
3	5	-4	3
:	:	:	:
2	4	2	-4

COGNITIVE CLASS

Fitting a Multiple Linear Model Estimator

1. We can extract the for 4 predictor variables and store them in the variable Z

```
Z = df[['horsepower', 'curb-weight', 'engine-size', 'highway-mpg']]
```

2. Then train the model as before:

```
lm.fit(Z, df['price'])
```

3. We can also obtain a prediction

```
Yhat=lm.predict(X)
```

x_1	x_2	x_3	x_4
3	5	-4	3
:	:	:	:
2	4	2	-4

Yhat
2
:
3

COGNITIVE CLASS

17

MLR – Estimated Linear Model

1. Find the intercept (b_0)

```
lm.intercept_
-15678.742628061467
```

2. Find the coefficients (b_1, b_2, b_3, b_4)

```
lm.coef_
array([52.65851272 ,4.69878948,81.95906216 , 33.58258185])
```

The Estimated Linear Model:

- **Price** = -15678.74 + (52.66) * **horsepower** + (4.70) * **curb-weight** + (81.96) * **engine-size** + (33.58) * **highway-mpg**

$$\hat{Y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4$$

COGNITIVE CLASS

Polynomial Regression and Pipelines

What do we do when a linear model is not the best fit for our data?

COGNITIVE CLASS

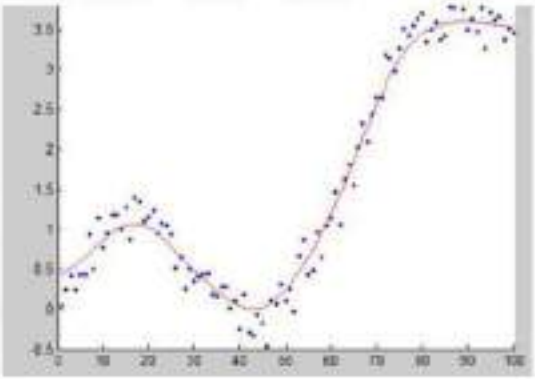
© 2017 IBM Corporation

Polynomial Regressions

- A special case of the general linear regression model
- Useful for describing curvilinear relationships

Curvilinear relationships:

By squaring or setting higher-order terms of the predictor variables



COGNITIVE CLASS.ai

Polynomial Regression

- Quadratic – 2nd order

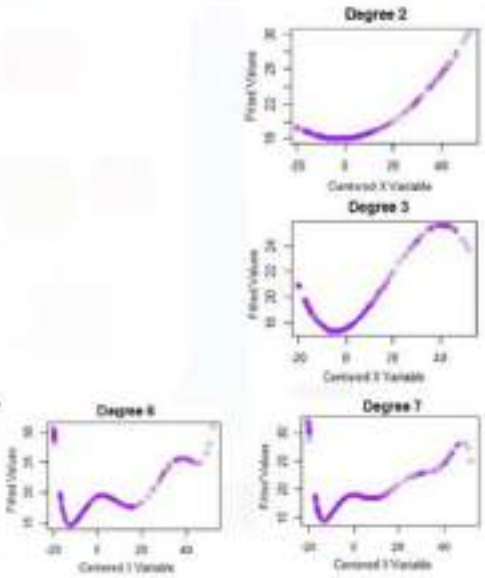
$$\hat{Y} = b_0 + b_1\, x_1 + b_2 (\, x_1)^2$$

- Cubic – 3rd order

$$\hat{Y} = b_0 + b_1\, x_1 + b_2 (\, x_1)^2 + b_3 (\, x_1)^3$$

- Higher order

$$\hat{Y} = b_0 + b_1\, x_1 + b_2 (\, x_1)^2 + b_3 (\, x_1)^3 + ..$$



COGNITIVE CLASS.ai

Polynomial Regression

1. Calculate Polynomial of 3rd order

```
f=np.polyfit(x,y,3)
p=np.polydl(f)
```

2. We can print out the model

```
print (p)
```

$$-1.557(x_1)^3 + 204.8(x_1)^2 + 8965\,x_1 + 1.37\text{x}10^5$$

Pipelines

© 2017 IBM Corporation

Pipelines

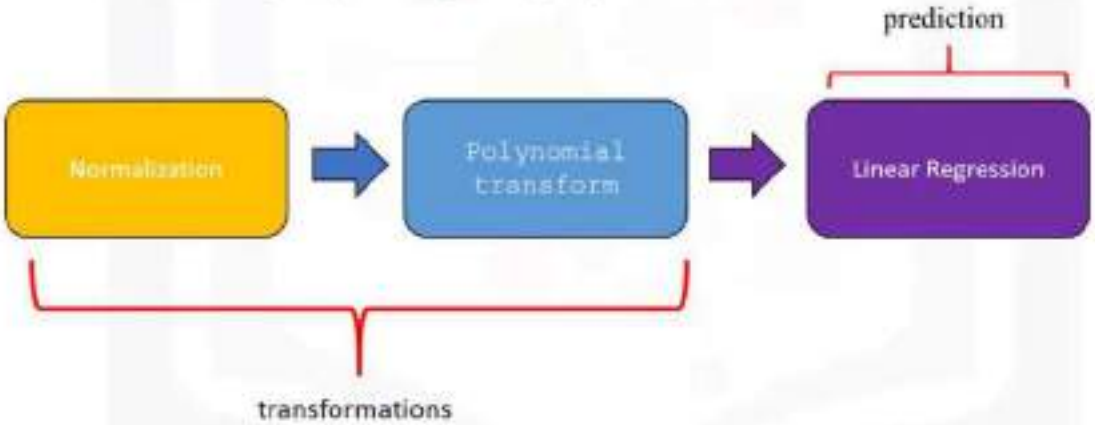
- There are many steps to getting a prediction

COGNITIVE CLASS.ai

© 2017 IBM Corporation

Pipelines

- There are many steps to getting a prediction



COGNITIVE CLASS.ai

© 2017 IBM Corporation

Pipelines

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
```


© 2017 IBM Corporation

Pipeline Constructor

```
Input=[('scale',StandardScaler()),('polynomial',PolynomialFeatures(degree=2),…
('mode',LinearRegression())]
```


© 2017 IBM Corporation

Pipeline Constructor

```
Input=[('scale',StandardScaler()),('polynomial',PolynomialFeatures(degree=2),…
('mode',LinearRegression())]
```

- Pipeline constructor

```
pipe=Pipeline(Input)
```

pipeline object

© 2017 IBM Corporation

Pipeline Constructor

- We can train the pipeline object

```
Pipe.train(X['horsepower', 'curb-weight', 'engine-size', 'highway-mpg'],y)

yhat=Pipe.predict(X[['horsepower', 'curb-weight', 'engine-size', 'highway-mpg']])
```

© 2017 IBM Corporation

Measures for In-Sample Evaluation

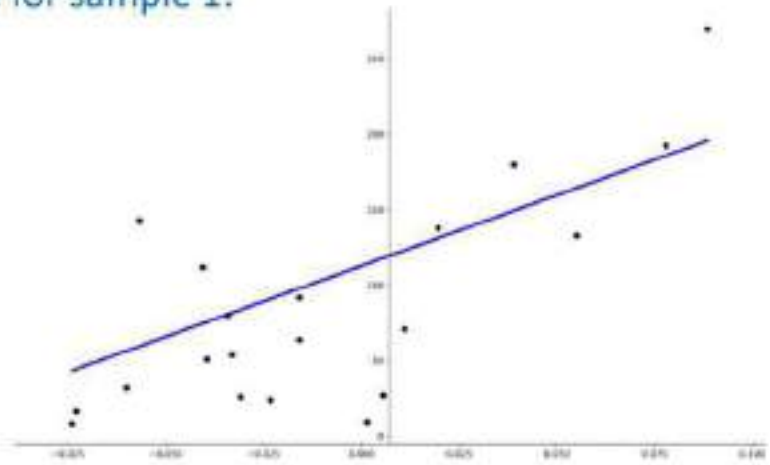
© 2017 IBM Corporation

Measures for In-Sample Evaluation

- A way to numerically determine how good the model fits on dataset.
- Two important measures to determine the fit of a model:
  - **Mean Squared Error (MSE)**
  - **R-squared (R^2)**

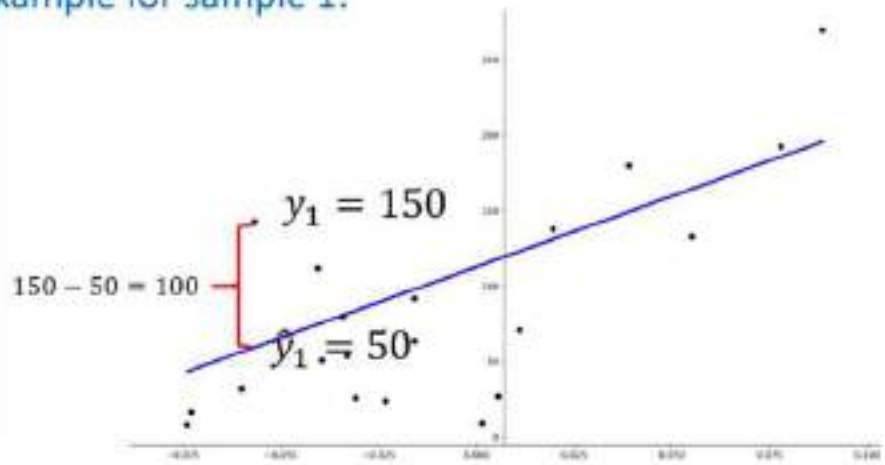
Mean Squared Error (MSE)

- For Example for sample 1:



Mean Squared Error (MSE)

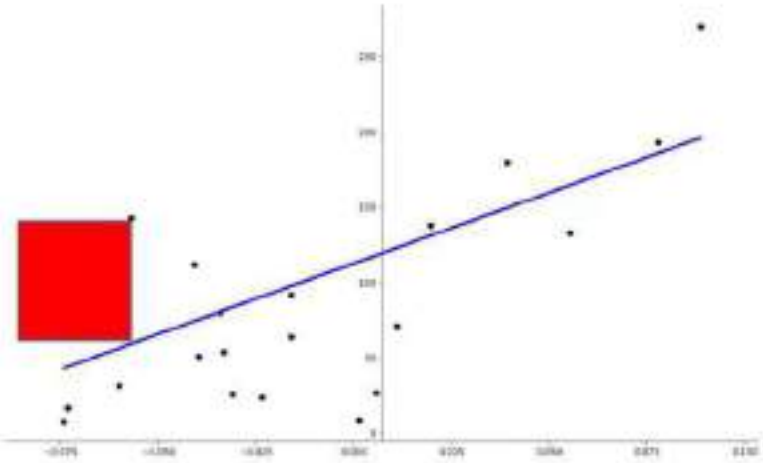
- For Example for sample 1:



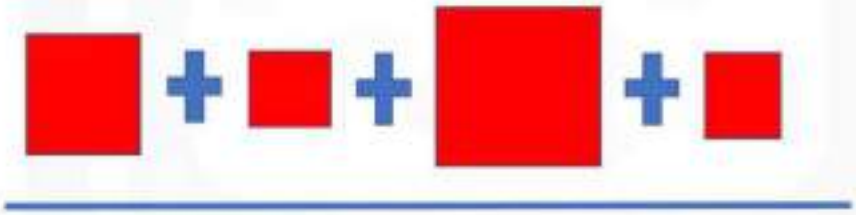
Mean Squared Error (MSE)

- To make all the values positive we square it

$(\ 50)^2$



Mean Squared Error (MSE)



Number of Samples

Mean Squared Error (MSE)

- In python we can measure the MSE as follows

```
from  sklearn.metrics import mean_squared_error

mean_squared_error(df['price'],Y_predict_simple_fit)

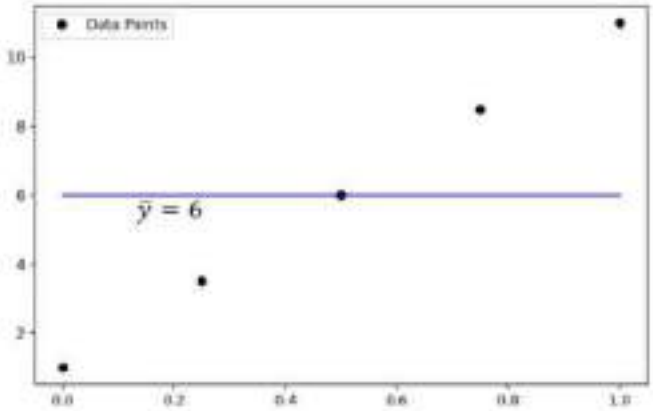

3163502.944639988
```


R-squared/ R^2

- The Coefficient of Determination or R squared (R^2)
- Is a measure to determine how close the data is to the fitted regression line.
- R^2: the percentage of variation of the target variable (Y) that is explained by the linear model.
- Think about as comparing a regression model to a simple model i.e the mean of the data points

Coefficient of Determination (R^2)

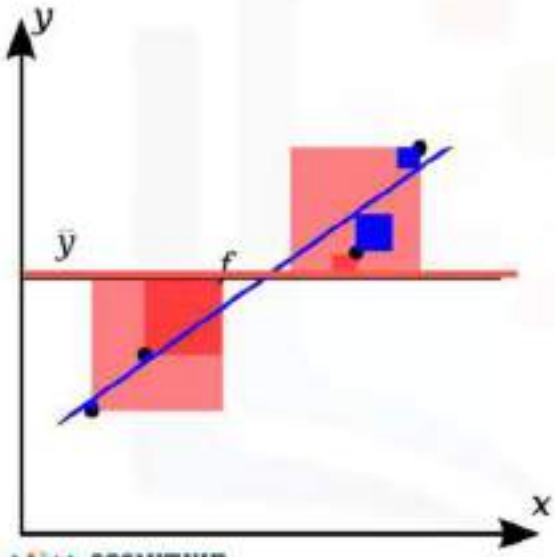
- In this example the average of the data points $\bar{y}$ is 6



Coefficient of Determination (R^2)

$$R^2 = \left(1 - \frac{\text{MSE of regression line}}{\text{MSE of the average of the data}}\right)$$

Coefficient of Determination (R^2)



- The blue line represents the regression line
- The blue squares represents the MSE of the regression line
- The red line represents the average value of the data points
- The red squares represent the MSE of the red line
- We see the area of the blue squares is much smaller then the area of the red squares

Source: https://en.wikipedia.org/wiki/Coefficient_of_determination

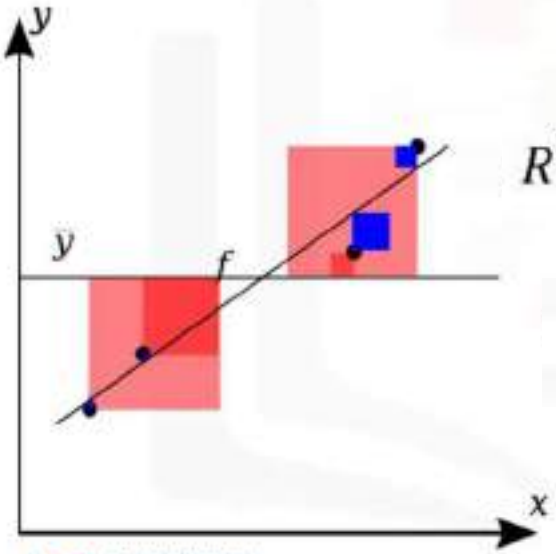
Coefficient of Determination (R^2)

- In this case ratio of the areas of MSE is close to zero

$$\frac{\text{MSE of regression line}}{\text{MSE of } \bar{y}} = \frac{\blacksquare + \blacksquare}{\blacksquare + \blacksquare}$$

$$= 0$$

Coefficient of Determination (R^2)



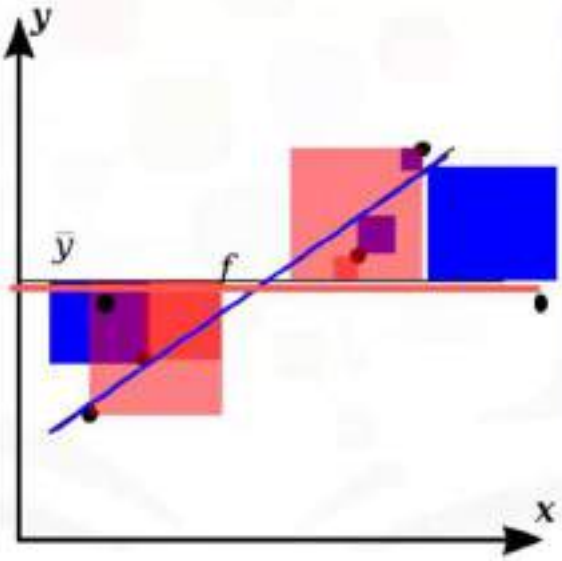
$$R^2 = \left(1 - \frac{\text{MSE of regression line}}{\text{MSE of } \bar{y}}\right)$$

$$= (1 - 0)$$

$$= 1$$

COGNITIVE CLASS.AI

Coefficient of Determination (R^2)



COGNITIVE CLASS.AI

Source: https://en.wikipedia.org/wiki/Coefficient_of_determination

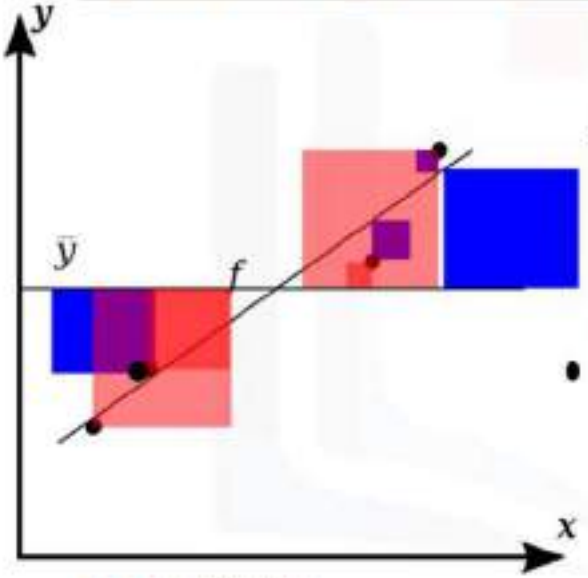
Coefficient of Determination (R^2)

$$\frac{\text{MSE of regression line}}{\text{MSE of } \bar{y}} = \frac{\blacksquare + \blacksquare}{\blacksquare + \blacksquare}$$

$$= 1$$

COGNITIVE CLASS.ai

Coefficient of Determination (R^2)



$$R^2 = \left(1 - \frac{\text{MSE of regression line}}{\text{MSE of } \bar{y}}\right)$$

$$= (1 - 1)$$

$$= 0$$

COGNITIVE CLASS.ai

R-squared/ R^2

- Generally the values of the MSE are between 0 and 1.
- WE can calculate the the R^2 as follows

```
X = df[['highway-mpg']]
Y = df['price']

lm.fit(X, Y)
```

```
lm.score(X,y)
0.496591188
```


Model Evaluation

© 2017 IBM Corporation

Model Evaluation

- In-sample evaluation tells us how well our model will fit the data used to train it
- Problem?
  - It does not tell us how well the trained model can be used to predict new data
- Solution?
  - In- sample data or training data
  - Out-of-sample evaluation or test set

Training/Testing Sets

Data:

Training/Testing Sets

Data:

- Split dataset into:
  - Training set (70%),
  - Testing set (30%)
- Build and train the model with a training set
- Use testing set to assess the performance of a predictive model

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']

COGNITIVE CLASS.ai

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']
- **x_train, y_train**: parts of available data as training set

COGNITIVE CLASS.ai

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']
- **x_train, y_train**: parts of available data as training set
- **x_test, y_test**: parts of available data as testing set

COGNITIVE CLASS.ai

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']
- **x_train, y_train**: parts of available data as training set
- **x_test, y_test**: parts of available data as testing set
- **test_size**: percentage of the data for testing (here 30%)

COGNITIVE CLASS.ai

Function train_test_split()

- Split data into random train and test subsets

```
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x_data, y_data, test_size=0.3, random_state=0)
```

- **x_data**: features or independent variables
- **y_data**: dataset target: df['price']
- **x_train, y_train**: parts of available data as training set
- **x_test, y_test**: parts of available data as testing set
- **test_size**: percentage of the data for testing (here 30%)
- **random_state**: number generator used for random sampling

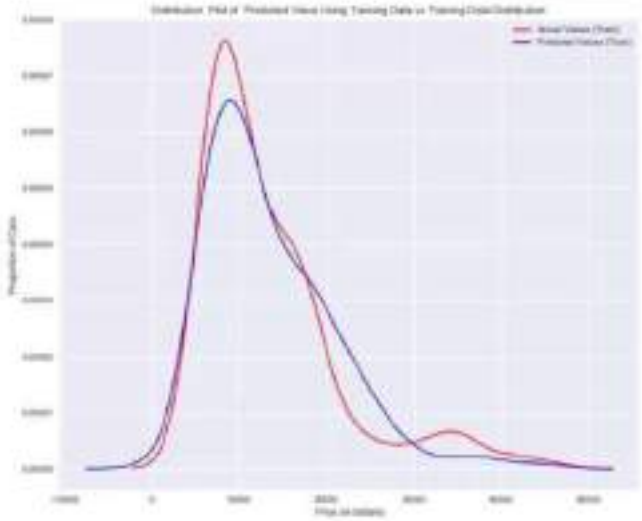
HW: Check out Stratified Train-Test Splits

Generalization Performance

- Generalization error is measure of how well our data does at predicting previously unseen data
- The error we obtain using our testing data is an approximation of this error

© 2017 IBM Corporation

Generalization Error



This figure shows the distribution of

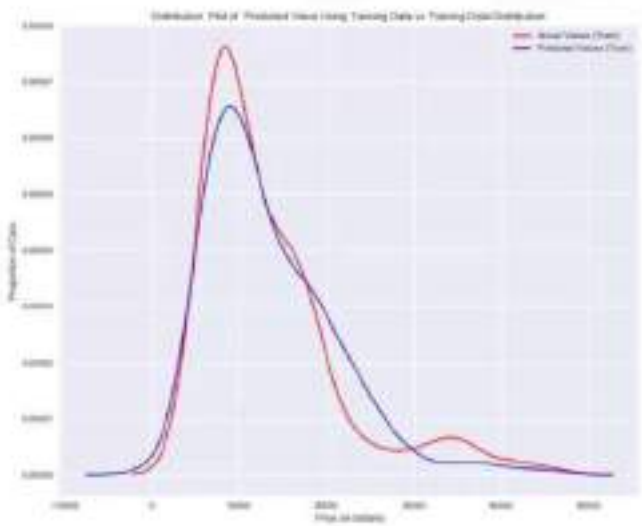
- the actual values in red compared to
- the predicted values from a linear regression in blue.

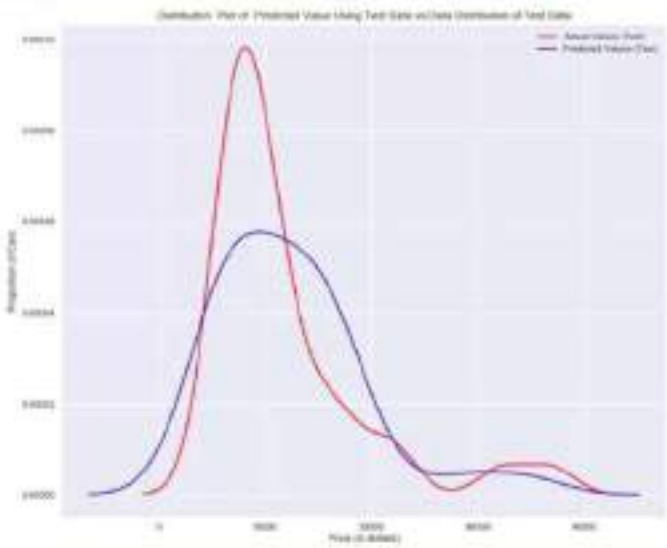
We see the distributions are somewhat similar.

Note that, we are using only the training data here.

© 2017 IBM Corporation

Generalization Error

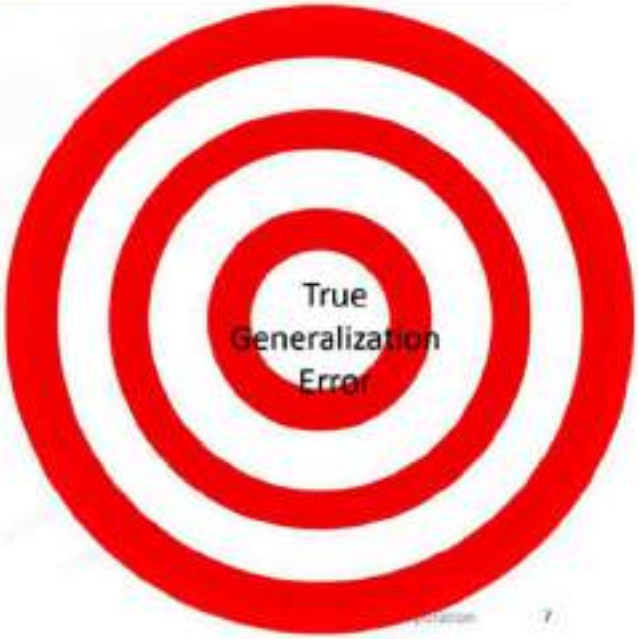




© 2017 IBM Corporation

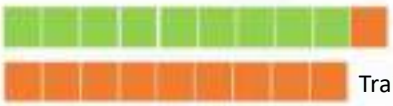
Lots of Training Data

Model

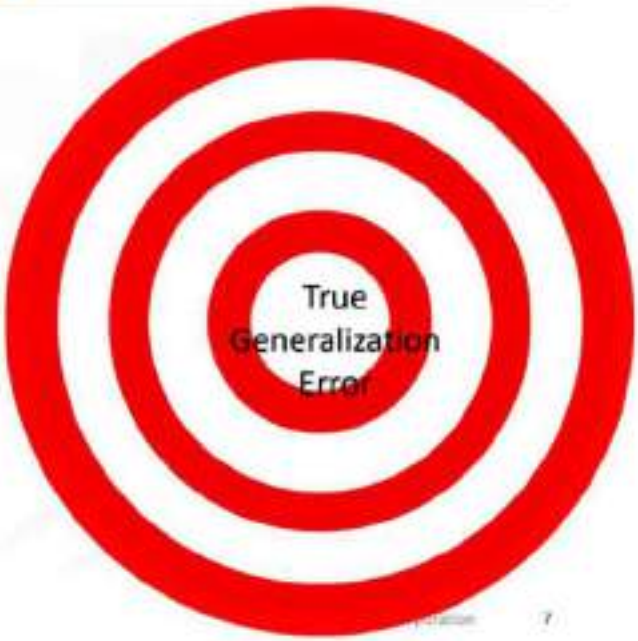


COGNITIVE CLASS.ai

Lots of Training Data



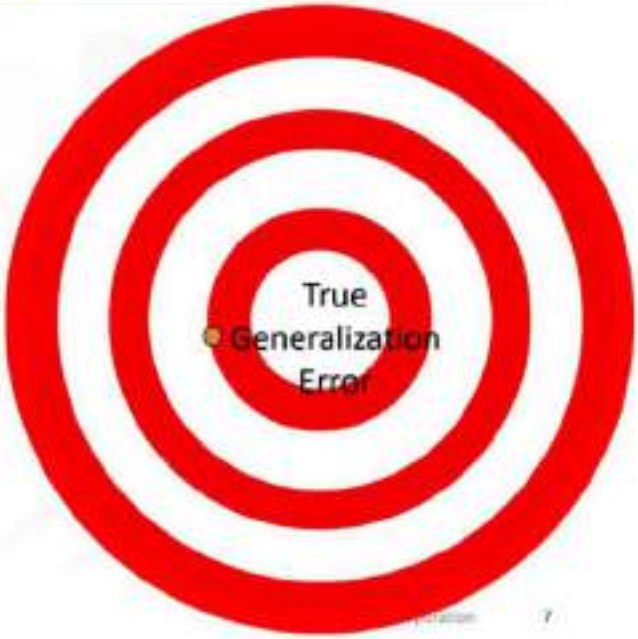
Model



COGNITIVE CLASS.ai

Lots of Training Data

Model



COGNITIVE CLASS.ai

Lots of Training Data

Model

COGNITIVE CLASS.ai

Lots of Training Data

Model

good approximation of the generalization error, but the precision is poor

COGNITIVE CLASS.ai

Lots of Training Data

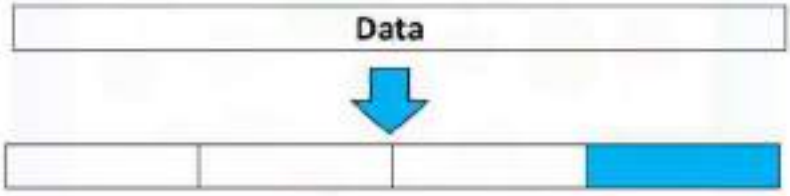
If we use fewer data points to train the model and more to test the model, the accuracy of the generalization performance will be less, but the model will have good precision.

All estimates are relatively close together, but they are further away from the true generalization performance.

COGNITIVE CLASS.ai

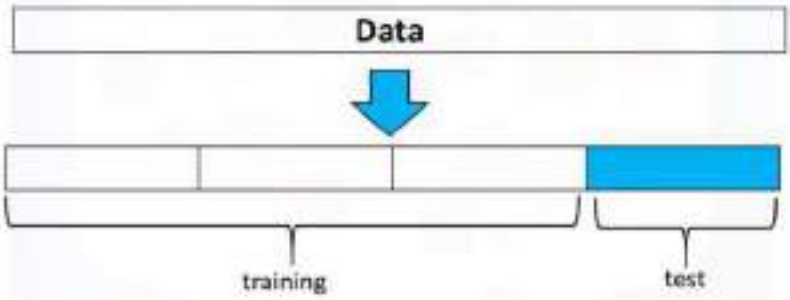
Cross Validation

- Most common out-of-sample evaluation metrics
- More effective use of data (each observation is used for both training and testing)



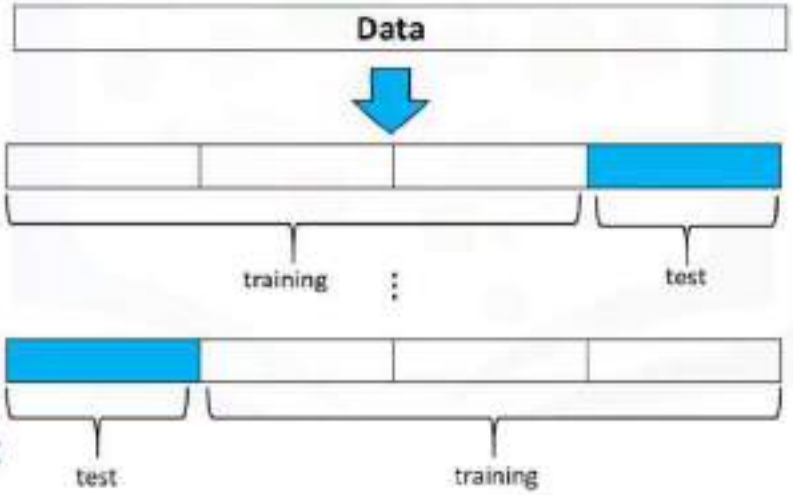
Cross Validation

- Most common out-of-sample evaluation metrics
- More effective use of data (each observation is used for both training and testing)



Cross Validation

- Most common out-of-sample evaluation metrics
- More effective use of data (each observation is used for both training and testing)



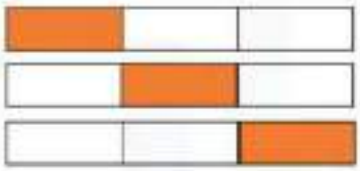
Function cross_val_score()

```
from sklearn.model_selection import cross_val_score

scores= cross_val_score(lr, x_data, y_data, cv=3)

np.mean(scores)
```

Function cross_val_score()

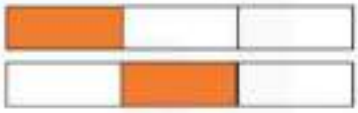


Model

scores

COGNITIVE CLASS.ai

Function cross_val_score()



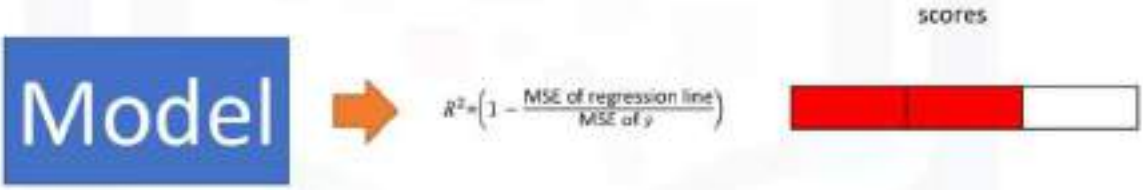
Model

$$R^2 = \left(1 - \frac{\text{MSE of regression line}}{\text{MSE of } y}\right)$$

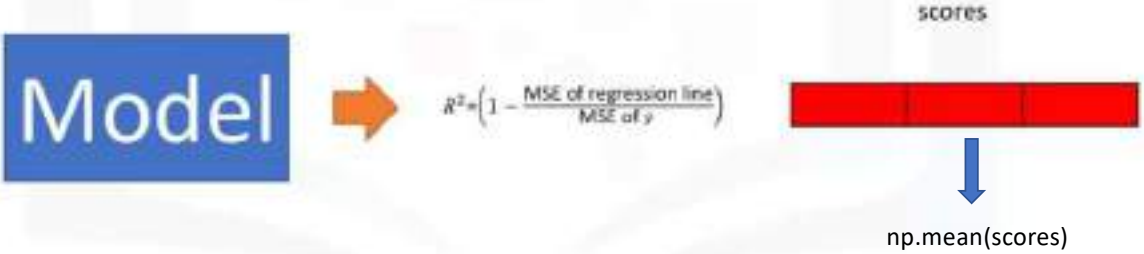
scores

COGNITIVE CLASS.ai

Function cross_val_score()



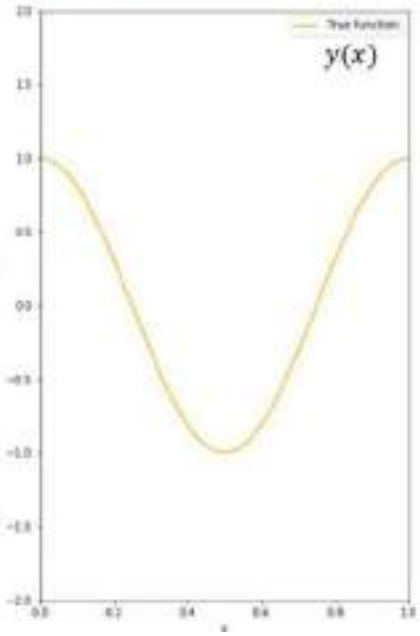
Function cross_val_score()



Over-fitting, Under-fitting and Model Selection

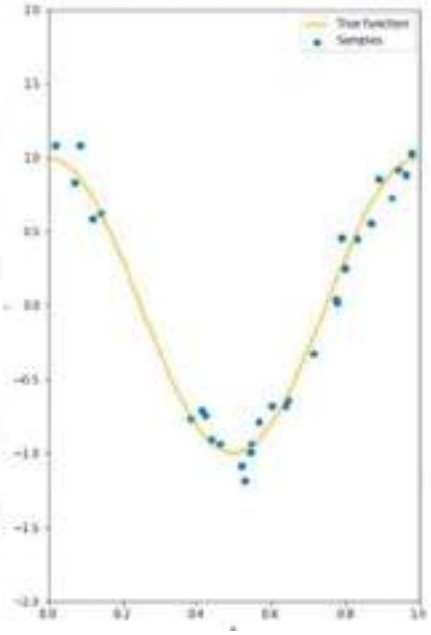
© 2017 IBM Corporation

Model Selection



© 2017 IBM Corporation

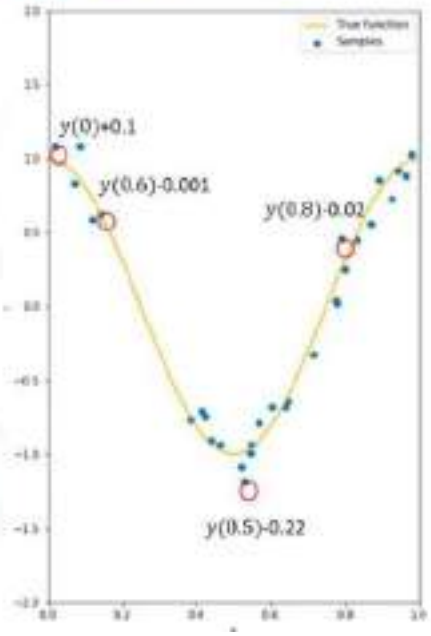
Model Selection



© 2017 IBM Corporation

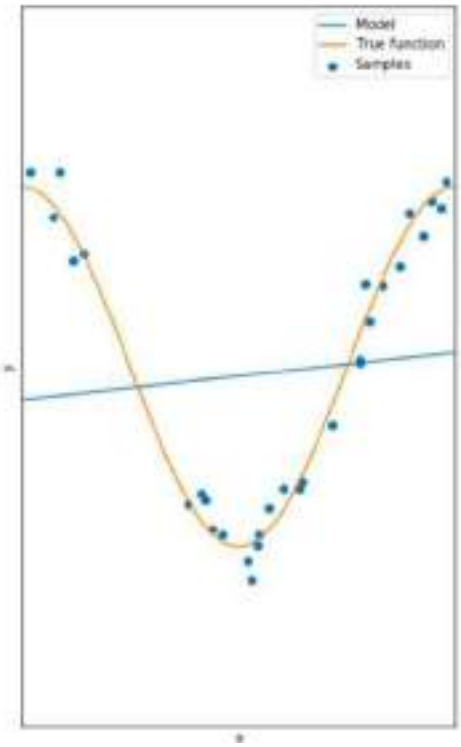
Model Selection

$y(x)$+noise



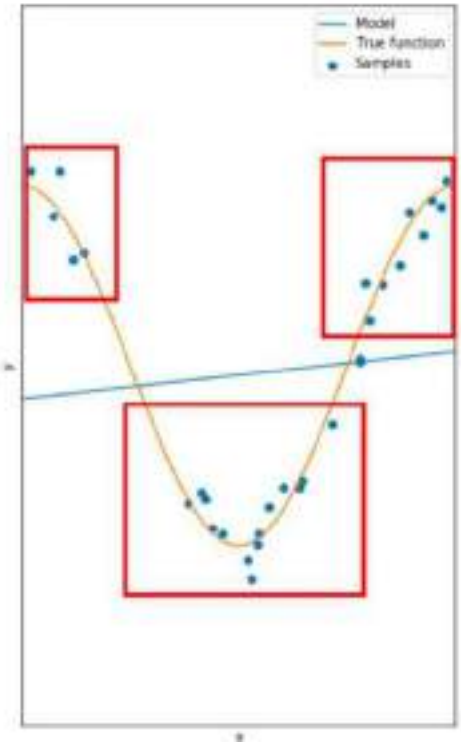
© 2017 IBM Corporation

$$y = b_0 + b_1\, x$$



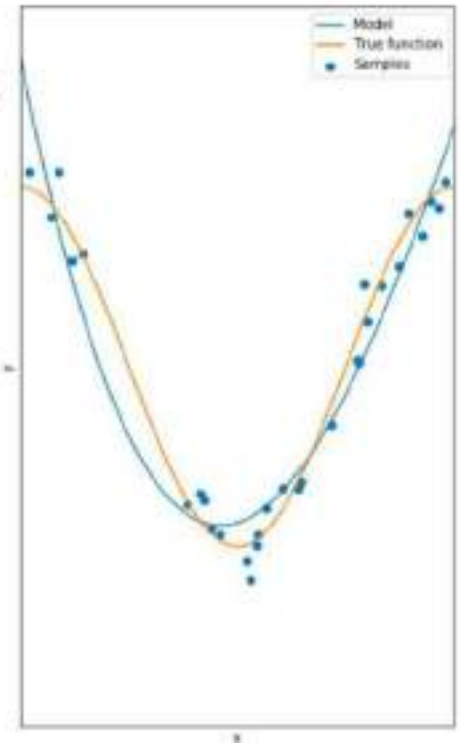
© 2017 IBM Corporation

$$y = b_0 + b_1\, x$$



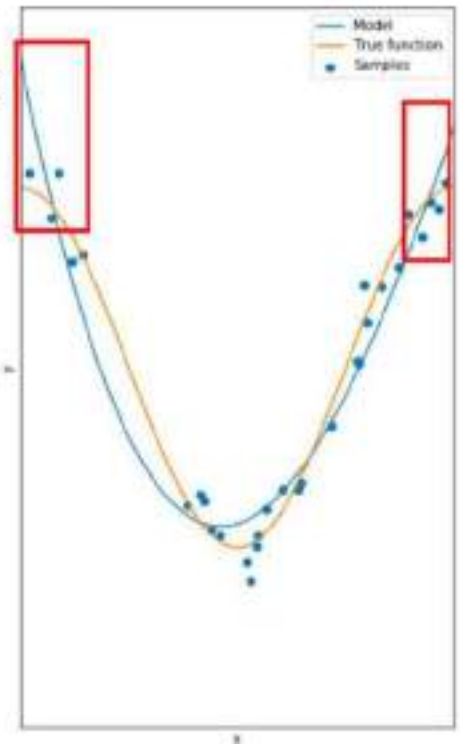
© 2017 IBM Corporation

$$y = b_0 + b_1\, x + b_2\, x^2$$



© 2017 IBM Corporation

$$y = b_0 + b_1\, x + b_2\, x^2$$

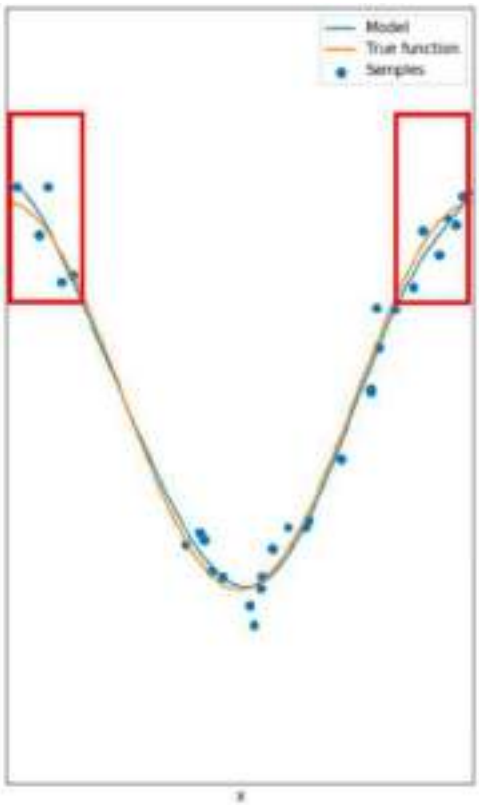


© 2017 IBM Corporation

$$y = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + b_5 x^5 + b_6 x^6 + b_7 x^7 + b_8 x^8$$

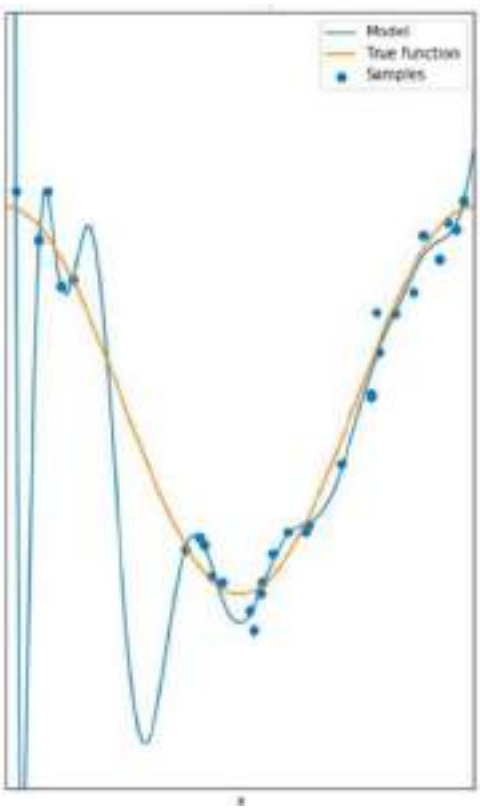
© 2017 IBM Corporation

$$y = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + b_5 x^5 + b_6 x^6 + b_7 x^7 + b_8 x^8$$



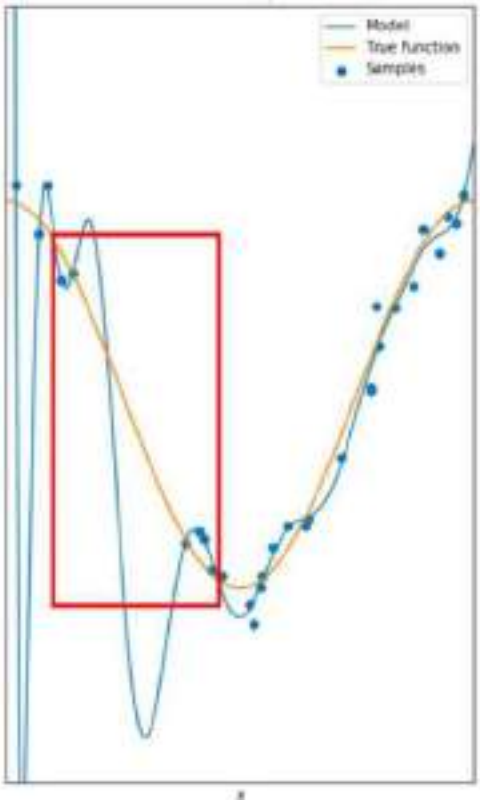
© 2017 IBM Corporation

$$\hat{y} = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + b_5 x^5 + b_6 x^6 + b_7 x^7 + b_8 x^8 + \ldots$$
$$+ b_9 x^9 + b_{10} x^{10} + b_{11} x^{11} + b_{12} x^{12} + b_{13} x^{13} + b_{14} x^{14} + b_{15} x^{15} + b_{16} x^{16}$$



© 2017 IBM Corporation

$$\hat{y} = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + b_5 x^5 + b_6 x^6 + b_7 x^7 + b_8 x^8 + \ldots$$
$$+ b_9 x^9 + b_{10} x^{10} + b_{11} x^{11} + b_{12} x^{12} + b_{13} x^{13} + b_{14} x^{14} + b_{15} x^{15} + b_{16} x^{16}$$



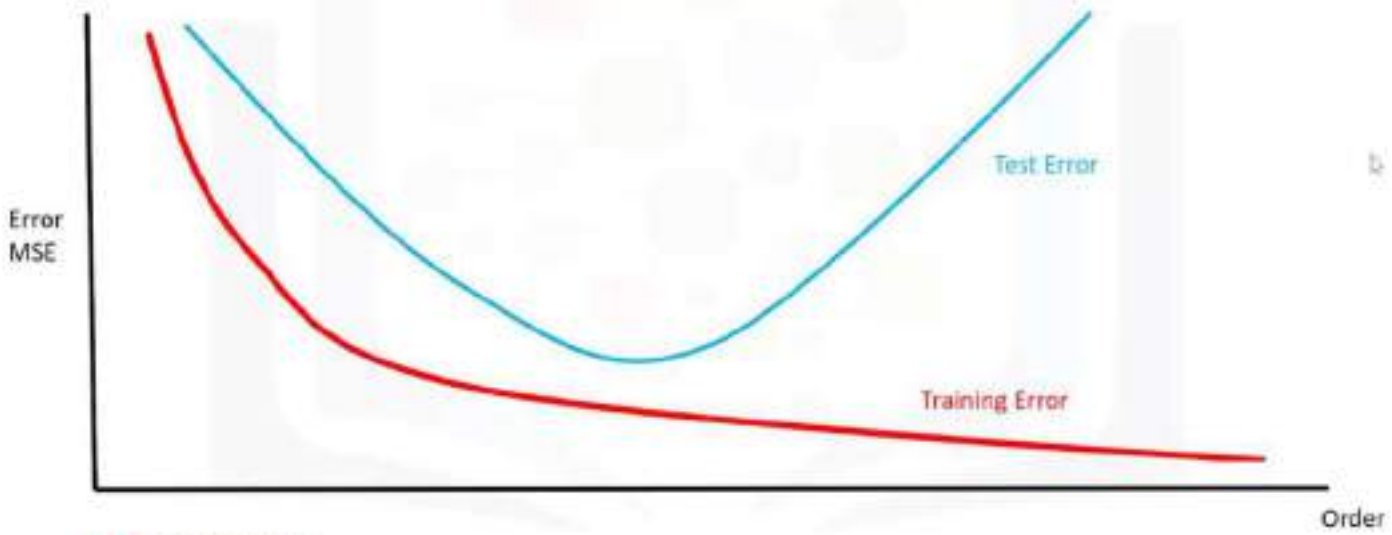
© 2017 IBM Corporation

Model Selection

The training error decreases with the order of the polynomial.

COGNITIVE CLASS.ai
© 2017 IBM Corporation

Model Selection



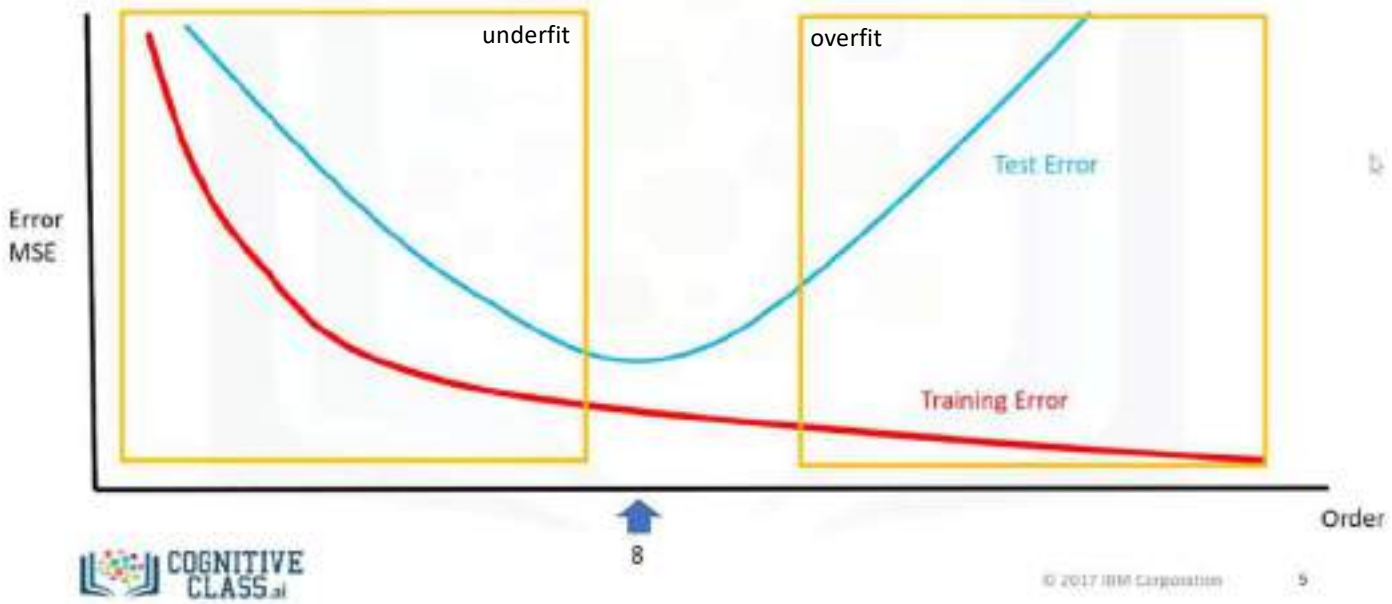
COGNITIVE CLASS.ai
© 2017 IBM Corporation

Model Selection

COGNITIVE CLASS.ai

© 2017 IBM Corporation

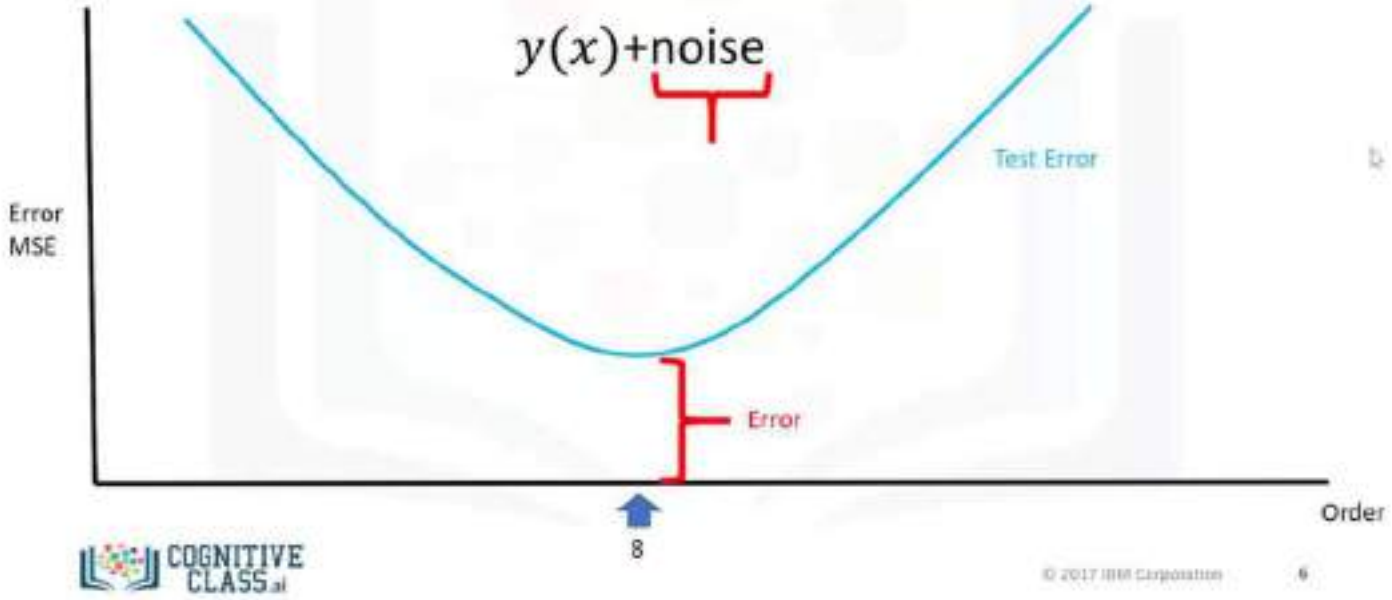
Model Selection



COGNITIVE CLASS.ai

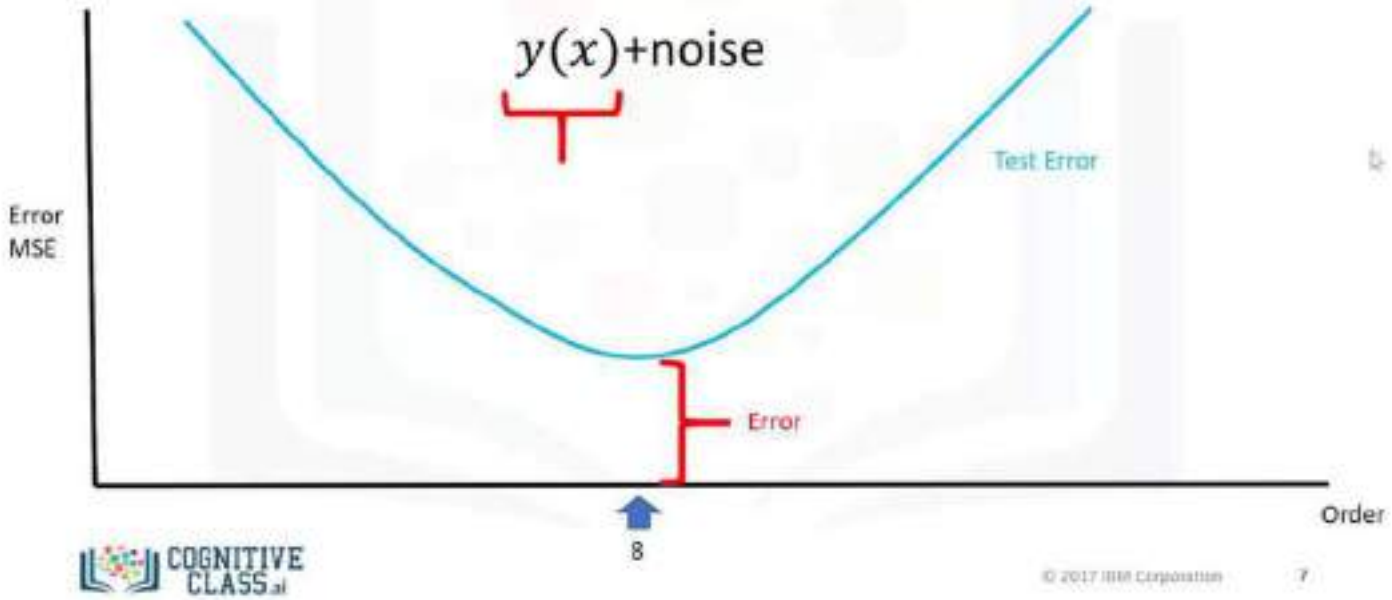
© 2017 IBM Corporation

Model Selection



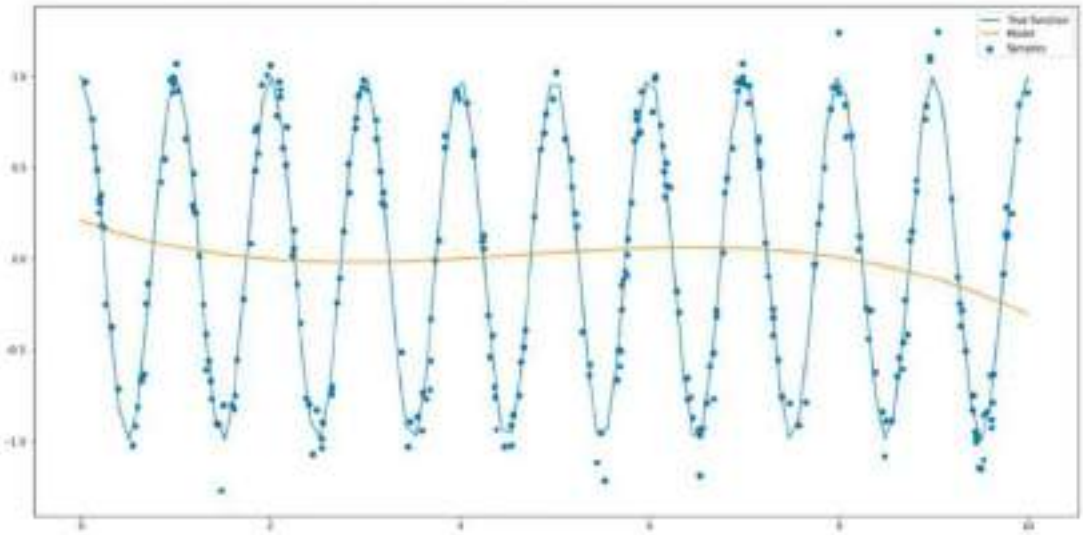
COGNITIVE CLASS.ai
© 2017 IBM Corporation

Model Selection



COGNITIVE CLASS.ai
© 2017 IBM Corporation

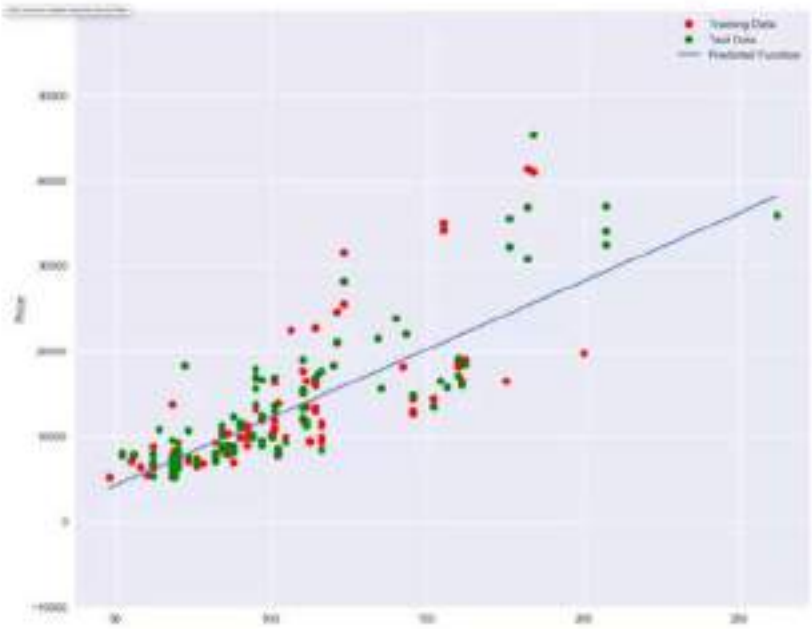
Model Selection



© 2017 IBM Corporation

If we just use the mean of the data, our model does not perform well.

© 2017 IBM Corporation

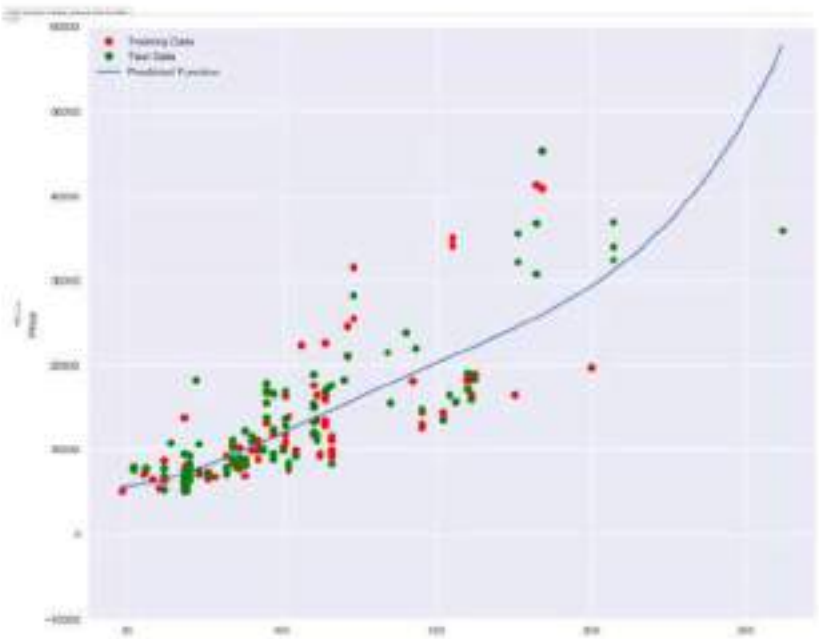


A linear function does fit the data better.

© 2017 IBM Corporation

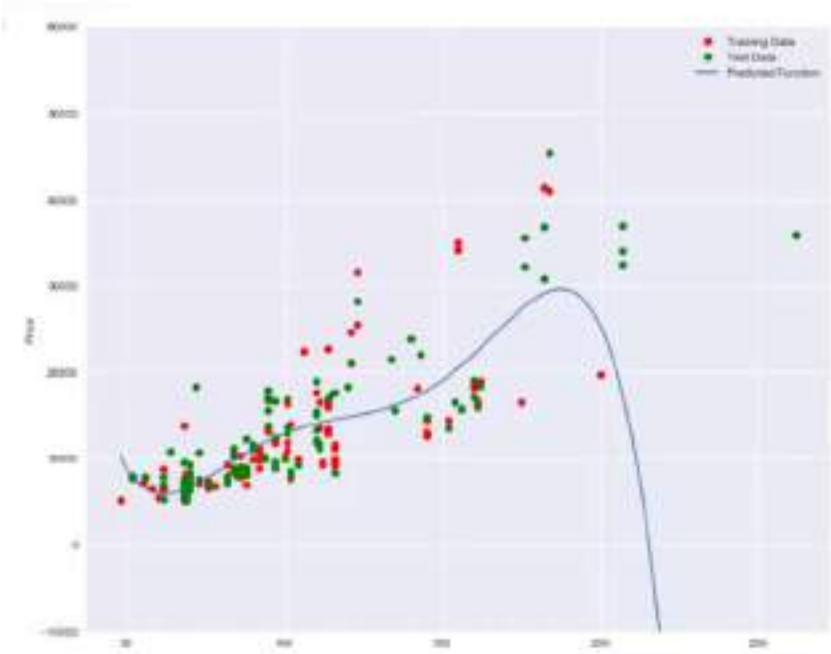
A second order model looks similar to the linear function.

© 2017 IBM Corporation



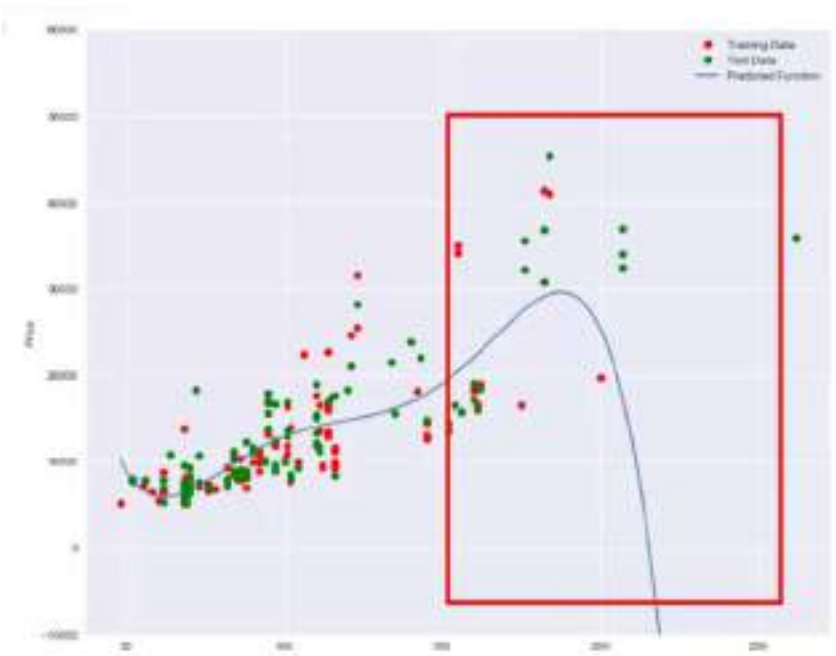
A third order function also appears to increase, like the previous two orders.

© 2017 IBM Corporation



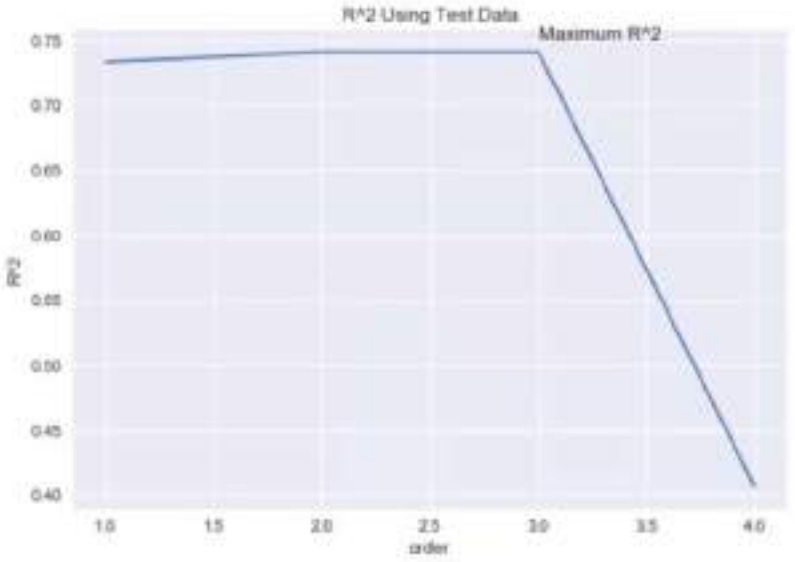
fourth order polynomial ☹

© 2017 IBM Corporation



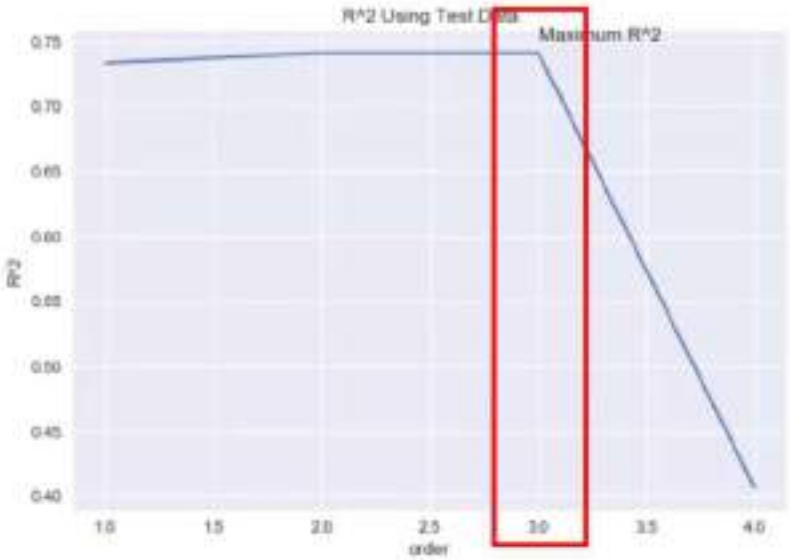
© 2017 IBM Corporation

Model Selection



Corporation

Model Selection




```
Rsqu_test=[]
order=[1,2,3,4]
for n in order:
    pr=PolynomialFeatures(degree=n)
    x_train_pr=pr.fit_transform(x_train[['horsepower']])
    x_test_pr=pr.fit_transform(x_test[['horsepower']])
    lr.fit(x_train_pr,y_train)
    Rsqu_test.append(lr.score(x_test_pr,y_test))
```

https://www.coursera.org/professional-certificates/ibm-data-science

coursera

This lecture was only a reduced version of the **Data Analysis in Python** course Offered by **IBM** on **Coursera**

Data Visualization with Python

Introduction to Data Visualization

COGNITIVE CLASS.ai

Why Build Visuals?

1. For exploratory data analysis
2. Communicate data clearly
3. Share unbiased representation of data
4. Use them to support recommendations to different stakeholders

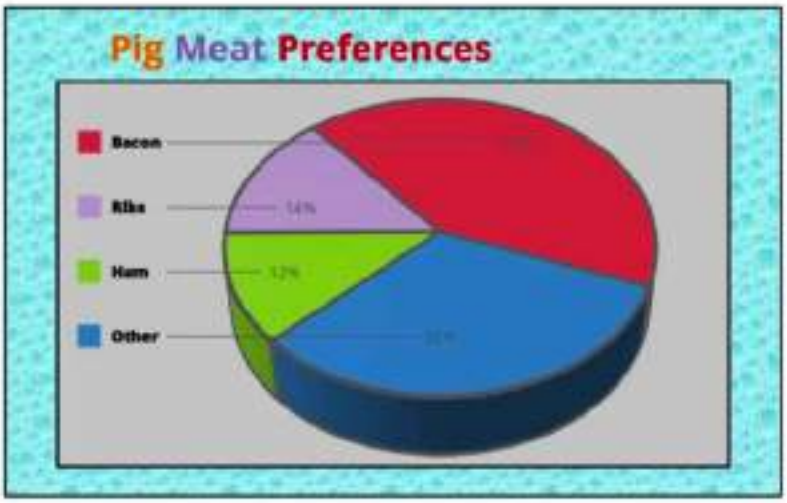
COGNITIVE CLASS.ai

Best Practices

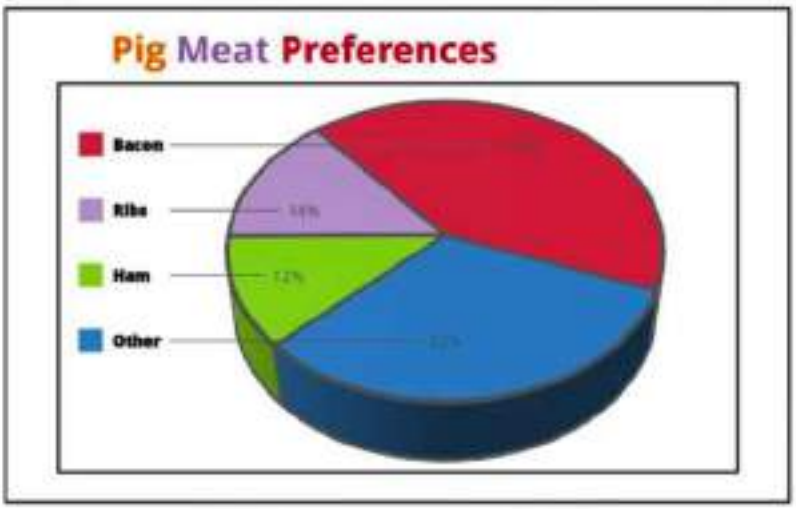
When creating a visual, always remember:

1. Less is more effective
2. Less is more attractive
3. Less is more impactive

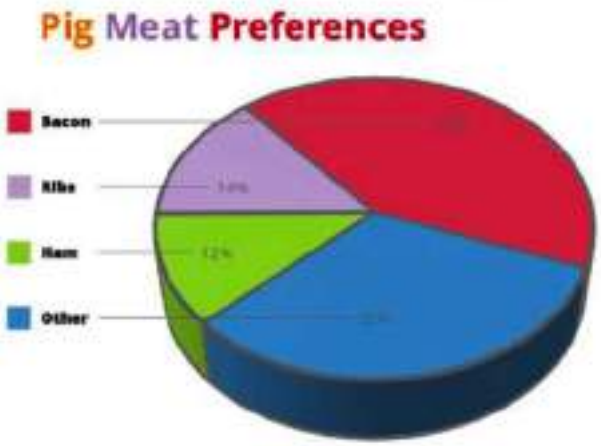
Example



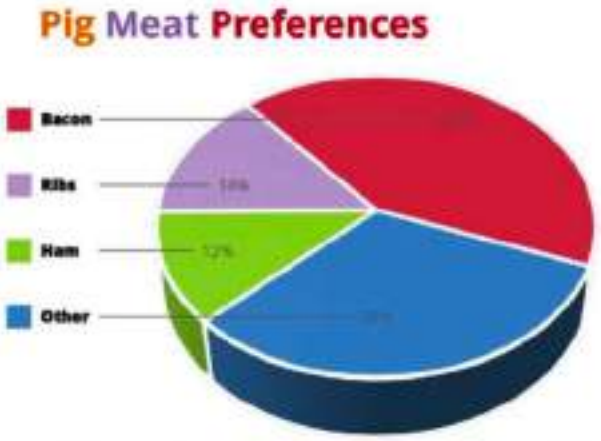
Example – Remove Background



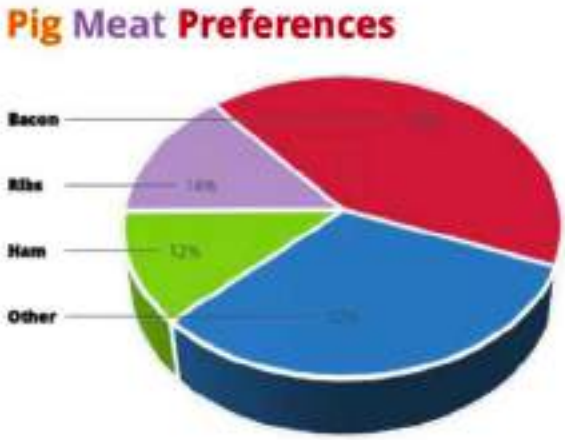
Example – Remove Borders



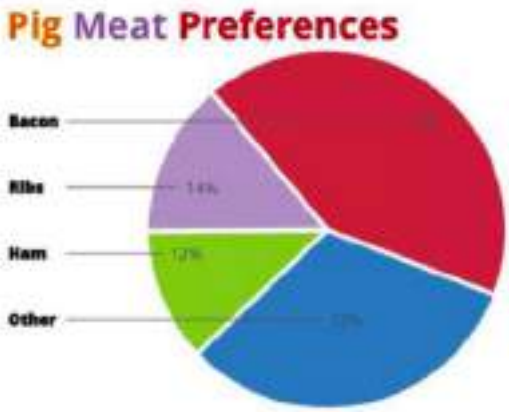
Example – Remove Borders



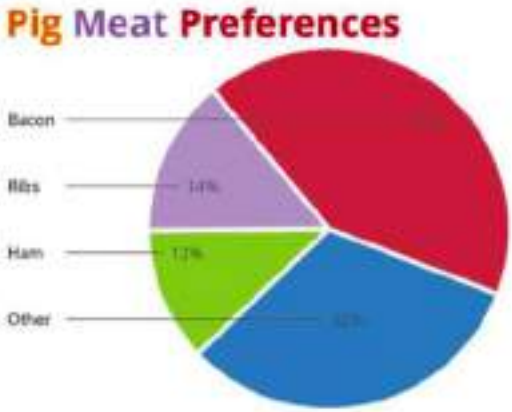
Example – Remove Redundant Legend



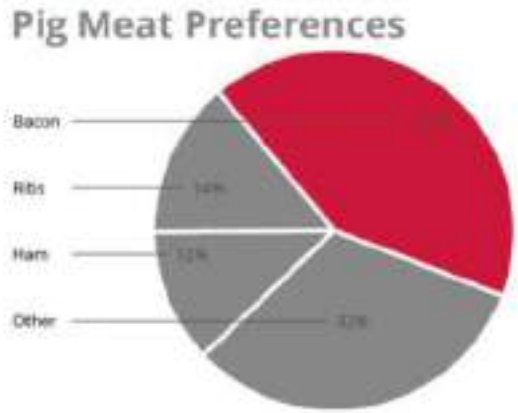
Example – Remove 3D



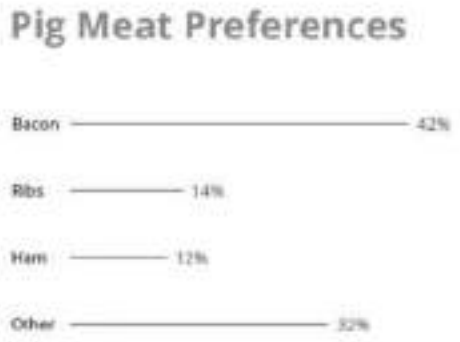
Example – Remove Text Bolding



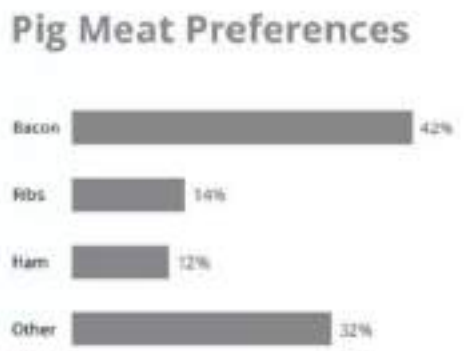
Example – Reduce Color



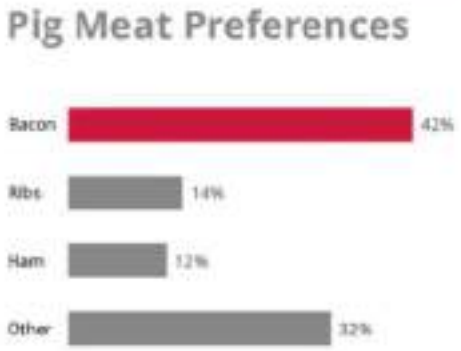
Example – Remove Wedges



Example - Thicken Lines

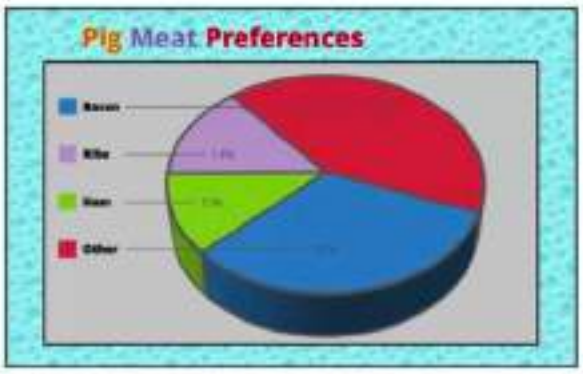


Example – Emphasize Bacon

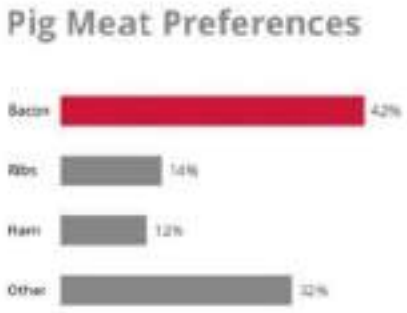


Comparison

Before



After



More Examples

For more examples, check out:

http://www.darkhorseanalytics.com/

Data Visualization with Python

Introduction to Matplotlib

© 2017 IBM Corporation

Matplotlib - History

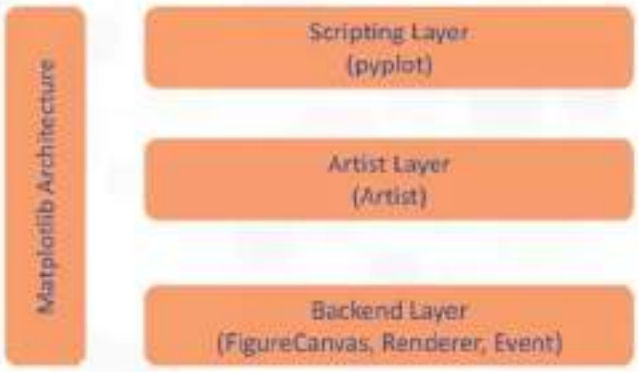
John Hunter (1968 – 2012)

EEG/ECoG Visualization Tool

Analogous to Matlab scripting interface

© 2017 IBM Corporation

Matplotlib Architecture



© 2017 IBM Corporation

Backend Layer

Has three built-in abstract interface classes:

1. FigureCanvas: **matplotlib.backend_bases.FigureCanvas**
   - Encompasses the area onto which the figure is drawn
2. Renderer: **matplotlib.backend_bases.Renderer**
   - Knows how to draw on the FigureCanvas
3. Event: **matplotlib.backend_bases.Event**
   - Handles user inputs such as keyboard strokes and mouse clicks

© 2017 IBM Corporation

Artist Layer

- Comprised of one main object – **Artist**:
  - Knows how to use the Renderer to draw on the canvas.
- Title, lines, tick labels, and images, all correspond to individual **Artist** instances.
- Two types of **Artist** objects:
  1. **Primitive**: Line2D, Rectangle, Circle, and Text.
  2. **Composite**: Axis, Tick, Axes, and Figure.

COGNITIVE CLASS.ai

© 2017 IBM Corporation

Artist Layer

- Comprised of one main object – **Artist**:
  - Knows how to use the Renderer to draw on the canvas.
- Title, lines, tick labels, and images, all correspond to individual **Artist** instances.
- Two types of **Artist** objects:
  1. **Primitive**: Line2D, Rectangle, Circle, and Text.
  2. **Composite**: Axis, Tick, Axes, and Figure.
- Each *composite* artist may contain other *composite* artists as well as *primitive* artists.

COGNITIVE CLASS.ai

© 2017 IBM Corporation

5

Putting the Artist Layer to Use

- Let's try to generate a histogram of some data using the **Artist** layer:

```
from matplotlib.backends.backend_agg import FigureCanvasAgg as FigureCanvas # import FigureCanvas
from matplotlib.figure import Figure # import figure artist
fig = Figure()
canvas = FigureCanvas(fig)

# create 10000 random numbers using numpy
import numpy as np
x = np.random.randn(10000)

ax = fig.add_subplot(111) # create an axes artist

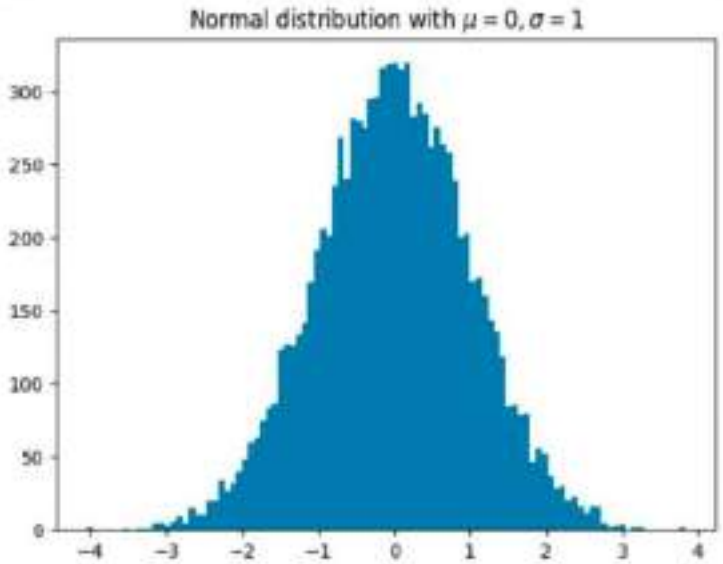
ax.hist(x, 100) # generate a histogram of the 10000 numbers

# add a title to the figure and save it
ax.set_title('Normal distribution with $\mu=0, \sigma=1$')
fig.savefig('matplotlib_histogram.png')
```

COGNITIVE CLASS.ai

© 2017 IBM Corporation

Putting the Artist Layer to Use



COGNITIVE CLASS.ai

© 2017 IBM Corporation

Scripting Layer

- Comprised mainly of **pyplot**, a scripting interface that is lighter that the **Artist** layer.
- Let's see how we can generate the same histogram of 10000 random values using the **pyplot** interface.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.randn(10000)
plt.hist(x, 100)
plt.title(r'Normal distribution with $\mu=0, \sigma=1$')
plt.savefig('matplotlib_histogram.png')
plt.show()
```


© 2017 IBM Corporation

Further Reading

You can find more information about Matplotlib: its history and architecture, by following the link below:

http://www.aosabook.org/en/matplotlib.html

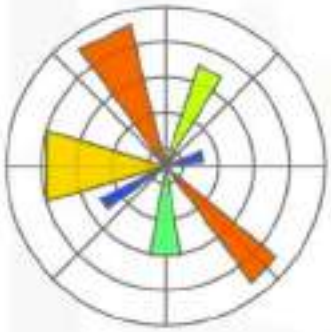
© 2017 IBM Corporation

Data Visualization with Python

Basic Plotting with Matplotlib

COGNITIVE CLASS.ai

Matplotlib – Jupyter Notebook



jupyter

COGNITIVE CLASS.ai

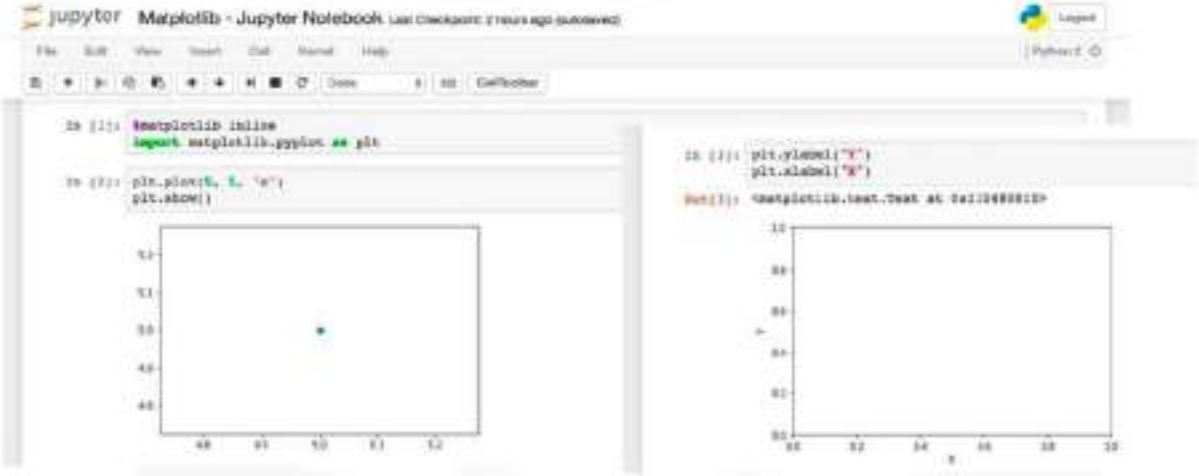
Matplotlib – Jupyter Notebook

Matplotlib – Jupyter Notebook

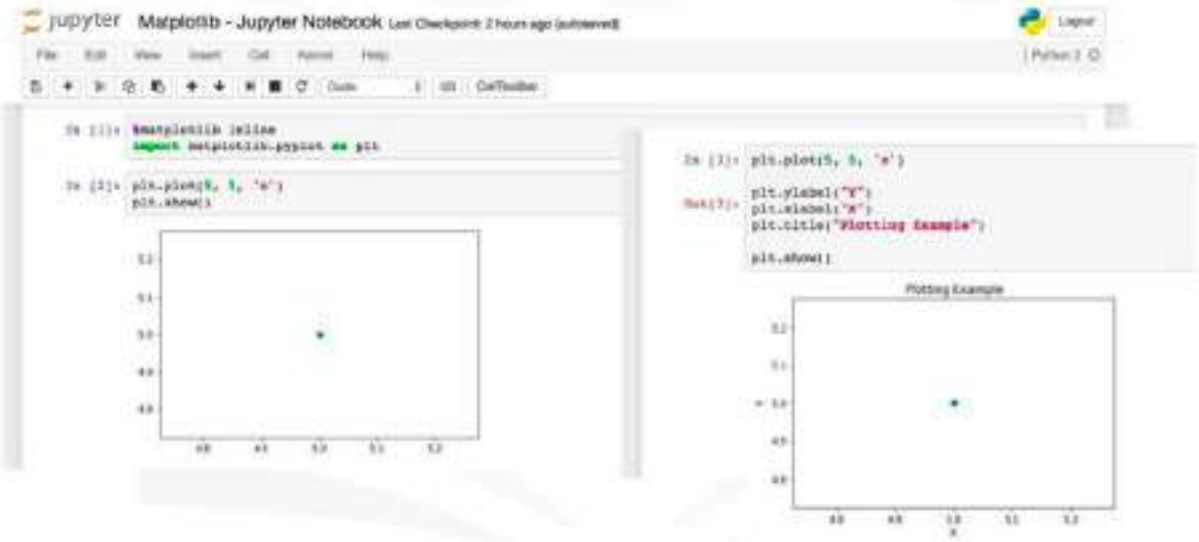
Matplotlib – Plot Function

Matplotlib Backends – Inline

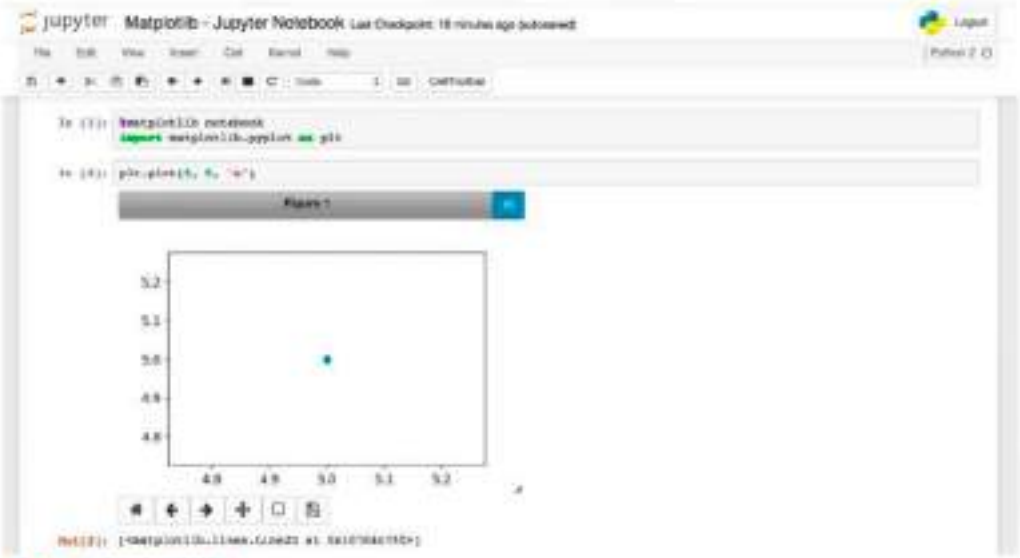
Matplotlib Backends – Inline



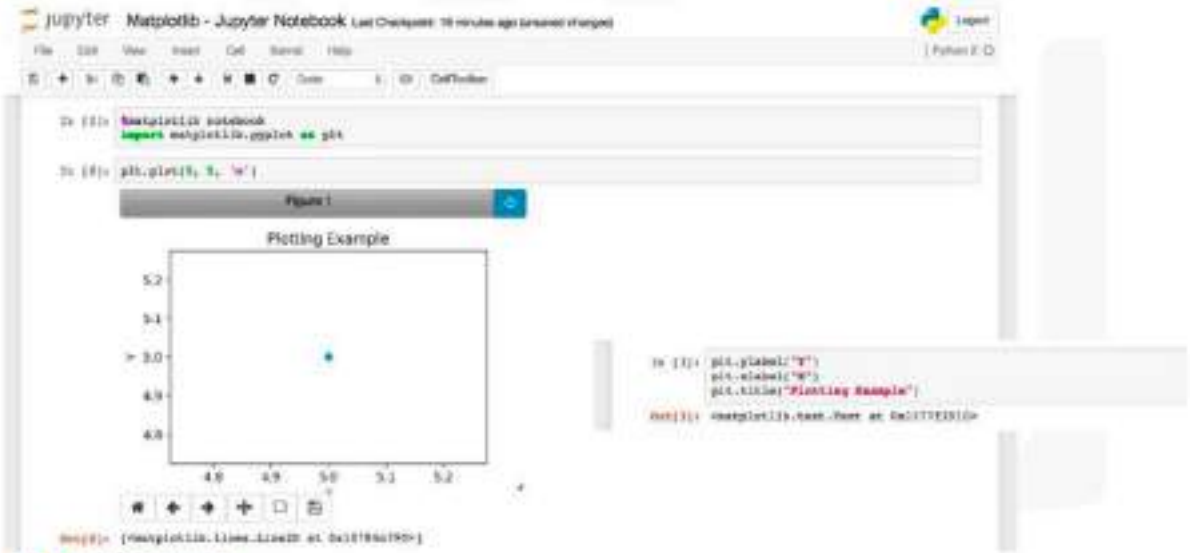
Matplotlib Backends – Inline



Matplotlib Backends – Notebook



Matplotlib Backends – Notebook



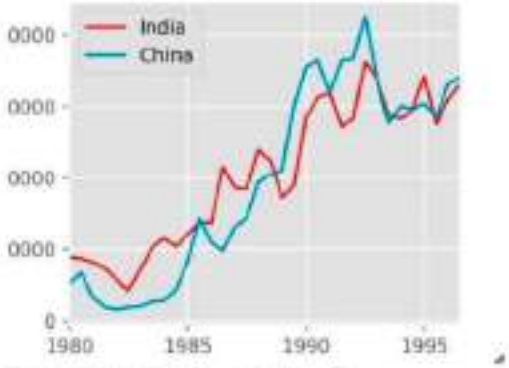
Matplotlib – Pandas

india_china_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df.plot(kind="line")
```

Figure 1



COGNITIVE CLASS.ai

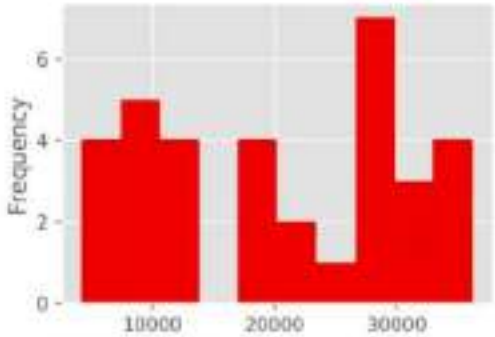
Matplotlib – Pandas

india_china_df

	India	China
1980	8880	5123
1981	8670	6682
1982	8147	3308
1983	7338	1863
1984	5704	1527

```
india_china_df['India'].plot(kind="hist")
```

Figure 1



COGNITIVE CLASS.ai

Data Visualization with Python

Dataset on Immigration to Canada

COGNITIVE CLASS.ai

Dataset

- The Population Division of the United Nations compiled data pertaining to 45 countries.
- For each country, annual data on the flows of international migrants is reported in addition to other metadata.
- We will primarily work with a United Nations data on immigration to Canada.

COGNITIVE CLASS.ai

Immigration Data to Canada

United Nations
Population Division
Department of Economic and Social Affairs

International Migration Flows to and from Selected Countries: The 2015 Revision

POP/DB/MIG/Flow/Rev.2015

December 2015 - Copyright © 2015 by United Nations. All rights reserved

Suggested citation: United Nations, Department of Economic and Social Affairs, Population Division (2015).
International Migration Flows to and from Selected Countries: The 2015 Revision. (United Nations database.

Reporting country: Canada
Criterion: Citizenship

Classification		Origin/Destination	Major area		Region		Development region								
Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982	1983	1984	1985
Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	39	39	47	71	340
Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	0	0	0	0	0
Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	67	71	69	63	44
Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	1	0	0	0	0

COGNITIVE CLASS.ai

Read Data into Pandas Dataframe

```
import numpy as np  # useful for many scientific computing in Python
import pandas as pd # primary data structure library
from __future__ import print_function # adds compatibility to python 2
```

```
# install xlrd
!pip install xlrd

print('xlrd installed!')
```

```
df_can = pd.read_excel(
    'https://ibm.box.com/shared/static/lw190pt9zpy5bd1ptyg2aw15awomz9pu.xlsx',
    sheetname="Canada by Citizenship",
    skiprows=range(20),
    skip_footer = 2)
```

COGNITIVE CLASS.ai

Display Dataframe

```
df_can.head()
```

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	...	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	...	2978	3436	3009	2652	2111	1746	1758	2203	2635	2004
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	...	1450	1223	856	702	560	716	561	539	620	603
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	...	3616	3626	4807	3623	4005	5393	4752	4325	3774	4331
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	...	0	0	1	0	0	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	...	0	0	1	1	0	0	0	0	1	1

Dataset - Recap

	Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	...	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
0	Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	...	2978	3436	3009	2652	2111	1746	1758	2203	2635	2004
1	Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	...	1450	1223	856	702	560	716	561	539	620	603
2	Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	...	3616	3626	4807	3623	4005	5393	4752	4325	3774	4331
3	Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	...	0	0	1	0	0	0	0	0	0	0
4	Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	...	0	0	1	1	0	0	0	0	1	1

Dataset - Processed

	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Country																					
Afghanistan	Asia	Southern Asia	Developing regions	16	39	39	47	71	340	496	...	3436	3009	2652	2111	1746	1758	2203	2635	2004	58639
Albania	Europe	Southern Europe	Developed regions	1	0	0	0	0	0	1	...	1223	856	702	560	716	561	539	620	603	15699
Algeria	Africa	Northern Africa	Developing regions	80	67	71	69	63	44	69	...	3626	4807	3623	4005	5393	4752	4325	3774	4331	69439
American Samoa	Oceania	Polynesia	Developing regions	0	1	0	0	0	0	0	...	0	1	0	0	0	0	0	0	0	6
Andorra	Europe	Southern Europe	Developed regions	0	0	0	0	0	0	2	...	0	1	1	0	0	0	0	1	1	15

COGNITIVE CLASS.ai

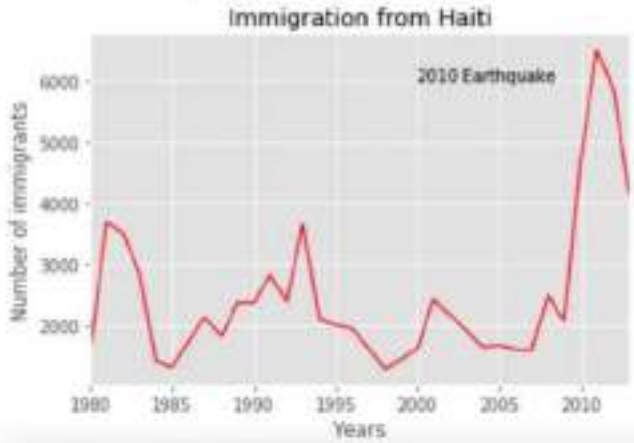
Data Visualization with Python

Line Plots

COGNITIVE CLASS.ai

Line Plots

A line plot is a type of plot which displays information as a series of data points called 'markers' connected by straight line segments.



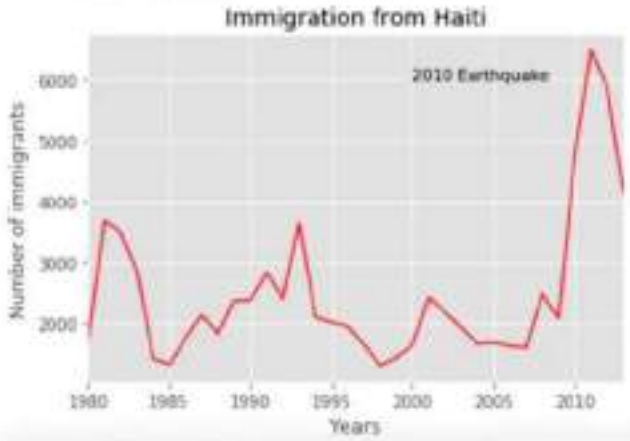
Creating Line Plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
years = list(map(str, range(1980, 2014)))

df_canada.loc['Haiti', years].plot(kind = 'line')
plt.title('Immigration from Haiti')
plt.ylabel('Number of immigrants')
plt.xlabel('Years')

plt.show()
```



Data Visualization with Python

Area Plots

COGNITIVE CLASS.ai

Area Plot

- Also known as area chart or area graph.
- Commonly used to represent cumulated totals using numbers or percentages over time.
- Is based on the line plot.

COGNITIVE CLASS.ai

Generating Area Plots

```
df_canada.sort_values(['Total'], ascending = False, axis = 0, inplace = True)
```

	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Country																					
India	Asia	Southern Asia	Developing regions	8880	8670	8147	7338	5704	4211	7150	...	36210	33848	28742	28261	29456	34235	27509	30933	33087	691904
China	Asia	Eastern Asia	Developing regions	5123	6682	3308	1863	1527	1816	1960	...	42584	33518	27642	30037	29622	30391	28502	33024	34129	659962
United Kingdom of Great Britain and Northern Ireland	Europe	Northern Europe	Developed regions	22045	24796	20620	10015	10170	9564	9470	...	7258	7140	8216	8979	8876	8724	6204	6195	5827	551500
Philippines	Asia	South-Eastern Asia	Developing regions	6051	5921	5249	4562	3801	3150	4166	...	18139	18400	19837	24887	28573	38617	36765	34315	29544	511391
Pakistan	Asia	Southern Asia	Developing regions	978	972	1201	900	668	514	691	...	14314	13127	10124	8994	7217	6811	7468	11227	12603	241600

India

COGNITIVE CLASS.ai

Generating Area Plots

```
df_canada.sort_values(['Total'], ascending = False, axis = 0, inplace = True)
```

	Continent	Region	DevName	1980	1981	1982	1983	1984	1985	1986	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Country																					
India	Asia	Southern Asia	Developing regions	8880	8670	8147	7338	5704	4211	7150	...	36210	33848	28742	28261	29456	34235	27509	30933	33087	691904
China	Asia	Eastern Asia	Developing regions	5123	6682	3308	1863	1527	1816	1960	...	42584	33518	27642	30037	29622	30391	28502	33024	34129	659962
United Kingdom of Great Britain and Northern Ireland	Europe	Northern Europe	Developed regions	22045	24796	20620	10015	10170	9564	9470	...	7258	7140	8216	8979	8876	8724	6204	6195	5827	551500
Philippines	Asia	South-Eastern Asia	Developing regions	6051	5921	5249	4562	3801	3150	4166	...	18139	18400	19837	24887	28573	38617	36765	34315	29544	511391
Pakistan	Asia	Southern Asia	Developing regions	978	972	1201	900	668	514	691	...	14314	13127	10124	8994	7217	6811	7468	11227	12603	241600

India China United Kingdom Philippines Pakistan

COGNITIVE CLASS.ai

Generating Area Plots

```
years = list(map(str, range(1980, 2014)))

df_canada.sort_values(['Total'], ascending = False, axis = 0, inplace = True)

df_top5 = df_canada.head()
df_top5 = df_top5[years].transpose()
```

Country	India	China	United Kingdom of Great Britain and Northern Ireland	Philippines	Pakistan
1980	8880	5123	22045	6051	978
1981	8670	6682	24796	5921	972
1982	8147	3308	20620	5249	1201
1983	7338	1863	10015	4562	900
1984	5704	1527	10170	3801	668

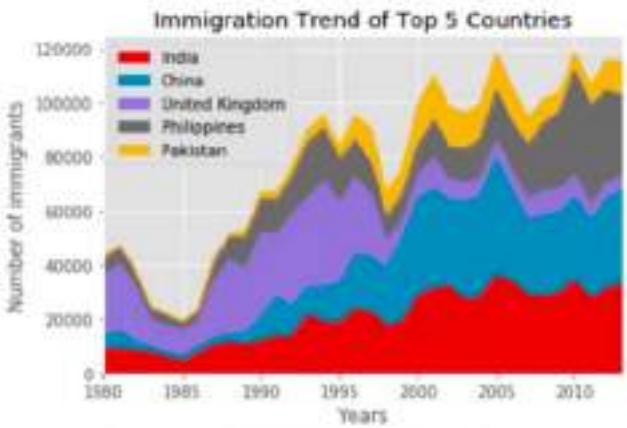
Area Plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_top5.plot(kind='area')

plt.title('Immigration trend of top 5 countries')
plt.ylabel('Number of immigrants')
plt.xlabel('Years')

plt.show()
```

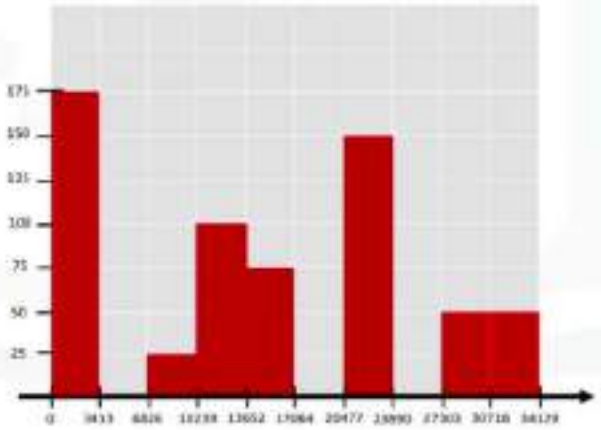


Data Visualization with Python

Histograms

Histogram

A histogram is a way of representing the frequency distribution of a variable.



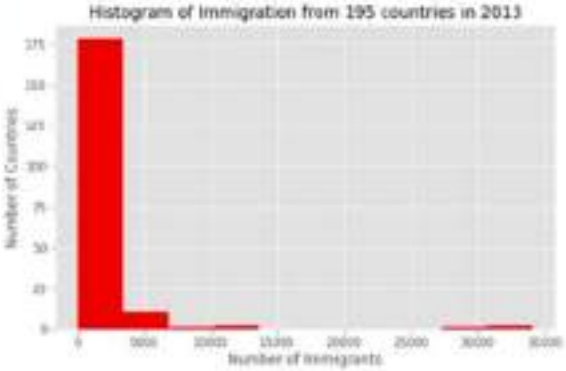
Histograms

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_canada['2013'].plot(kind='hist')

plt.title('Histogram of Immigration from 195 countries in 2013')
plt.ylabel('Number of Countries')
plt.xlabel('Number of Immigrants')

plt.show()
```



Histograms

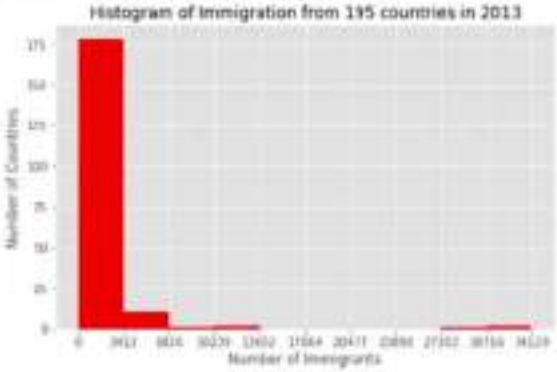
```
import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
```

```
count, bin_edges = np.histogram(df_canada['2013'])

df_canada['2013'].plot(kind='hist', xticks = bin_edges)

plt.title('Histogram of Immigration from 195 countries in 2013')
plt.ylabel('Number of Countries')
plt.xlabel('Number of Immigrants')

plt.show()
```

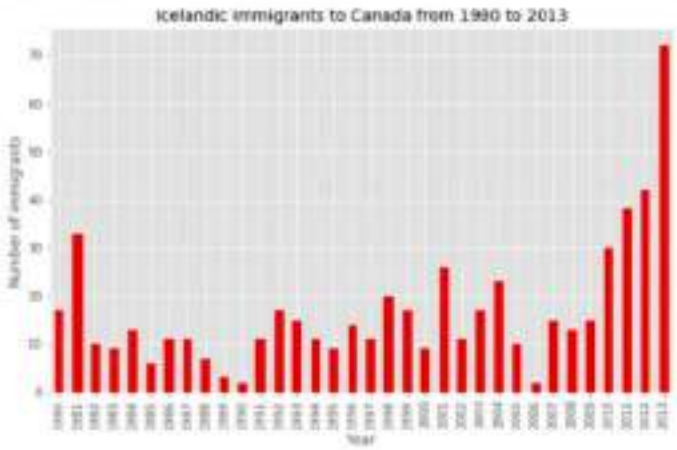


Data Visualization with Python

Bar Charts

Bar Chart

Unlike a histogram, a bar chart is commonly used to compare the values of a variable at a given point in time.



Bar Chart

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

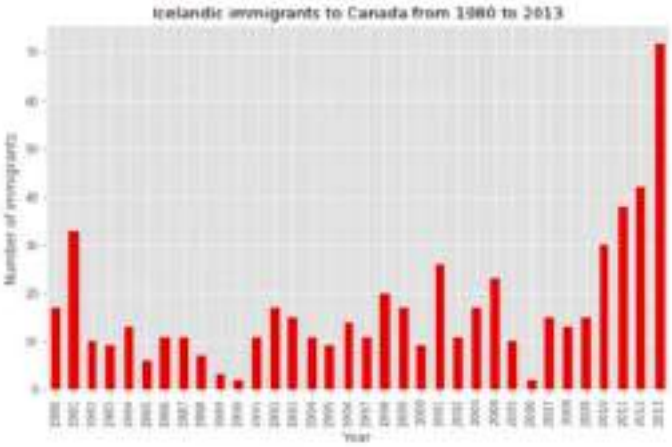
```
years = list(map(str, range(1980, 2014)))

df_iceland = df_canada.loc['Iceland', years]
```

```
df_iceland.plot(kind='bar')

plt.title('Icelandic immigrants to Canada from 1980 to 2013')
plt.xlabel('Year')
plt.ylabel('Number of immigrants')

plt.show()
```



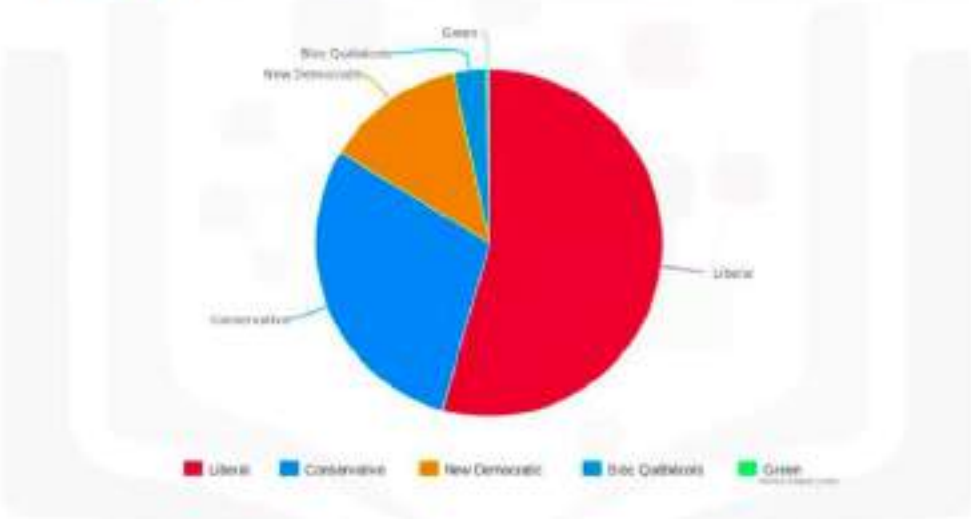
COGNITIVE CLASS.ai

Data Visualization with Python

Pie Charts

COGNITIVE CLASS.ai

Pie Chart



Pie Chart

```
df_continents = df_canada.groupby('Continent', axis = 0).sum()
```

	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	...	2005	2006	2007	2008	2009	2010	2011	2012	2013	Total
Continent																					
Africa	3951	4363	3819	2671	2639	2650	3782	7494	7552	9894	...	27523	29188	28284	29890	34534	40892	35441	38083	38543	618948
Asia	31025	34314	30214	24696	27274	23850	28739	43203	47454	60256	...	159253	149054	133459	139894	141434	163845	146894	152218	155075	3317794
Europe	39760	44802	42720	24638	22287	20844	24370	46698	54726	60893	...	35955	33053	33495	34692	35078	33425	26778	29177	28691	1410947
Latin America and the Caribbean	13081	15215	16769	15427	13678	15171	21179	28471	21924	25060	...	24747	24676	26011	26547	26867	28818	27856	27173	24950	765148
Northern America	9378	10030	9074	7100	6661	6543	7074	7705	6469	6790	...	8394	9613	9463	10190	8995	8142	7677	7892	8503	241142
Oceania	1942	1839	1675	1018	878	933	904	1200	1181	1539	...	1565	1473	1693	1804	1960	1834	1546	1679	1775	53174

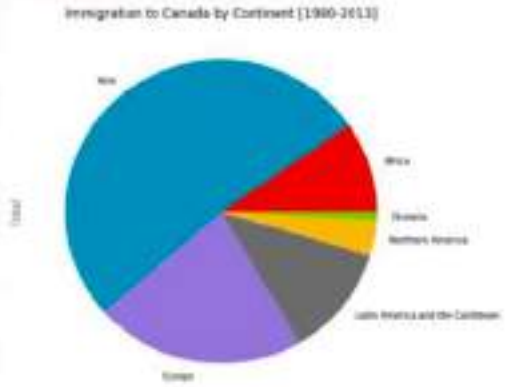
Pie Chart

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_continents['Total'].plot(kind='pie')

plt.title('Immigration to Canada by Continent [1980-2013]')

plt.show()
```



Pie Charts

Pie Charts Flaws

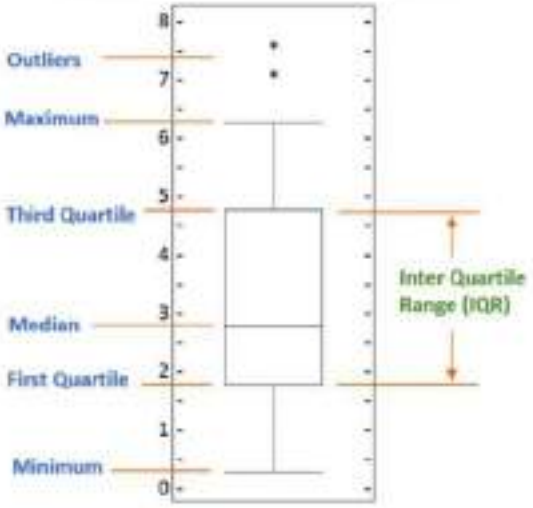
To learn more about the arguments again pie charts:

https://www.surveygizmo.com/survey-blog/pie-chart-or-bar-graph/

Data Visualization with Python

Box Plots

Box Plots



COGNITIVE CLASS.ai

Box Plots

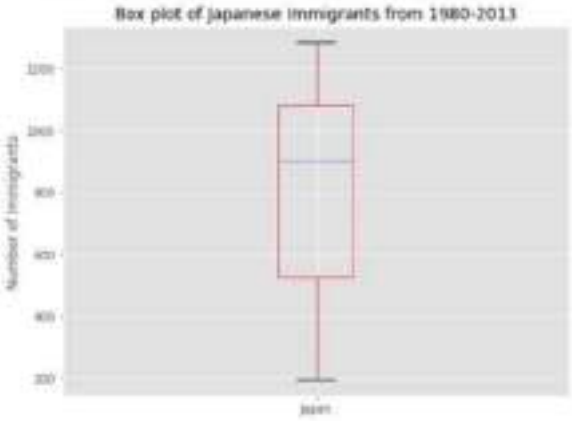
```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_japan = df_canada.loc[['Japan'], years].transpose()

df_japan.plot(kind='box')

plt.title('Box plot of Japanese Immigrants from 1980-2013')
plt.ylabel('Number of Immigrants')

plt.show()
```

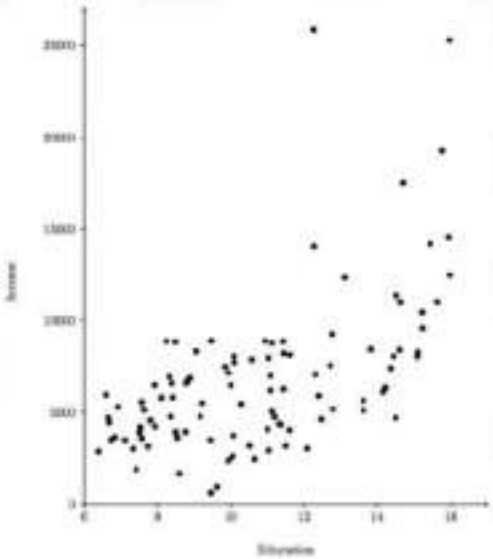


COGNITIVE CLASS.ai

Data Visualization with Python

Scatter Plots

Scatter Plots



Scatter Plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_total.plot(
    kind='scatter',
    x='year',
    y='total',
)

plt.title('Total Immigrant population to Canada from 1980 - 2013')
plt.xlabel ('Year')
plt.ylabel('Number of Immigrants')

plt.show()
```

df_total

year	total
1980	99137
1981	110563
1982	104271
1983	75550
1984	73417
⋮	⋮

COGNITIVE CLASS.ai

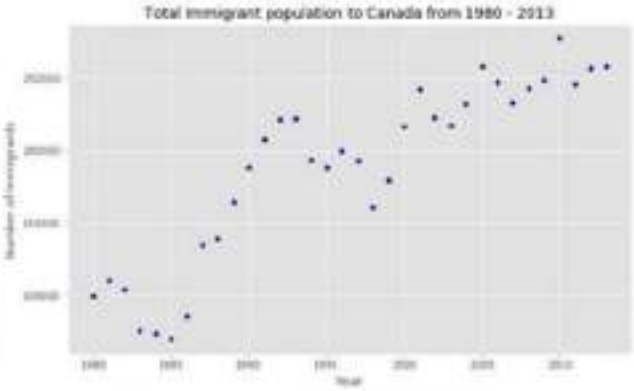
Scatter Plots

```
import matplotlib as mpl
import matplotlib.pyplot as plt
```

```
df_total.plot(
    kind='scatter',
    x='year',
    y='total',
)

plt.title('Total Immigrant population to Canada from 1980 - 2013')
plt.xlabel ('Year')
plt.ylabel('Number of Immigrants')

plt.show()
```



COGNITIVE CLASS.ai

Data Visualization with Python

Waffle Charts

COGNITIVE CLASS.ai

Waffle Charts

- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

COGNITIVE CLASS.ai

Waffle Charts

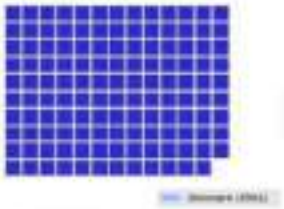
- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

Country	Total Immigrants
Denmark	3901
Norway	2327
Sweden	5866

Waffle Charts

- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

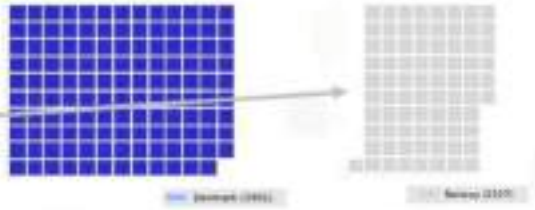
Country	Total Immigrants
Denmark	3901
Norway	2327
Sweden	5866



Waffle Charts

- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

Country	Total Immigrants
Denmark	3901
Norway	2327
Sweden	5866

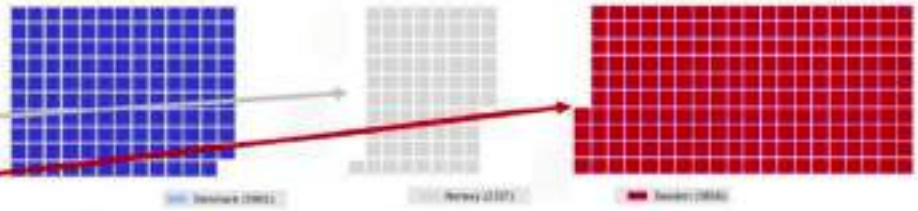


COGNITIVE CLASS.ai

Waffle Charts

- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

Country	Total Immigrants
Denmark	3901
Norway	2327
Sweden	5866

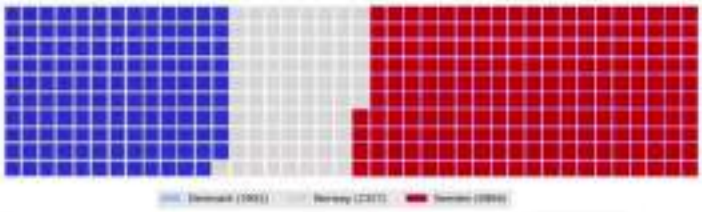


COGNITIVE CLASS.ai

Waffle Charts

- A waffle chart is an interesting visualization that is normally created to display progress toward goals.

Country	Total Immigrants
Denmark	3901
Norway	2327
Sweden	5866

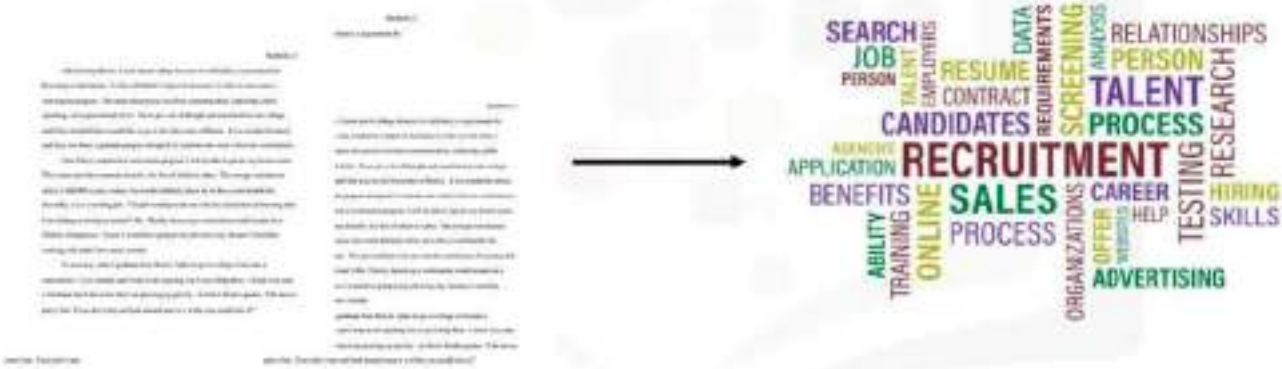


Data Visualization with Python

Word Clouds

Word Clouds

- A Word cloud is a depiction of the frequency of different words in some textual data.



© 2017 IBM Corporation

Data Visualization with Python

Seaborn and Regression Plots

Seaborn

- Seaborn is a Python visualization library based on Matplotlib.
- Visuals that need ~20 lines of code using Matplotlib to be created, with seaborn, the number of lines of code is reduced by 5-fold.
- Let's look at creating scatter plots with regression lines for example.

Regression Plots

df_total

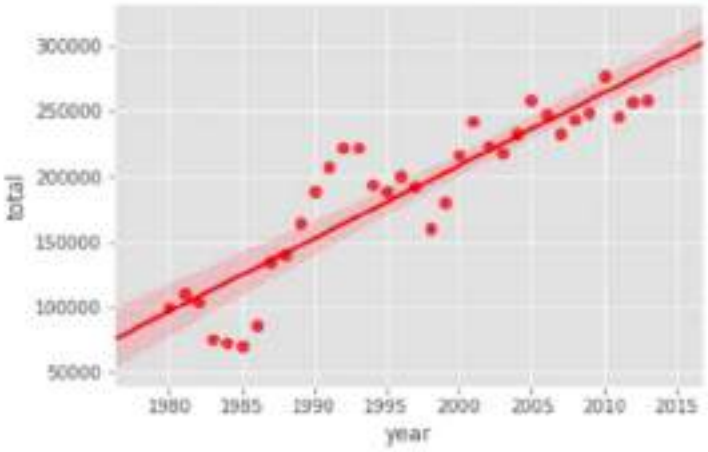
year	total
1980	99137
1981	110563
1982	104271
1983	75550
1984	73417
.	.
.	.

Regression Plots

df_total

year	total
1980	99137
1981	110563
1982	104271
1983	75550
1984	73417
.	.
.	.

```
import seaborn as sns
ax = sns.regplot(x='year', y='total', data=df_tot)
```



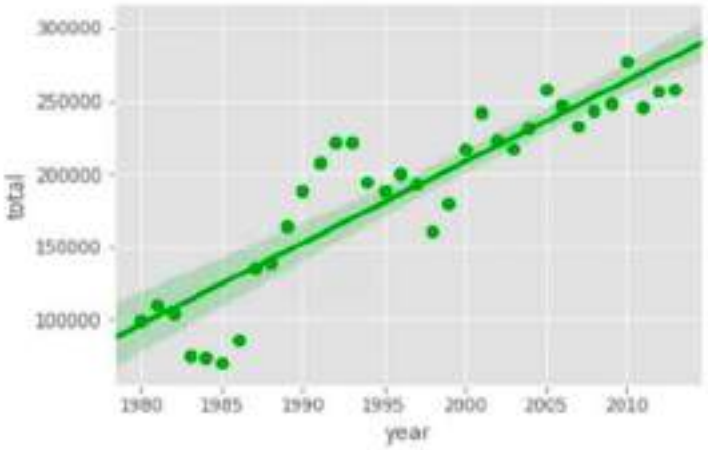
COGNITIVE CLASS.ai

Regression Plots

df_total

year	total
1980	99137
1981	110563
1982	104271
1983	75550
1984	73417
.	.
.	.

```
import seaborn as sns
ax = sns.regplot(x='year', y='total', data=df_tot,
                 color='green')
```



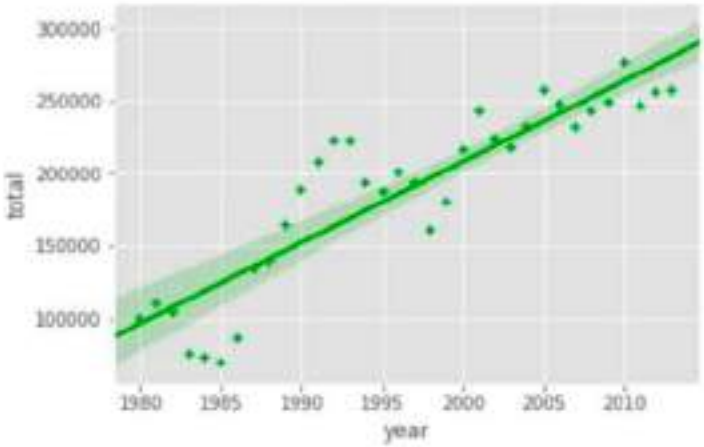
COGNITIVE CLASS.ai

Regression Plots

df_total

year	total
1980	99137
1981	110563
1982	104271
1983	75550
1984	73417
.	.
.	.

```
import seaborn as sns
ax = sns.regplot(x='year', y='total', data=df_tot,
                 color='green', marker='+')
```



COGNITIVE CLASS.ai

Data Visualization with Python

Introduction to Folium

COGNITIVE CLASS.ai

What is Folium?

- Folium is a powerful Python library that helps you create several types of Leaflet maps.
- It enables both the binding of data to a map for choropleth visualizations as well as passing visualizations as markers on the map.
- The library has a number of built-in tilesets from OpenStreetMap, Mapbox, and Stamen, and supports custom tilesets with Mapbox API keys.

COGNITIVE CLASS.ai

Creating a World Map

```
# define the world map
world_map = folium.Map()

# display world map
world_map
```

COGNITIVE CLASS.ai

Creating a Map of Canada

```
# define the world map centered around
# Canada with a low zoom level
world_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)

# display world map
world_map
```

COGNITIVE CLASS.ai

Map Styles – Stamen Toner

```
# create a Stamen Toner map of
# the world centered around Canada
world_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4,
    tiles='Stamen Toner'
)

# display map
world_map
```

COGNITIVE CLASS.ai

Map Styles – Stamen Terrain

```
# create a Stamen Toner map of
# the world centered around Canada
world_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4,
    tiles='Stamen Terrain'
)

# display map
world_map
```

Data Visualization with Python

Maps with Markers

Add a Marker

```
# generate map of Canada
canada_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)
```

```
# display map
canada_map
```

COGNITIVE CLASS.ai

Add a Marker

```
# generate map of Canada
canada_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)
```

```
# display map
canada_map
```

COGNITIVE CLASS.ai

Add a Marker

```
# generate map of Canada
canada_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)

## add a red marker to Ontario

# create a feature group
ontario = folium.map.FeatureGroup()

# style the feature group
ontario.add_child(
    folium.features.CircleMarker(
    [51.25, -85.32], radius = 5,
    color = "red", fill_color = "Red"
    )
)
```

```
# display map
canada_map
```

COGNITIVE CLASS.ai

Add a Marker

```
# generate map of Canada
canada_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)

## add a red marker to Ontario

# create a feature group
ontario = folium.map.FeatureGroup()

# style the feature group
ontario.add_child(
    folium.features.CircleMarker(
    [51.25, -85.32], radius = 5,
    color = "red", fill_color = "Red"
    )
)

# add the feature group to the map
canada_map.add_child(ontario)
```

```
# display map
canada_map
```

COGNITIVE CLASS.ai

Label the Marker

```
# generate map of Canada
canada_map = folium.Map(
    location=[56.130, -106.35],
    zoom_start=4
)

## add a red marker to Ontario

# create a feature group
ontario = folium.map.FeatureGroup()

# style the feature group
ontario.add_child(
    folium.features.CircleMarker(
    [51.25, -85.32], radius = 5,
    color = "red", fill_color = "Red"
    )
)

# add the feature group to the map
canada_map.add_child(ontario)

# label the marker
folium.Marker([51.25, -85.32],
    popup='Ontario').add_to(canada_map)

# display map
canada_map
```

Data Visualization with Python

Choropleth Maps

Choropleth Maps

COGNITIVE CLASS.ai

Geojson File

```
{
  "type": "FeatureCollection",
  "features": [{
    "type": "Feature",
    "properties": {
      "name": "Brunei"
    },
    "geometry": {
      "type": "Polygon",
      "coordinates": [
        [
          [114.204017, 4.525874], [114.599961, 4.900011],[115.45871, 5.44773],
          [115.4057, 4.955228],[115.347461, 4.3166636],[114.869557, 4.348314],
          [114.659596, 4.007637], [114.204017, 4.525874]
        ]
      ]
    },
    "id": "BRN"
  },
```

COGNITIVE CLASS.ai

Creating the Map

Creating the Map

```
# create a plain world map
world_map = folium.Map(
    zoom_start=2,
    tiles='Mapbox Bright'
)
```

```
# display map
world_map
```

Creating the Map

```
# create a plain world map
world_map = folium.Map(
    zoom_start=2,
    tiles='Mapbox Bright'
)

## geojson file
world_geo = r'world_countries.json'

# generate choropleth map using the total
# population of each country to Canada from
# 1980 to 2013
world_map.choropleth(
    geo_path=world_geo,
    data=df_canada,
    columns=['Country', 'Total'],
    key_on='feature.properties.name',
    fill_color='YlOrRd',
    legend_name='Immigration to Canada'
)

# display map
world_map
```

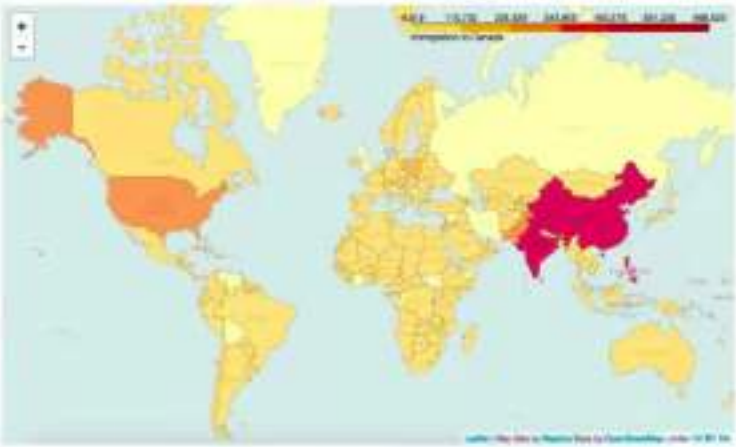
Creating the Map

```
# create a plain world map
world_map = folium.Map(
    zoom_start=2,
    tiles='Mapbox Bright'
)

## geojson file
world_geo = r'world_countries.json'

# generate choropleth map using the total
# population of each country to Canada from
# 1980 to 2013
world_map.choropleth(
    geo_path=world_geo,
    data=df_canada,
    columns=['Country', 'Total'],
    key_on='feature.properties.name',
    fill_color='YlOrRd',
    legend_name='Immigration to Canada'
)

# display map
world_map
```

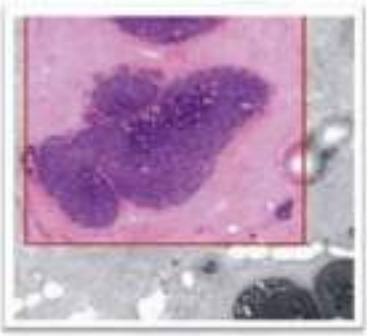


Intro to Machine Learning

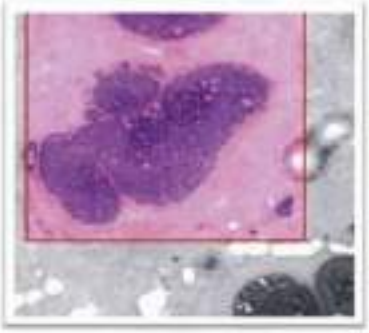
Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Is this a benign or malignant cell?



Is this a benign or malignant cell?

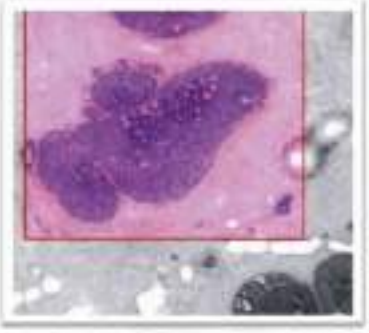


ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	7	1	3	1	1	

COGNITIVE CLASS

2

Is this a benign or malignant cell?



ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	7	1	3	1	1	

Benign or Malignant ?

COGNITIVE CLASS

2

Machine learning helps with predictions!

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

COGNITIVE CLASS

Machine learning helps with predictions!

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	7	1	3	1	1	

COGNITIVE CLASS

Machine learning helps with predictions!

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Modeling

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	7	1	3	1	1	

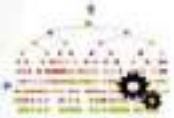
Machine learning helps with predictions!

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Modeling

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	7	1	3	1	1	

Prediction



Machine learning helps with predictions!

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	1	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Modeling

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000015	6	1	1	1	2	1	3	1	1	Benign

Prediction

Accuracy = 89%

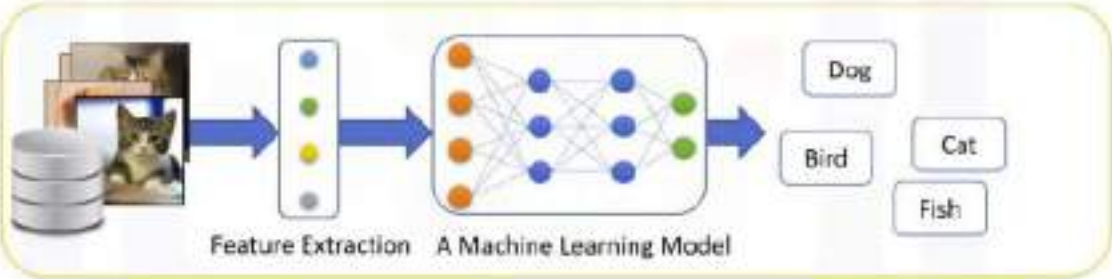
What is machine learning?

Machine learning is the subfield of computer science that gives "**computers the ability to learn without being explicitly programmed.**"

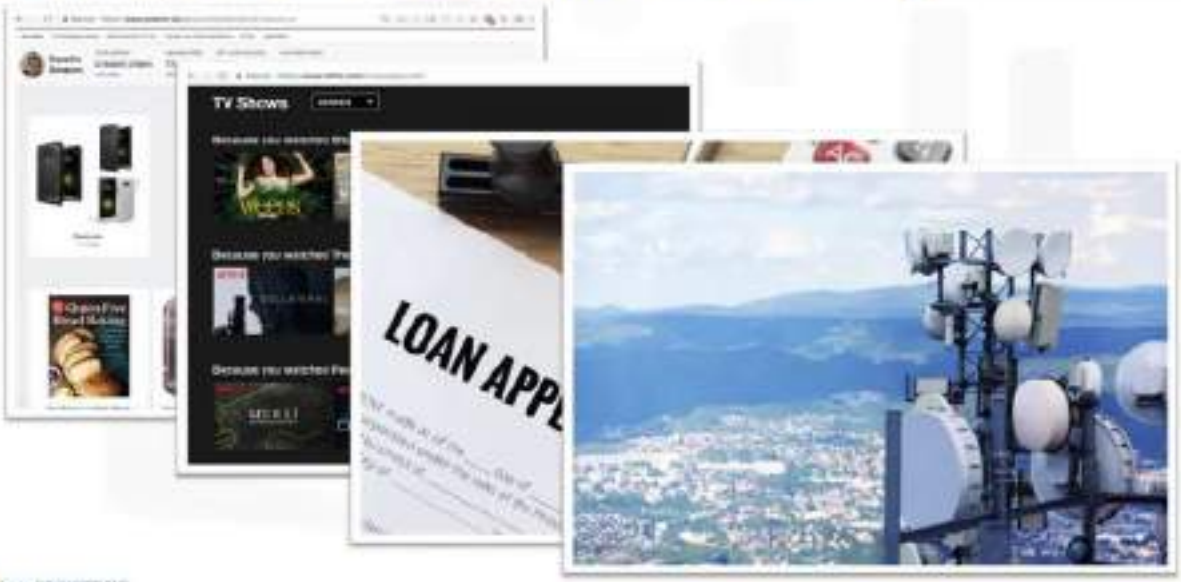
Arthur Samuel
American pioneer in the field of computer gaming and artificial intelligence, coined the term "machine learning" in 1959 while at IBM.

How machine learning works?

How machine learning works?



Examples of machine learning



… such as predicting the right drug

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY

… such as predicting the right drug

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL			
4	61	F	LOW			

Use ML to make decisions

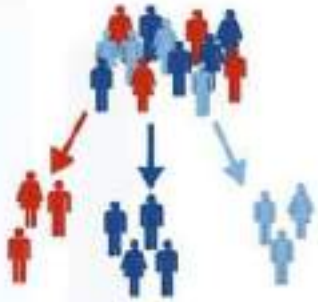
age	ed	employ	address	income	debtinc	creddebt	othdebt	default
41	3	17	12	176	9.3	11.159	5.009	1
27	1	10	6	31	17.3	1.362	4.001	0
40	1	15	14	55	5.5	0.856	2.169	0
41	1	15	14	120	2.9	2.659	0.821	0
24	2	2	0	28	17.3	1.787	3.057	1
41	2	5	5	25	10.2	0.393	2.157	0
39	[illegible]	[illegible]	0	67	30.6	3.834	16.668	0
43	[illegible]	[illegible]	[illegible]	38	3.6	0.129	1.239	0
[illegible]	[illegible]	[illegible]	[illegible]	19	24.4	1.358	3.278	1
[illegible]	[illegible]	[illegible]	[illegible]	25	19.7	2.778	2.147	0

Categorical Variable

[illegible]ome	debtinc	creddebt	othdebt	default
[illegible]30	9.3	10.23	3.21	

Use ML for customer segmentation

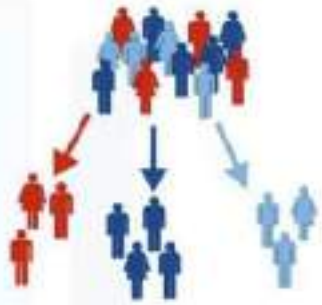
Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio	Defaulted
1	41	2	6	19	0.124	1.073	NBA001	6.3	0
2	47	1	26	100	4.582	8.218	NBA021	12.8	0
3	33	2	10	57	6.111	5.802	NBA013	20.9	1
4	29	2	4	19	0.681	0.516	NBA009	6.3	0
5	47	1	31	253	9.308	8.908	NBA008	7.2	0
6	40	1	23	81	0.998	7.831	NBA016	10.9	1
7	38	2	4	56	0.442	0.454	NBA013	1.6	0
8	42	3	0	64	0.279	3.945	NBA009	6.6	0
9	26	1	5	18	0.575	2.215	NBA006	15.5	1

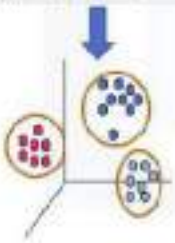


COGNITIVE CLASS

Use ML for customer segmentation

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio	Defaulted
1	41	2	6	19	0.124	1.073	NBA001	6.3	0
2	47	1	26	100	4.582	8.218	NBA021	12.8	0
3	33	2	10	57	6.111	5.802	NBA013	20.9	1
4	29	2	4	19	0.681	0.516	NBA009	6.3	0
5	47	1	31	253	9.308	8.908	NBA008	7.2	0
6	40	1	23	81	0.998	7.831	NBA016	10.9	1
7	38	2	4	56	0.442	0.454	NBA013	1.6	0
8	42	3	0	64	0.279	3.945	NBA009	6.6	0
9	26	1	5	18	0.575	2.215	NBA006	15.5	1

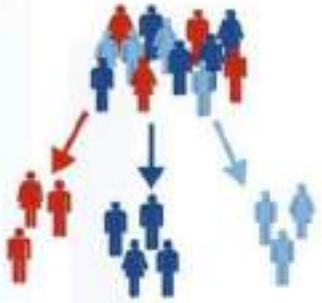


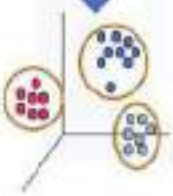


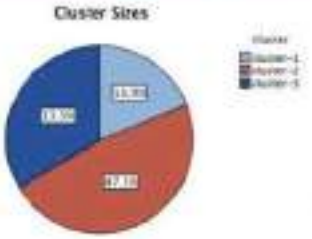
COGNITIVE CLASS

Use ML for customer segmentation

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio	Defaulted
1	41	2	6	19	0.124	1.073	NBA001	6.3	0
2	47	1	26	100	4.582	8.218	NBA021	12.8	0
3	33	2	10	57	6.111	5.802	NBA013	20.9	1
4	29	2	4	19	0.681	0.516	NBA009	6.3	0
5	47	1	31	253	9.308	8.908	NBA008	7.2	0
6	40	1	23	81	0.998	7.831	NBA016	10.9	1
7	38	2	4	56	0.442	0.454	NBA013	1.6	0
8	42	3	0	64	0.279	3.945	NBA009	6.6	0
9	26	1	5	18	0.575	2.215	NBA006	15.5	1



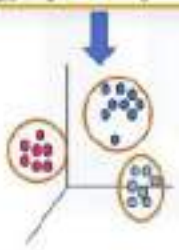


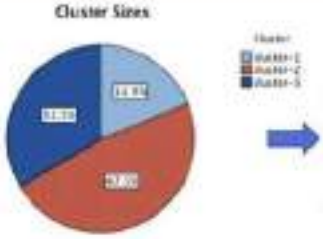


COGNITIVE CLASS

Use ML for customer segmentation

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio	Defaulted
1	41	2	6	19	0.124	1.073	NBA001	6.3	0
2	47	1	26	100	4.582	8.218	NBA021	12.8	0
3	33	2	10	57	6.111	5.802	NBA013	20.9	1
4	29	2	4	19	0.681	0.516	NBA009	6.3	0
5	47	1	31	253	9.308	8.908	NBA008	7.2	0
6	40	1	23	81	0.998	7.831	NBA016	10.9	1
7	38	2	4	56	0.442	0.454	NBA013	1.6	0
8	42	3	0	64	0.279	3.945	NBA009	6.6	0
9	26	1	5	18	0.575	2.215	NBA006	15.5	1





Cluster	Segment Name
cluster-1	AFFLUENT AND MIDDLE AGED
cluster-2	YOUNG EDUCATED AND MIDDLE INCOME
cluster-3	YOUNG AND LOW INCOME

COGNITIVE CLASS

Use ML for recommendation systems

Major machine learning techniques

- Regression/Estimation
  - Predicting continuous values
- Classification
  - Predicting the item class/category of a case
- Clustering
  - Finding the structure of data; summarization
- Associations
  - Associating frequent co-occurring items/events

Major machine learning techniques

- Anomaly detection
  - Discovering abnormal and unusual cases
- Sequence mining
  - Predicting next events; click-stream (Markov Model, HMM)
- Dimension Reduction
  - Reducing the size of data (PCA)
- Recommendation systems
  - Recommending items

Difference between artificial intelligence, machine learning, and deep learning

- AI components:
  - Computer Vision
  - Language Processing
  - Creativity
  - Etc.
- Machine learning:
  - Classification
  - Clustering
  - Neural Network
  - Etc.
- Revolution in ML:
  - Deep learning

Python for Machine Learning

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Python libraries for machine learning

pandas

scikit learn

NumPy

SciPy

matplotlib

python

More about scikit-learn

- Free software machine learning library
- Classification, Regression and Clustering algorithms
- Works with NumPy and SciPy
- Great documentation
- Easy to implement

scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```


scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```

```
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100.)
```

scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```

```
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100.)
```

```
clf.fit(X_train, y_train)
```


scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```

```
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100.)
```

```
clf.fit(X_train, y_train)
```

```
clf.predict(X_test)
```

scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```

```
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100.)
```

```
clf.fit(X_train, y_train)
```

```
clf.predict(X_test)
```

```
from sklearn.metrics import confusion_matrix
print(confusion_matrix(y_test, yhat, labels=[1,0]))
```


scikit-learn functions

```
from sklearn import preprocessing
X = preprocessing.StandardScaler().fit(X).transform(X)
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
```

```
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100.)
```

```
clf.fit(X_train, y_train)
```

```
clf.predict(X_test)
```

```
from sklearn.metrics import confusion_matrix
print(confusion_matrix(y_test, yhat, labels=[1,0]))
```

```
import pickle
s = pickle.dumps(clf)
```

Supervised vs Unsupervised

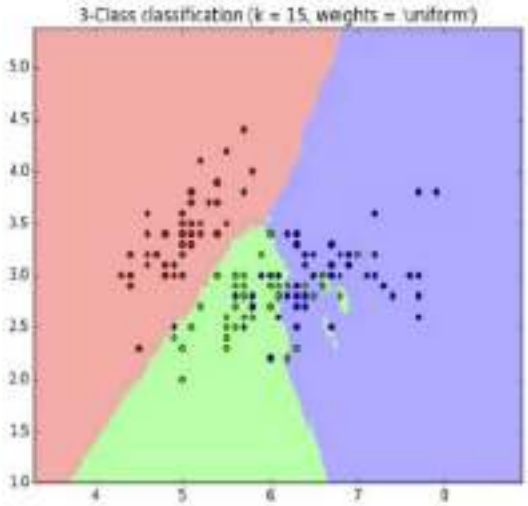
Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

What is supervised learning?

What is supervised learning?

What is supervised learning?



We "teach the model," then with that knowledge, it can predict unknown or future instances.

Teaching the model with labeled data

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Teaching the model with labeled data

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

COGNITIVE CLASS

3

Teaching the model with labeled data

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

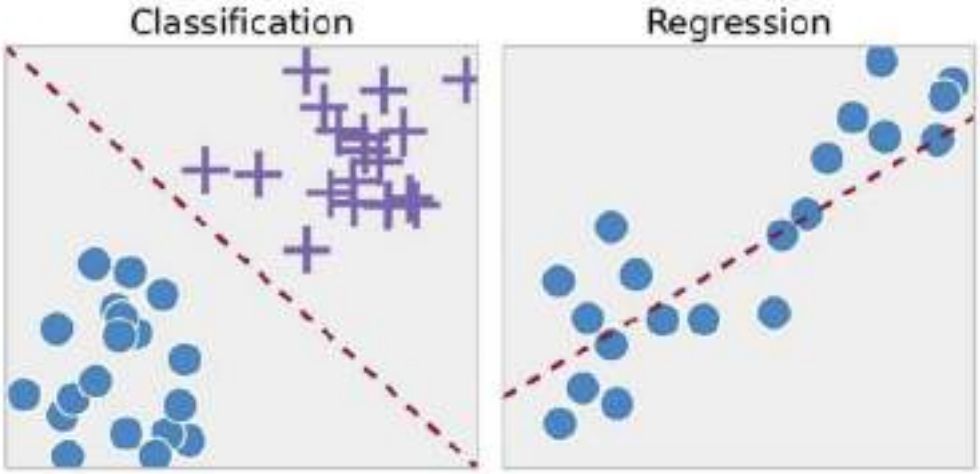
Teaching the model with labeled data

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Teaching the model with labeled data

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

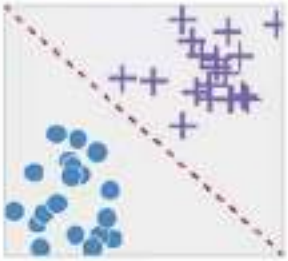
Types of supervised learning



What is classification?

Classification is the process of predicting discrete class labels or categories.

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	3	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

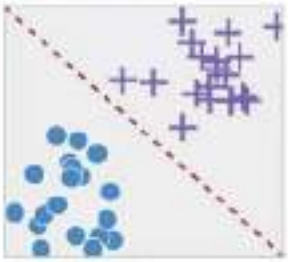


What is classification?

Classification is the process of predicting discrete class labels or categories.

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	3	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Categorical Values

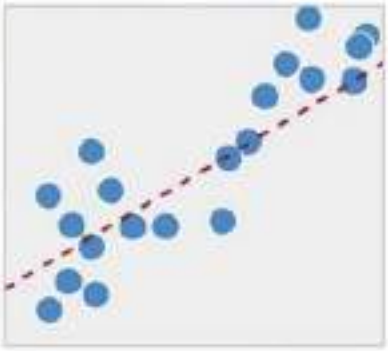


What is regression?

Regression is the process of predicting continuous values.

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Continuous Values



What is unsupervised learning?

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio
1	41	2	6	19	0.124	1.073	NBA001	6.3
2	47	1	26	100	4.582	8.218	NBA021	12.8
3	33	2	10	57	6.111	5.802	NBA013	20.9
4	29	2	4	19	0.681	0.516	NBA009	6.3
5	47	1	31	253	9.308	8.908	NBA008	7.2
6	40	1	23	81	0.998	7.831	NBA016	10.9
7	38	2	4	56	0.442	0.454	NBA013	1.6
8	42	3	0	64	0.279	3.545	NBA009	6.6
9	26	1	6	18	0.575	2.215	NBA006	15.5
10	47	3	23	115	0.653	3.947	NBA011	4
11	44	3	8	88	0.285	5.083	NBA010	6.1
12	34	2	9	40	0.374	0.266	NBA003	1.6

What is unsupervised learning?

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio
1	41	2	6	19	0.124	1.073	NBA001	6.3
2	47	1	26	100	4.582	8.218	NBA021	12.8
3	33	2	10	57	6.111	5.802	NBA013	20.9
4	29	2	4	19	0.681	0.516	NBA009	6.3
5	47	1	31	253	9.308	8.908	NBA008	7.2
6	40	1	23	81	0.998	7.831	NBA016	10.9
7	38	2	4	56	0.442	0.454	NBA013	1.6
8	42	3	0	64	0.279	3.945	NBA009	6.6
9	26	1	5	18	0.575	2.215	NBA006	15.5
10	47	3	23	115	0.653	3.947	NBA011	4
11	44	3	8	88	0.285	5.083	NBA010	6.1
12	34	2	9	40	0.374	0.266	NBA003	1.6

The model works on its own to discover information.

COGNITIVE CLASS

What is unsupervised learning?

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio
1	41	2	6	19	0.124	1.073	NBA001	6.3
2	47	1	26	100	4.582	8.218	NBA021	12.8
3	33	2	10	57	6.111	5.802	NBA013	20.9
4	29	2	4	19	0.681	0.516	NBA009	6.3
5	47	1	31	253	9.308	8.908	NBA008	7.2
6	40	1	23	81	0.998	7.831	NBA016	10.9
7	38	2	4	56	0.442	0.454	NBA013	1.6
8	42	3	0	64	0.279	3.945	NBA009	6.6
9	26	1	5	18	0.575	2.215	NBA006	15.5
10	47	3	23	115	0.653	3.947	NBA011	4
11	44	3	8	88	0.285	5.083	NBA010	6.1
12	34	2	9	40	0.374	0.266	NBA003	1.6

ALL OF THIS DATA IS UNLABELED

The model works on its own to discover information.

COGNITIVE CLASS

What is unsupervised learning?

Customer Id	Age	Edu	Years Employed	Income	Card Debt	Other Debt	Address	DebtIncomeRatio
1	41	2	6	19	0.124	1.073	NBA001	6.3
2	47	1	26	100	4.582	8.218	NBA021	12.8
3	33	2	10	57	6.111	5.802	NBA013	20.9
4	29	2	4	19	0.681	0.516	NBA009	6.3
5	47	1	31	253	9.308	8.908	NBA008	7.2
6	40	1	23	81	0.998	7.831	NBA016	10.9
7	38	2	4	56	0.442	0.454	NBA013	1.6
8	42	3	0	64	0.279	3.945	NBA009	6.6
9	26	1	5	18	0.575	2.215	NBA006	15.5
10	47	3	23	115	0.653	3.947	NBA011	4
11	44	3	8	88	0.285	5.083	NBA010	6.1
12	34	2	9	40	0.374	0.266	NBA003	1.6

Unsupervised learning techniques:

- Dimension reduction
- Density estimation
- Market basket analysis
- Clustering

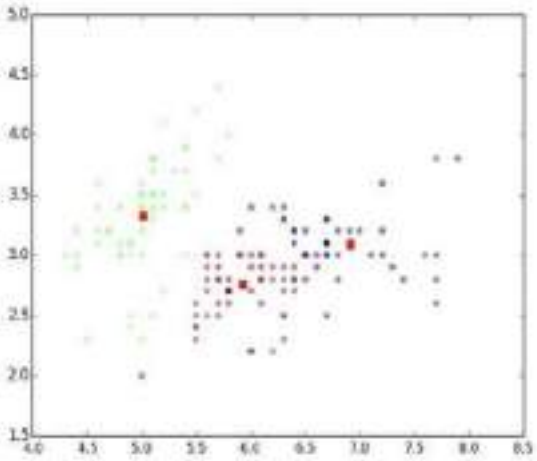
ALL OF THIS DATA IS UNLABELED

The model works on its own to discover information.

What is clustering?

Clustering is grouping of data points or objects that are somehow similar by:

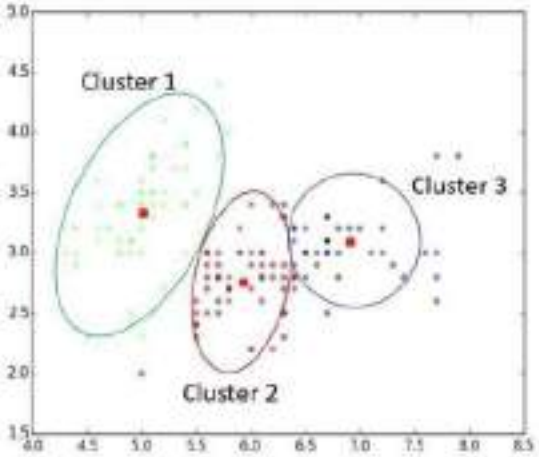
- Discovering structure
- Summarization
- Anomaly detection



What is clustering?

Clustering is grouping of data points or objects that are somehow similar by:

- Discovering structure
- Summarization
- Anomaly detection



Supervised vs unsupervised learning

Supervised Learning

- **Classification:**
  Classifies labeled data
- **Regression:**
  Predicts trends using previous labeled data
- Has more evaluation methods than unsupervised learning
- Controlled environment

Unsupervised Learning

- **Clustering:**
  Finds patterns and groupings from unlabeled data
- Has fewer evaluation methods than supervised learning
- Less controlled environment

Intro to Regression

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

What is regression?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

What is regression?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Regression is the process of predicting a continuous value

What is regression?

X: Independent variable

Y: Dependent variable

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Regression is the process of predicting a continuous value

COGNITIVE CLASS

2

What is regression?

X: Independent variable

Y: Dependent variable

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

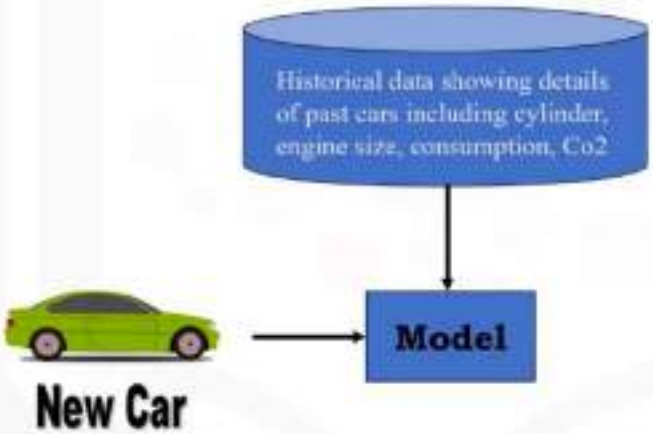
Continuous Values

Regression is the process of predicting a continuous value

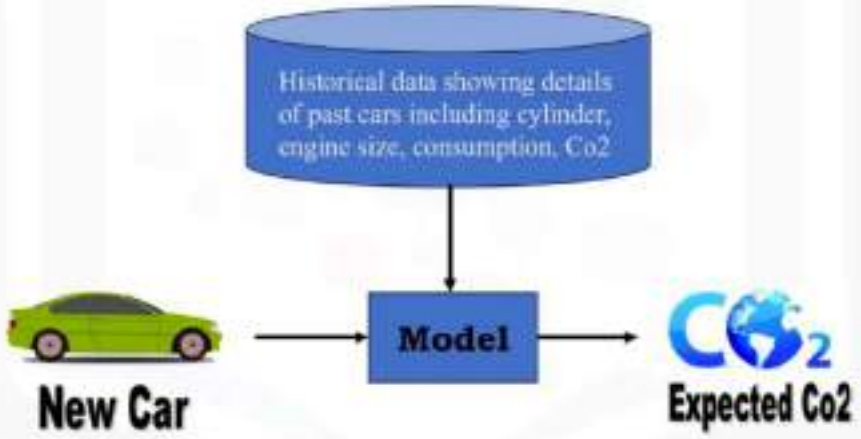
What is a regression model?

What is a regression model?

What is a regression model?



What is a regression model?



Types of regression models

- Simple Regression:
- Multiple Regression:

Types of regression models

- Simple Regression:
  - Simple Linear Regression
  - Simple Non-linear Regression
- Multiple Regression:
  - Multiple Linear Regression
  - Multiple Non-linear Regression

Predict co2emission vs EngineSize of all cars

Predict co2emission vs EngineSize and Cylinders of all cars

Applications of regression

- Sales forecasting
- Satisfaction analysis
- Price estimation
- Employment income

Regression algorithms

- Ordinal regression
- Poisson regression
- Fast forest quantile regression
- Linear, Polynomial, Lasso, Stepwise, Ridge regression
- Bayesian linear regression
- Neural network regression
- Decision forest regression
- Boosted decision tree regression
- KNN (K-nearest neighbors)

Simple Linear Regression

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Using linear regression to predict continuous values

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Using linear regression to predict continuous values

X: Independent variable

Y: Dependent variable

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Continuous Values

Linear regression topology

- Simple Linear Regression:

- Multiple Linear Regression:

Linear regression topology

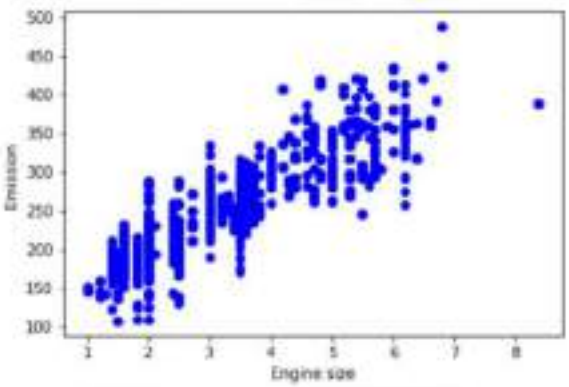
- Simple Linear Regression:
  - Predict co2emission vs EngineSize of all cars
    - Independent variable (x): EngineSize
    - Dependent variable (y): co2emission
- Multiple Linear Regression:
  - Predict co2emission vs EngineSize and Cylinders of all cars
    - Independent variable (x): EngineSize, Cylinders, etc
    - Dependent variable (y): co2emission

How does linear regression work?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

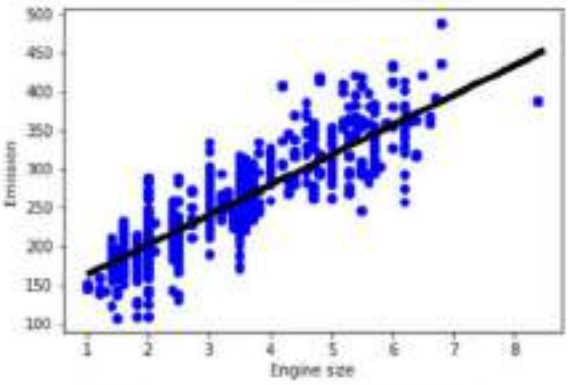
How does linear regression work?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

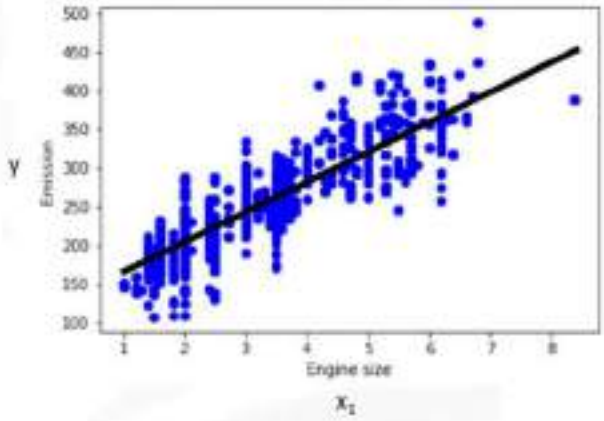


How does linear regression work?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

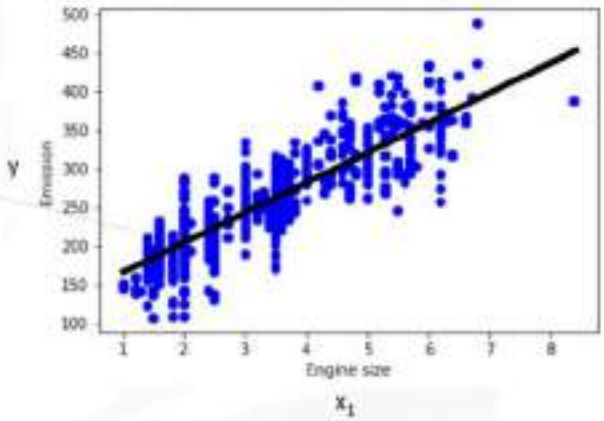


Linear regression model representation



Linear regression model representation

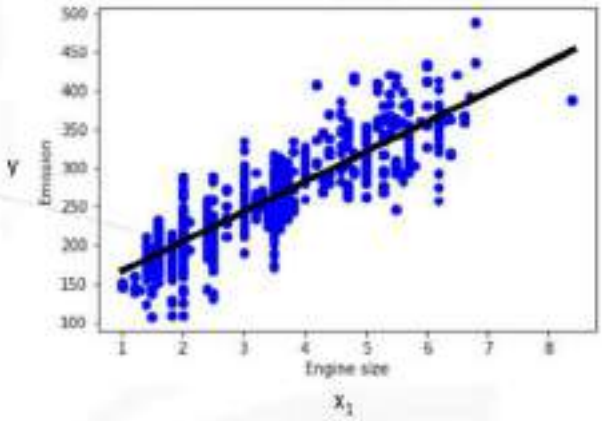
$$\hat{y} = \theta_0 + \theta_1 x_1$$



Linear regression model representation

$$\hat{y} = \theta_0 + \theta_1 x_1$$

response variable

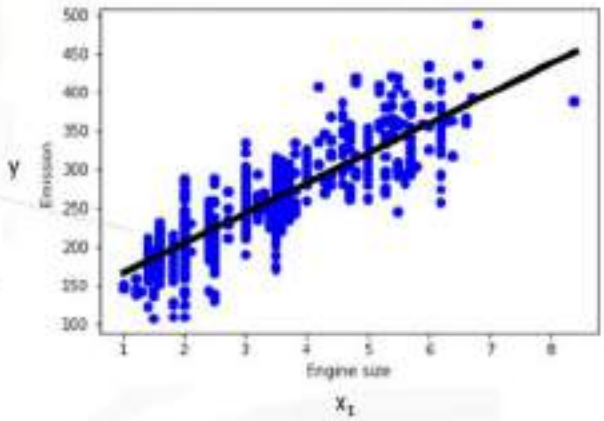


Linear regression model representation

$$\hat{y} = \theta_0 + \theta_1 x_1$$

a single predictor

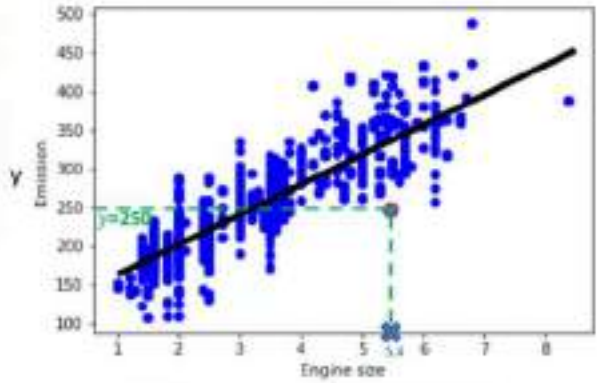
response variable



How to find the best fit?

x_1 = 2.4 independent variable

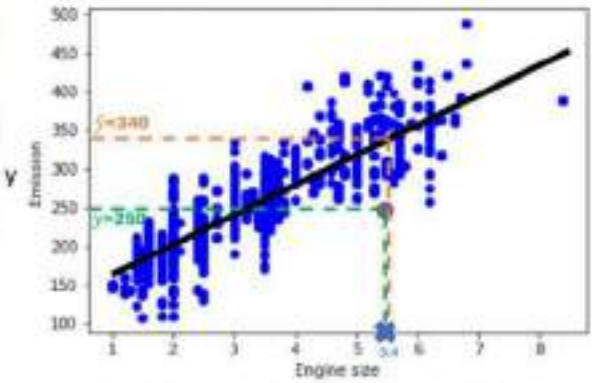
y= 250 actual Co2 emission of x1



How to find the best fit?

$x_1 = 2.4$ independent variable
y= 250 actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y}$ = 340 the predicted emission of x1

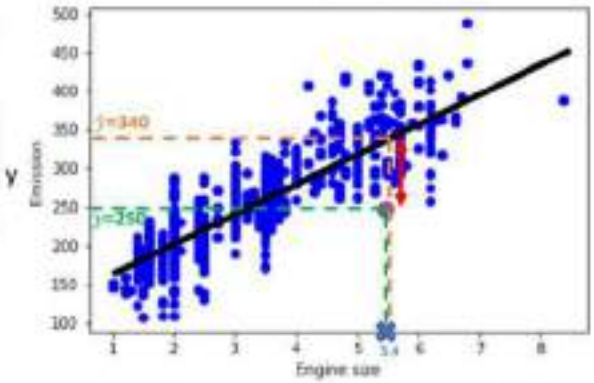


How to find the best fit?

$x_1 = 2.4$ independent variable
y= 250 actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y}$ = 340 the predicted emission of x1

Error = y- $\hat{y}$
= 250 − 340
= -90

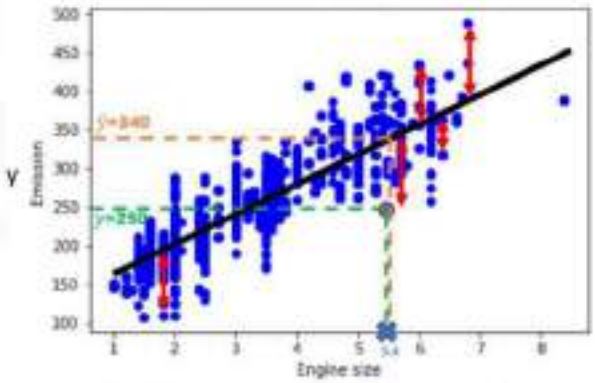


How to find the best fit?

$x_1 = 2.4$ independent variable
$y = 250$ actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y} = 340$ the predicted emission of x1

Error = $y - \hat{y}$
$= 250 - 340$
$= -90$



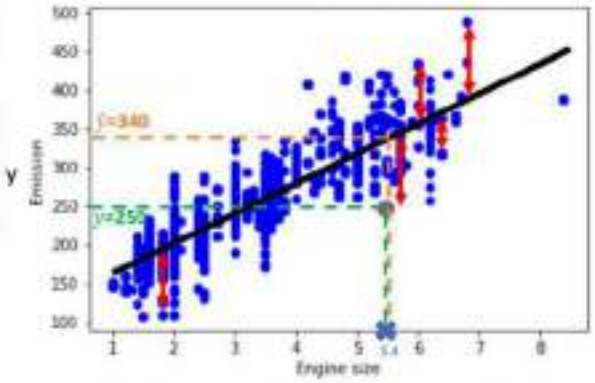
How to find the best fit?

$x_1 = 2.4$ independent variable
$y = 250$ actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y} = 340$ the predicted emission of x1

Error = $y - \hat{y}$
$= 250 - 340$
$= -90$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$



Estimating the parameters

$$\hat{y} = \theta_0 + \theta_1 x_1$$

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

Estimating the parameters

$$\hat{y} = \theta_0 + \theta_1 x_1$$

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimating the parameters

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

COGNITIVE CLASS

Estimating the parameters

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$\theta_1 = 39$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

COGNITIVE CLASS

Estimating the parameters

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$\theta_1 = 39$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$\theta_0 = 256 - 39 * 3.34$

$\theta_0 = 125.74$

COGNITIVE CLASS

Estimating the parameters

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$\theta_1 = 39$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$\theta_0 = 256 - 39 * 3.34$

$\theta_0 = 125.74$

$$\boxed{\hat{y} = 125.74 + 39x_1}$$

COGNITIVE CLASS

Predictions with linear regression

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$Co2Emission = \theta_0 + \theta_1\ EngineSize$$

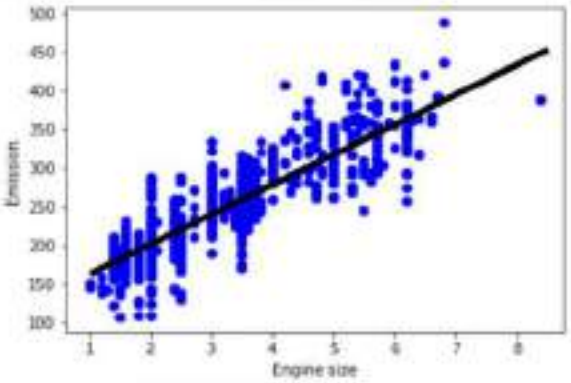
$$Co2Emission = 125 + 39\ EngineSize$$

$$Co2Emission = 125 + 39 \times 2.4$$

$$Co2Emission = 218.6$$

Pros of linear regression

- Very fast
- No parameter tuning
- Easy to understand, and highly interpretable



Model Evaluation in Regression Models

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Model evaluation approaches

Model evaluation approaches

- Train and Test on the Same Dataset
- Train/Test Split

Regression Evaluation Metrics

Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.6	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Best approach for most accurate results?

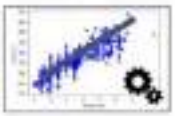
	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

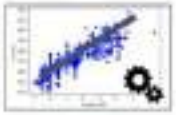


Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



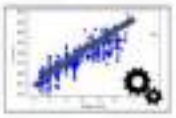
Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test

Actual values

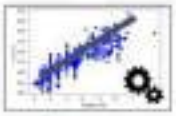


Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



Actual values

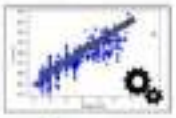
COGNITIVE CLASS

Best approach for most accurate results?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



	Prediction
6	234
7	256
8	267
9	210

Actual values

Predicted values

COGNITIVE CLASS

Calculating the accuracy of a model

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Test

Actual values

$$Error = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

	Prediction
6	234
7	256
8	267
9	210

Predicted values

COGNITIVE CLASS

4

Calculating the accuracy of a model

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Test

y

Actual values

$$Error = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

$\hat{y}$

	Prediction
6	234
7	256
8	267
9	210

Predicted values

COGNITIVE CLASS

4

Calculating the accuracy of a model

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Test

y

Actual values

$$Error = \frac{(232 - 234) + (255 - 256) + \ ...}{4}$$

$$Error = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

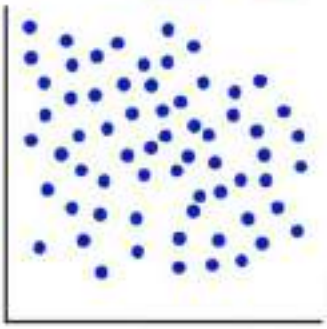
$\hat{y}$

	Prediction
6	234
7	256
8	267
9	210

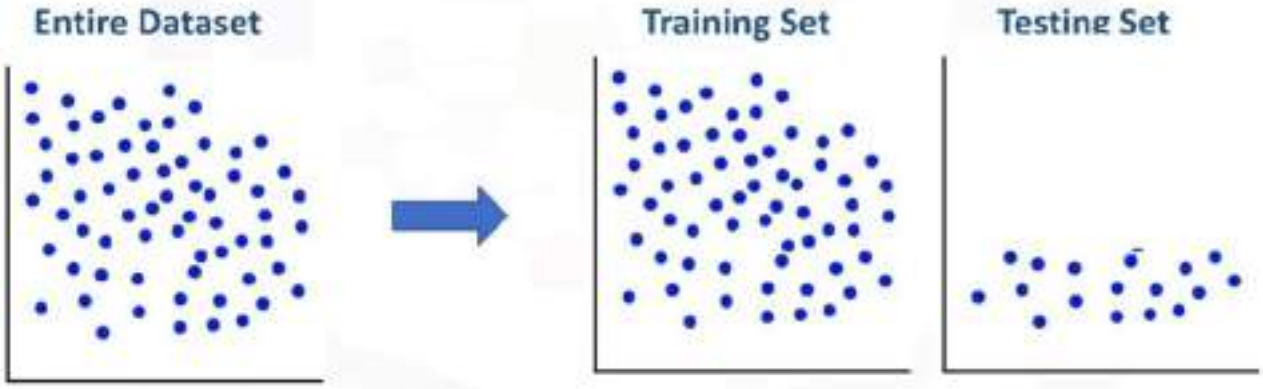
Predicted values

Train and test on the same dataset

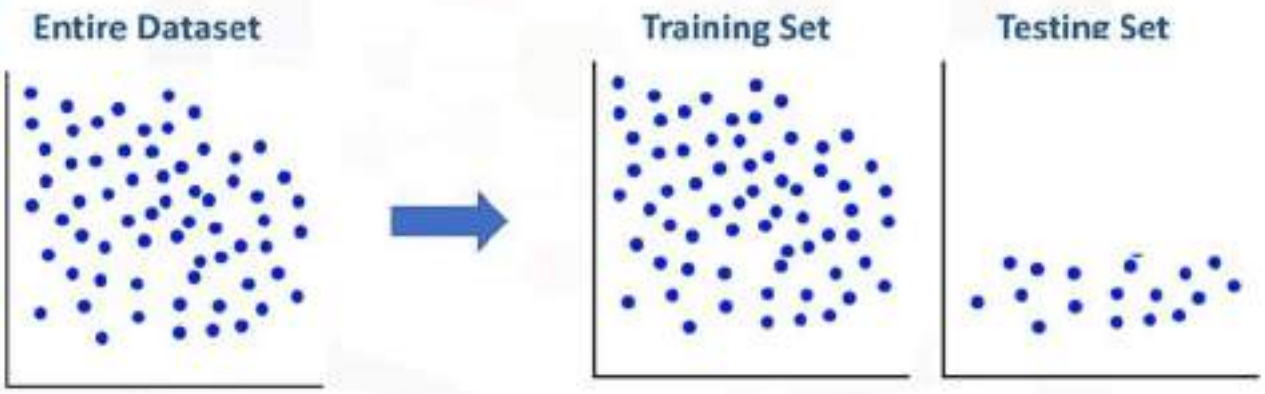
Entire Dataset



Train and test on the same dataset



Train and test on the same dataset



High "training accuracy"
Low "out-of-sample accuracy"

What is training & out-of-sample accuracy?

- **Training Accuracy**
  - High training accuracy isn't necessarily a good thing
  - Result of over-fitting
    - Over-fit: the model is overly trained to the dataset, which may capture noise and produce a non-generalized model
- **Out-of-Sample Accuracy**
  - It's important that our models have a high, out-of-sample accuracy
  - How can we improve out-of-sample accuracy?

Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

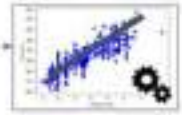
Train

Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS	
0	2.0	4	8.5	196	Train
1	2.4	4	9.6	221	
2	1.5	4	5.9	136	
3	3.5	6	11.1	255	
4	3.5	6	10.6	244	
5	3.5	6	10.0	230	
6	3.5	6	10.1	232	Test
7	3.7	6	11.1	255	
8	3.7	6	11.6	267	
9	2.4	4	9.2	212	

Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS	
0	2.0	4	8.5	196	Train
1	2.4	4	9.6	221	
2	1.5	4	5.9	136	
3	3.5	6	11.1	255	
4	3.5	6	10.6	244	
5	3.5	6	10.0	230	
6	3.5	6	10.1	232	Test
7	3.7	6	11.1	255	
8	3.7	6	11.6	267	
9	2.4	4	9.2	212	



Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test

COGNITIVE CLASS

Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test

	Prediction
6	234
7	256
8	267
9	210

Predicted values

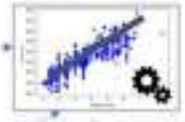
COGNITIVE CLASS

Train/Test split evaluation approach

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	212

Train

Test



	Prediction
6	234
7	256
8	267
9	210

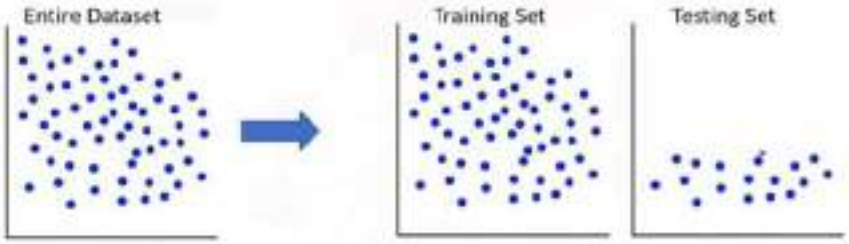
Actual values

Predicted values

COGNITIVE CLASS

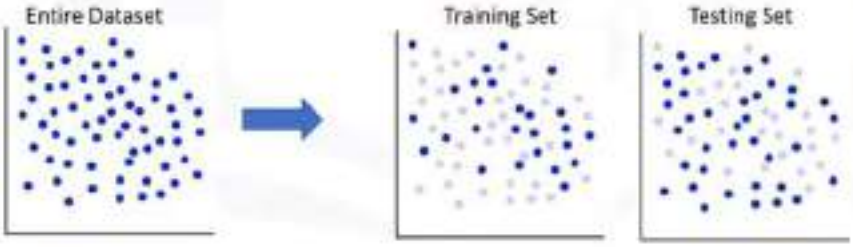
Train/Test split evaluation approach

Test on a portion of train set



- Test-set is a portion of the train-set
- High "training accuracy"
- Low "out-of-sample accuracy"

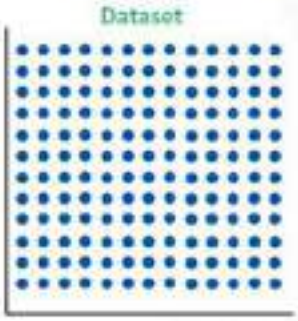
Train/Test Split



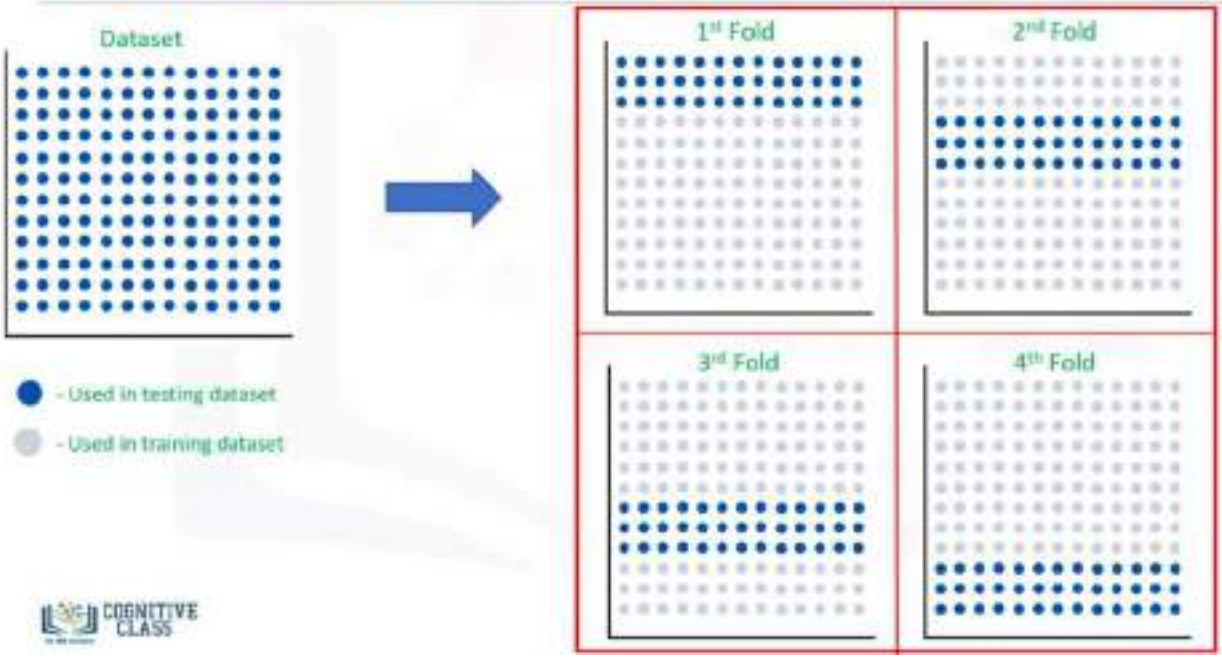
- Mutually exclusive
- More accurate evaluation on out-of-sample accuracy
- Highly dependent on which datasets the data is trained and tested

COGNITIVE CLASS

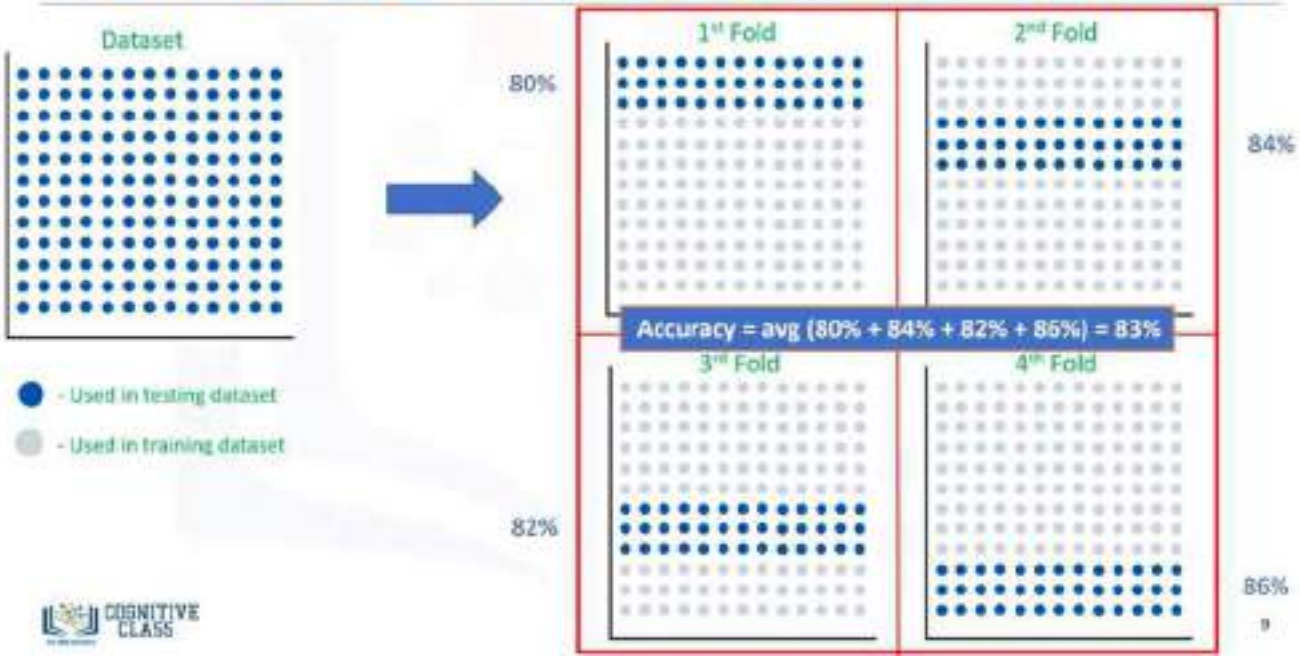
How to use K-fold cross-validation?



How to use K-fold cross-validation?



How to use K-fold cross-validation?



Multiple Linear Regression

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Types of regression models

- Simple Linear Regression
  - Predict Co2emission vs EngineSize of all cars
    - Independent variable (x): EngineSize
    - Dependent variable (y): Co2emission
- Multiple Linear Regression
  - Predict Co2emission vs EngineSize and Cylinders of all cars
    - Independent variable (x): EngineSize, Cylinders, etc.
    - Dependent variable (y): Co2emission

Examples of multiple linear regression

- Independent variables effectiveness on prediction
  - Does revision time, test anxiety, lecture attendance and gender have any effect on the exam performance of students?
- Predicting impacts of changes
  - How much does blood pressure go up (or down) for every unit increase (or decrease) in the BMI of a patient?

Predicting continuous values with multiple linear regression

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Predicting continuous values with multiple linear regression

$$Co2\ Em = \theta_0 + \theta_1 Engine\ size + \theta_2 Cylinders + \ ...$$

$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ ... + \theta_n x_n$$

$$\hat{y} = \theta^T X$$

$$\theta^T = [\theta_0, \theta_1, \theta_2, ...] \qquad X = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ ... \end{bmatrix}$$

X: Independent variable

Y: Dependent variable

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Using MSE to expose the errors in the model

$\hat{y} = \theta^T X$

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

Using MSE to expose the errors in the model

$\hat{y} = \theta^T X$

$\hat{y}_i = 140$ the predicted emission of x_i

$y_i = 196$ actual value of x_i

$y_i - \hat{y}_i = 196 - 140 = 56$ residual error

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

COGNITIVE CLASS

5

Estimating multiple linear regression parameters

- How to estimate θ?
  - Ordinary Least Squares
    - Linear algebra operations
    - Takes a long time for large datasets (10K+ rows)
  - An optimization algorithm
    - Gradient Descent
    - Proper approach if you have a very large dataset

Making predictions with multiple linear regression

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

$\hat{y} = \theta^T X$

$\theta^T = [125, 6.2, 14, \ldots]$

$\hat{y} = 125 + 6.2x_1 + 14x_2 +$

$Co2Em = 125 + 6.2EngSize + 14\ \text{Cylinders} + \ldots$

$Co2Em = 125 + 6.2 \times 2.4 + 14 \times 4 + \ldots$

$Co2Em = 214.1$

Intro to Classification

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

What is classification?

- A supervised learning approach
- Categorizing some unknown items into a discrete set of categories or "classes"
- The target attribute is a categorical variable

How does classification work?

Classification determines the class label for an unlabeled test case.

age	ed	employ	address	income	debtinc	creddebt	othdebt	default
41	3	17	12	176	9.3	11.359	5.009	1
27	1	10	6	31	17.3	1.362	4.001	0
40	1	15	14	55	5.5	0.856	2.169	0
41	1	15	14	120	2.9	2.659	0.821	0
24	2	2	0	28	17.3	1.787	3.057	1
41	2	5	5	25	10.2	0.393	2.157	0
39	1	20	9	67	30.6	3.834	16.668	0
43	1	12	11	38	3.6	0.129	1.239	0
24	1	3	4	19	24.4	1.358	3.278	1
36	1	0	13	25	19.7	2.778	2.147	0

Categorical Variable

age	ed	employ	address	income	debtinc	creddebt	othdebt	default
37	2	16	10	130	9.3	10.23	3.21	

COGNITIVE CLASS

How does classification work?

Classification determines the class label for an unlabeled test case.

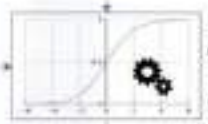
age	ed	employ	address	income	debtinc	creddebt	othdebt	default
41	3	17	12	176	9.3	11.359	5.009	1
27	1	10	6	31	17.3	1.362	4.001	0
40	1	15	14	55	5.5	0.856	2.169	0
41	1	15	14	120	2.9	2.659	0.821	0
24	2	2	0	28	17.3	1.787	3.057	1
41	2	5	5	25	10.2	0.393	2.157	0
39	1	20	9	67	30.6	3.834	16.668	0
43	1	12	11	38	3.6	0.129	1.239	0
24	1	3	4	19	24.4	1.358	3.278	1
36	1	0	13	25	19.7	2.778	2.147	0

Categorical Variable

Modeling

age	ed	employ	address	income	debtinc	creddebt	othdebt	default
37	2	16	10	130	9.3	10.23	3.21	0

Prediction



Classifier

Predicted Labels

COGNITIVE CLASS

Example of multi-class classification

Age	Sex	BP	Cholesterol	Na	K	Drug
23	F	HIGH	HIGH	0.793	0.031	drugY
47	M	LOW	HIGH	0.739	0.056	drugC
47	M	LOW	HIGH	0.697	0.069	drugC
28	F	NORMAL	HIGH	0.564	0.072	drugX
61	F	LOW	HIGH	0.559	0.031	drugY
22	F	NORMAL	HIGH	0.677	0.079	drugX
49	F	NORMAL	HIGH	0.79	0.049	drugY
41	M	LOW	HIGH	0.767	0.069	drugC
60	M	NORMAL	HIGH	0.777	0.051	drugY
43	M	LOW	NORMAL	0.526	0.027	drugY

Categorical Variable

Age	Sex	BP	Cholesterol	Na	K	Drug
36	F	LOW	HIGH	0.697	0.069	

Example of multi-class classification

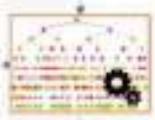
Age	Sex	BP	Cholesterol	Na	K	Drug
23	F	HIGH	HIGH	0.793	0.031	drugY
47	M	LOW	HIGH	0.739	0.056	drugC
47	M	LOW	HIGH	0.697	0.069	drugC
28	F	NORMAL	HIGH	0.564	0.072	drugX
61	F	LOW	HIGH	0.559	0.031	drugY
22	F	NORMAL	HIGH	0.677	0.079	drugX
49	F	NORMAL	HIGH	0.79	0.049	drugY
41	M	LOW	HIGH	0.767	0.069	drugC
60	M	NORMAL	HIGH	0.777	0.051	drugY
43	M	LOW	NORMAL	0.526	0.027	drugY

Categorical Variable

Modeling

Age	Sex	BP	Cholesterol	Na	K	Drug
36	F	LOW	HIGH	0.697	0.069	DrugX

Prediction



Classifier

Predicted Labels

Classification use cases

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

Classification use cases

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

- Which category a customer belongs to?
- Whether a customer switches to another provider/brand?
- Whether a customer responds to a particular advertising campaign?

Classification applications

Classification algorithms in machine learning

- Decision Trees (ID3, C4.5, C5.0)
- Naïve Bayes
- Linear Discriminant Analysis
- *k*-Nearest Neighbor
- Logistic Regression
- Neural Networks
- Support Vector Machines (SVM)

K-Nearest Neighbors

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Intro to KNN

X: Independent variable

Y: Dependent variable

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	188	4	31	0	0	5	?

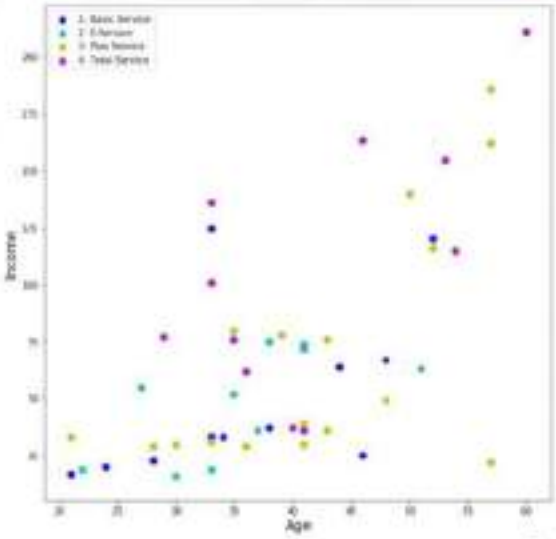
Value	Label
1	Basic Service
2	E-Service
3	Plus Service
4	Total Service

A telecommunications provider has segmented his customer base by service usage patterns, categorizing the customers into four groups.

If demographic data can be used to predict group membership, the company can customize offers for individual perspective customers.

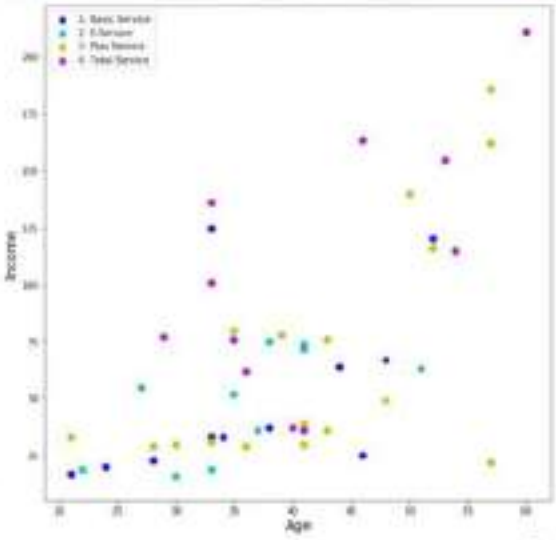
Determining the class using 1st KNN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	169	4	31	0	0	5	?



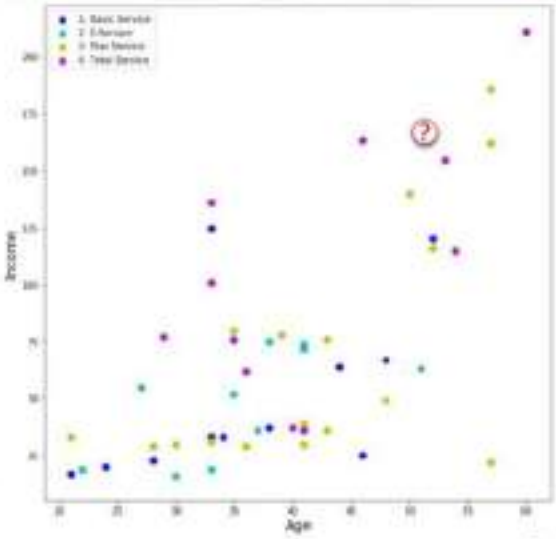
Determining the class using 1st KNN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	169	4	31	0	0	5	?



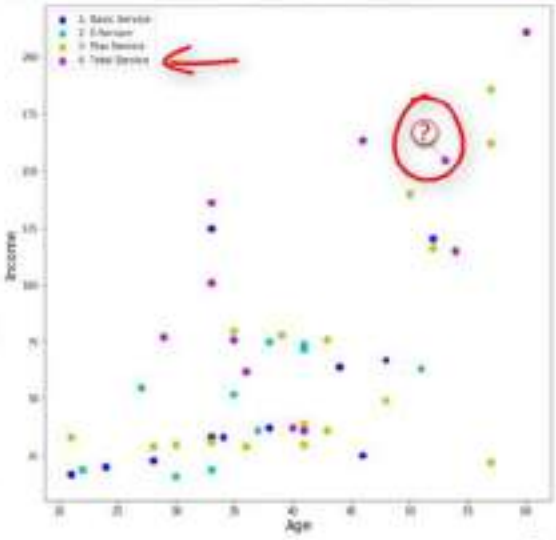
Determining the class using 1st KNN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	6	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	60	1	7	168	4	31	0	0	5	?



Determining the class using 1st KNN

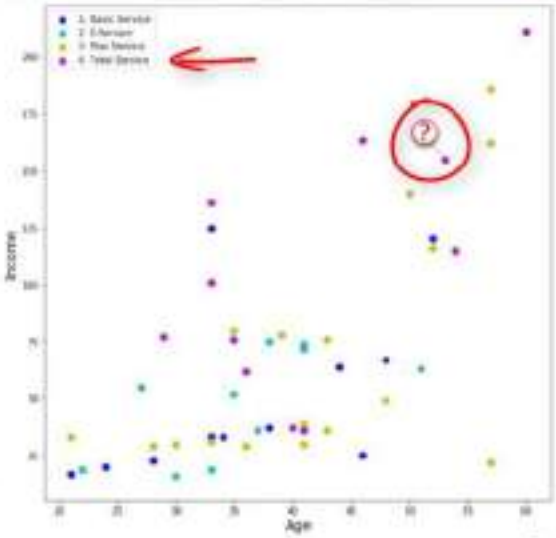
	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	6	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	60	1	7	168	4	31	0	0	5	?



Determining the class using 1st KNN

	region	age	marital	address	income	ed	employ	retire	gender	residx	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	35	1	7	136	5	5	0	0	6	4
2	3	62	1	36	116	1	26	0	1	2	3
3	3	35	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?

1-NN → 4: Total Service

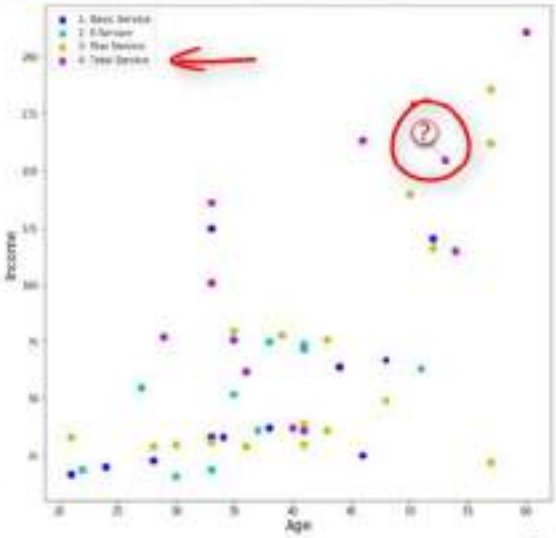


COGNITIVE CLASS

Determining the class using 1st KNN

	region	age	marital	address	income	ed	employ	retire	gender	residx	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	35	1	7	136	5	5	0	0	6	4
2	3	62	1	36	116	1	26	0	1	2	3
3	3	35	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?

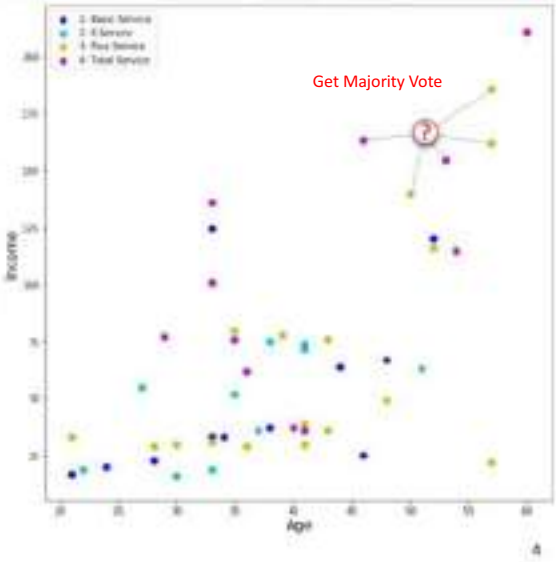
1-NN → 4: Total Service



COGNITIVE CLASS

Determining the class using the 5 KNNs

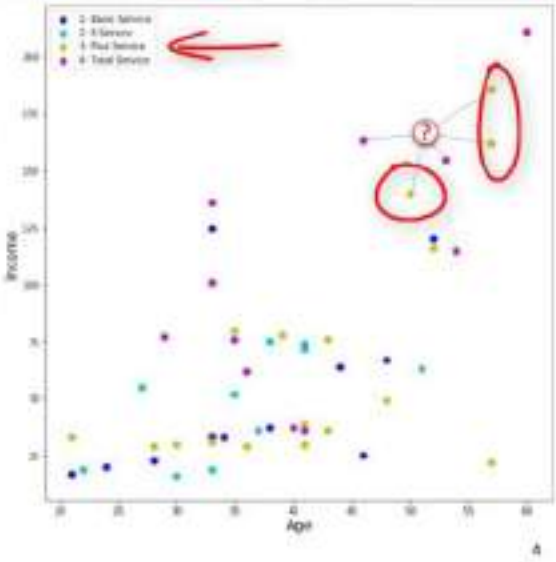
	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	3	0	2	1
1	3	33	1	7	136	5	5	3	0	6	4
2	3	52	1	24	116	1	29	3	1	2	3
3	2	33	0	12	33	2	0	3	1	1	1
4	2	30	1	9	30	1	2	3	0	4	3
5	2	39	0	17	78	2	16	3	1	1	3
6	3	22	1	2	19	2	4	3	1	5	2
7	2	35	0	5	70	2	10	3	0	3	4
8	3	50	1	7	166	4	31	3	0	5	?



COGNITIVE CLASS

Determining the class using the 5 KNNs

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	3	0	2	1
1	3	33	1	7	136	5	5	3	0	6	4
2	3	52	1	24	116	1	29	3	1	2	3
3	2	33	0	12	33	2	0	3	1	1	1
4	2	30	1	9	30	1	2	3	0	4	3
5	2	39	0	17	78	2	16	3	1	1	3
6	3	22	1	2	19	2	4	3	1	5	2
7	2	35	0	5	70	2	10	3	0	3	4
8	3	50	1	7	166	4	31	3	0	5	?

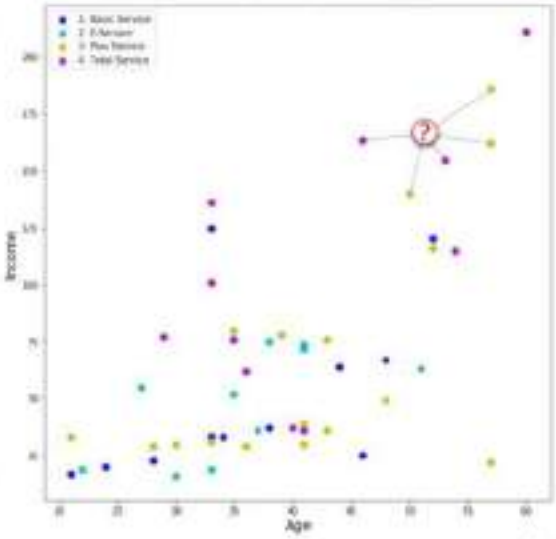


COGNITIVE CLASS

Determining the class using the 5 KNNs

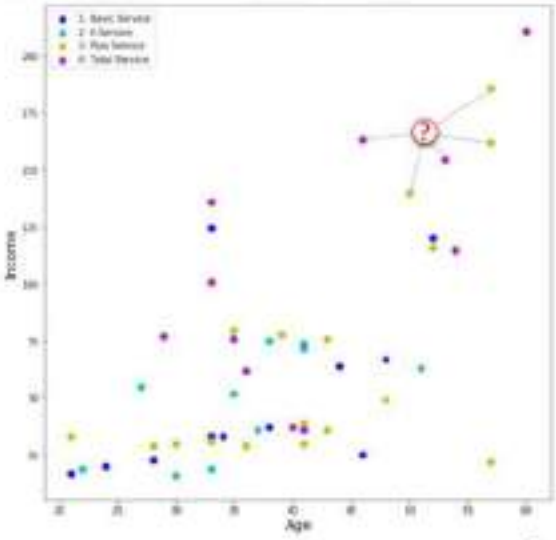
	region	age	marital	address	income	ed	employ	retire	gender	residx	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	6	2
7	2	35	0	5	70	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?

5-NN → 3: Plus Service



What is K-Nearest Neighbor (or KNN)?

- A method for classifying cases based on their similarity to other cases
- Cases that are near each other are said to be "neighbors"
- Based on similar cases with same class labels are near each other



The K-Nearest Neighbors algorithm

1. Pick a value for K.
2. Calculate the distance of unknown case from all cases.
3. Select the K-observations in the training data that are "nearest" to the unknown data point.
4. Predict the response of the unknown data point using the most popular response value from the K-nearest neighbors.

How to calculate distance?
How to pick a value for k?

Calculating the similarity/distance in a 1-dimensional space

Customer 1
Age
54

Customer 2
Age
50

We can use the Euclidian distance

Calculating the similarity/distance in a 1-dimensional space

Customer 1
Age
54

Customer 2
Age
50

$$\text{Dis}\,(x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

Calculating the similarity/distance in a 1-dimensional space

Customer 1
Age
54

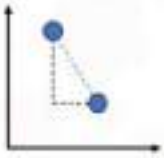
Customer 2
Age
50

$$\text{Dis}\,(x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

$$\text{Dis}\,(x_1, x_2) = \sqrt{(34 - 30)^2} = 4$$

Calculating the similarity/distance in a 2-dimensional space

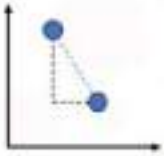
Customer 1	
Age	Income
54	190



Customer 2	
Age	Income
50	200

Calculating the similarity/distance in a 2-dimensional space

Customer 1	
Age	Income
54	190



Customer 2	
Age	Income
50	200

$$\text{Dis}\,(x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

$$= \sqrt{(54-50)^2 + (190-200)^2} = 10.77$$

Calculating the similarity/distance in a multi-dimensional space

Customer 1		
Age	Income	Education
54	190	3

Customer 2		
Age	Income	Education
50	200	8

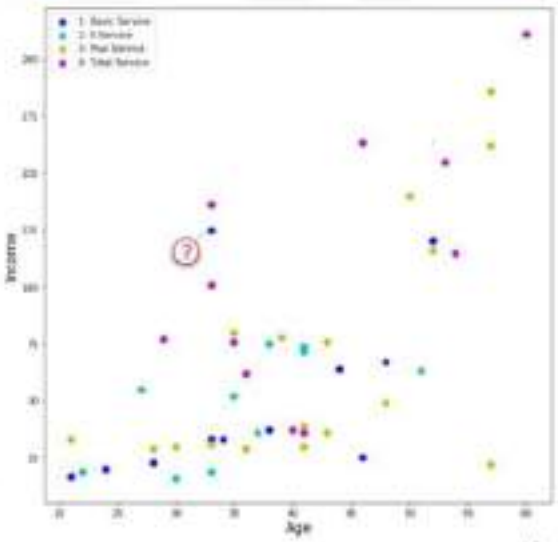
$$\text{Dis}\ (x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

$$= \sqrt{(54-50)^2+(190-200)^2+(3-8)^2} = 11.87$$

Do not forget to normalize!

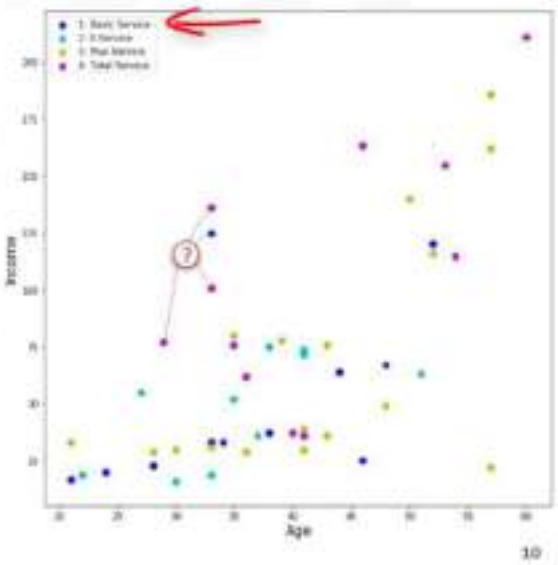
What is the best value of K for KNN?

- K =1 class 1



What is the best value of K for KNN?

- K =1 class 1



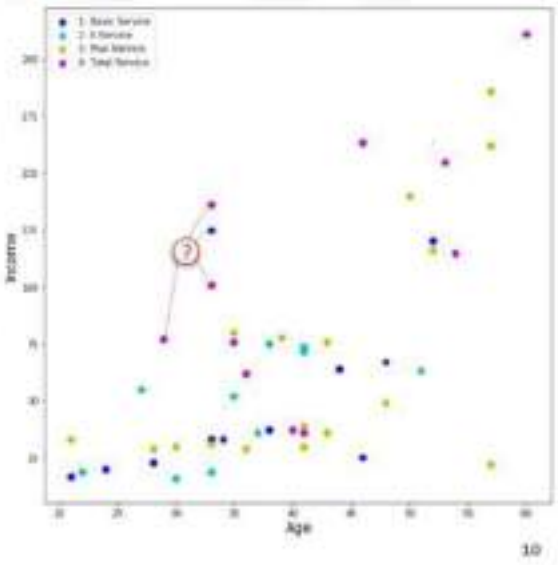
Since its nearest neighbor is blue we can say that we capture the noise in the data or,

we chose one of the points that was an anomaly in the data.

A low value of K causes a highly complex model as well, which might result in overfitting of the model.

What is the best value of K for KNN?

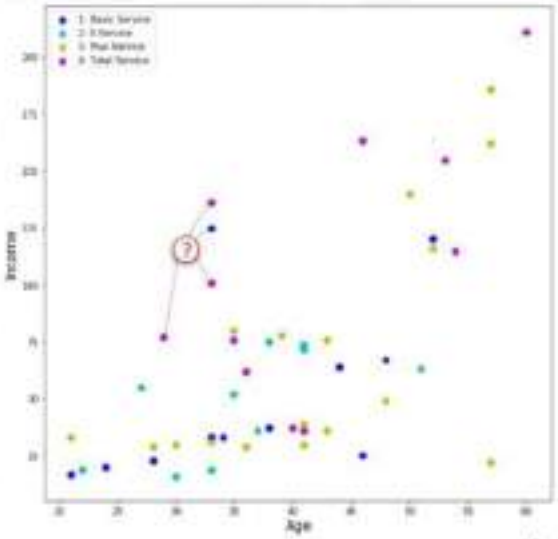
- K =1 class 1
- K =20 ?



if we choose a very high value of K such as K equals 20, then the model becomes overly generalized.

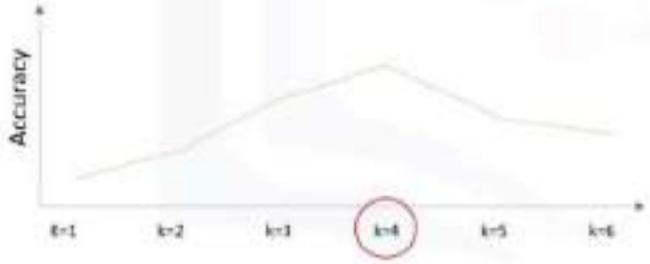
What is the best value of K for KNN?

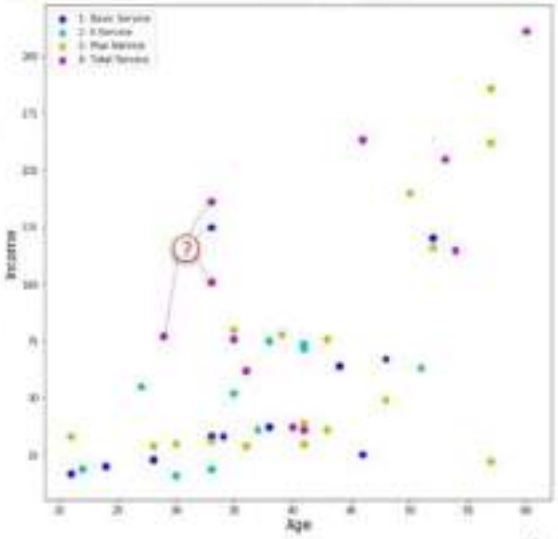
- K =1 class 1
- K =20 ?



What is the best value of K for KNN?

- K =1 class 1
- K =20 ?





Evaluation Metrics in Classification

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Classification accuracy

Classification accuracy

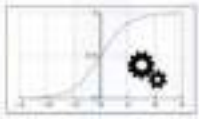
Classification accuracy

Test set

	tenure	age	address	income	ed	employ	equip	calicard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test

Classification accuracy



Test set

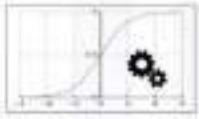
	tenure	age	address	income	ed	employ	equip	calicard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test

Predicted churn
1
1
0
1
0

COGNITIVE CLASS

Classification accuracy



Test set

	tenure	age	address	income	ed	employ	equip	calicard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test

Predicted churn
1
1
0
1
0

Actual Labels y

$\hat{y}$ Predicted Labels

COGNITIVE CLASS

Jaccard index

y: Actual labels

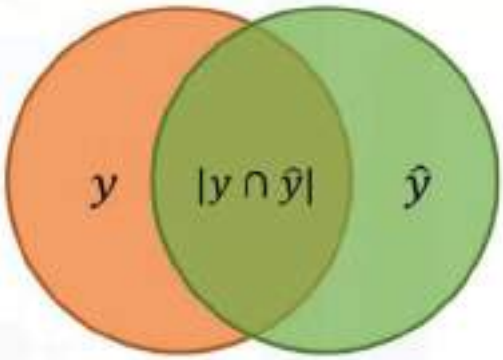
$\hat{y}$: Predicted labels

Jaccard index

y: Actual labels

$\hat{y}$: Predicted labels

$$J(y, \hat{y}) = \frac{|y \cap \hat{y}|}{|y \cup \hat{y}|} = \frac{|y \cap \hat{y}|}{|y| + |\hat{y}| - |y \cap \hat{y}|}$$



y: [0, 0, 0, 0, 0, 1, 1, 1, ,1 ,1]

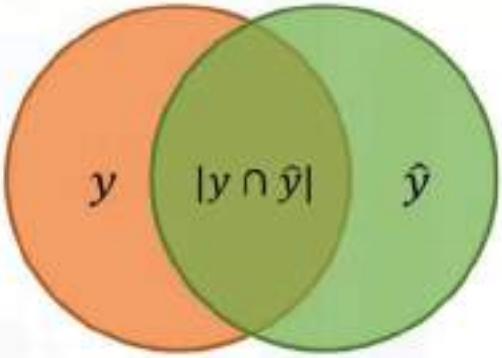
$\hat{y}$: [1, 1, 0, 0, 0, 1, 1, 1, ,1 ,1]

Jaccard index

y: Actual labels

$\hat{y}$: Predicted labels

$$J(y,\hat{y}) = \frac{|y \cap \hat{y}|}{|y \cup \hat{y}|} = \frac{|y \cap \hat{y}|}{|y| + |\hat{y}| - |y \cap \hat{y}|}$$



y: [0, 0, 0, 0, 0, 1, 1, 1, 1, 1]

$\hat{y}$: [1, 1, 0, 0, 0, 1, 1, 1, 1, 1]

$$\boxed{J(y,\hat{y}) = \frac{8}{10+10-8} = 0.66}$$

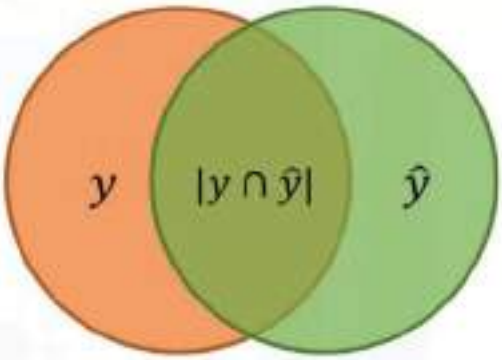
COGNITIVE CLASS

Jaccard index

y: Actual labels

$\hat{y}$: Predicted labels

$$J(y,\hat{y}) = \frac{|y \cap \hat{y}|}{|y \cup \hat{y}|} = \frac{|y \cap \hat{y}|}{|y| + |\hat{y}| - |y \cap \hat{y}|}$$



y: [0, 0, 0, 0, 0, 1, 1, 1, 1, 1]

$\hat{y}$: [1, 1, 0, 0, 0, 1, 1, 1, 1, 1]

$$J(y,\hat{y}) = \frac{8}{10+10-8} = 0.66$$

$J(y,\hat{y}) = 1.0$

Higher Accuracy

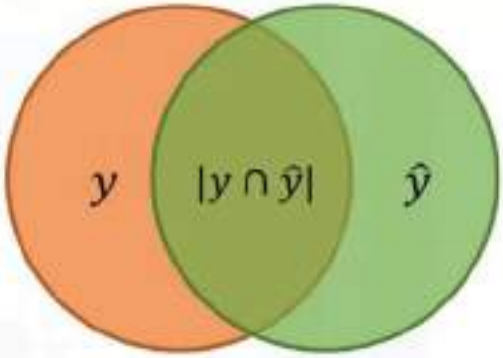
COGNITIVE CLASS

Jaccard index

y: Actual labels

$\hat{y}$: Predicted labels

$$J(y,\hat{y}) = \frac{|y \cap \hat{y}|}{|y \cup \hat{y}|} = \frac{|y \cap \hat{y}|}{|y| + |\hat{y}| - |y \cap \hat{y}|}$$

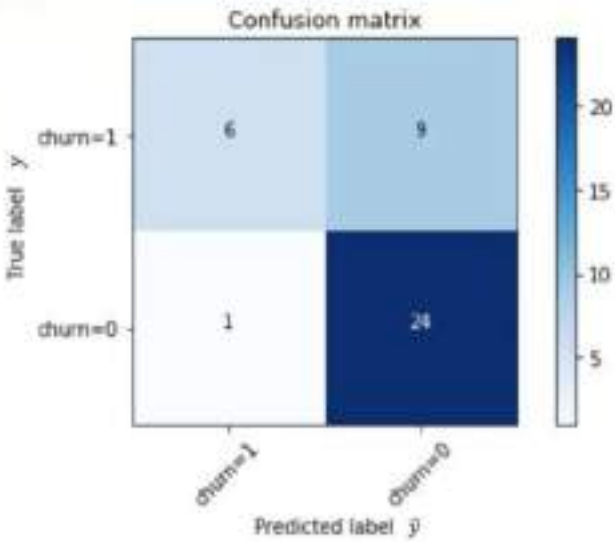


y: [0, 0, 0, 0, 0, 1, 1, 1, 1, 1]

ŷ: [1, 1, 0, 0, 0, 1, 1, 1, 1, 1]

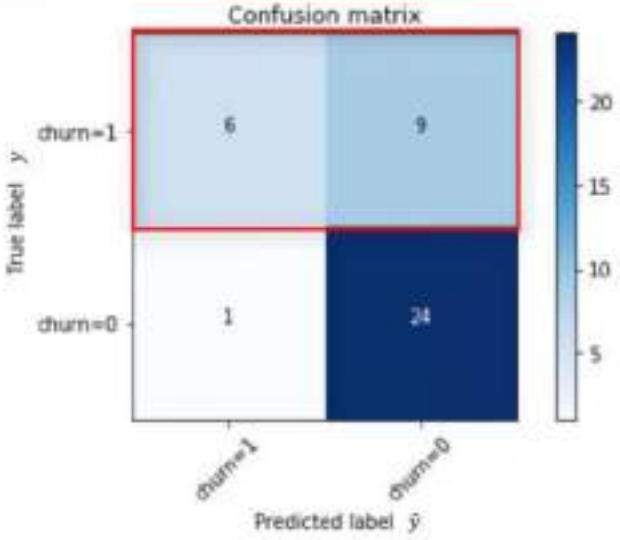
$$J(y,\hat{y}) = \frac{8}{10+10-8} = 0.66$$

F1-score

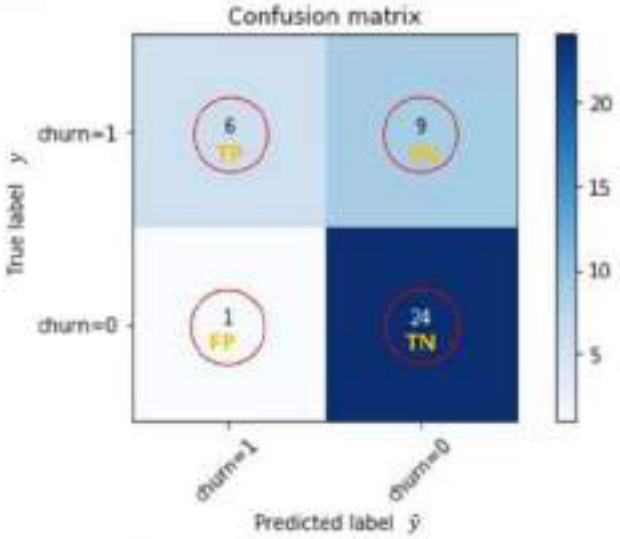


Another way of looking at accuracy of classifiers is to look at a confusion matrix.

F1-score



F1-score

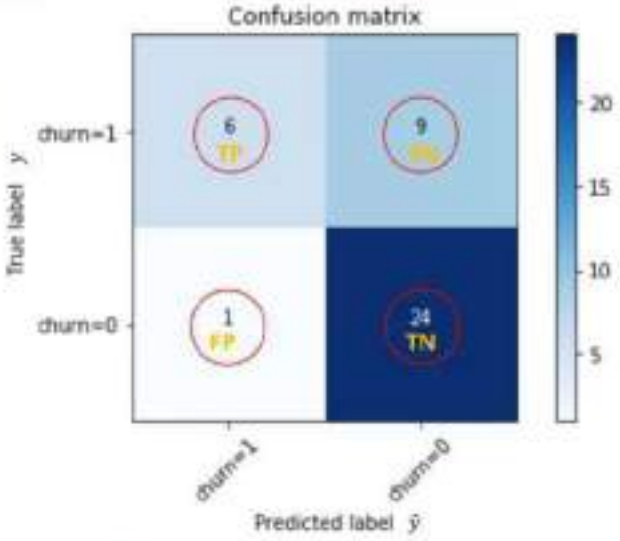


F1-score

- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)

The F1 score is the harmonic average of the precision and recall.

	precision	recall
Churn = 0	0.73	0.96
Churn = 1	0.86	0.40



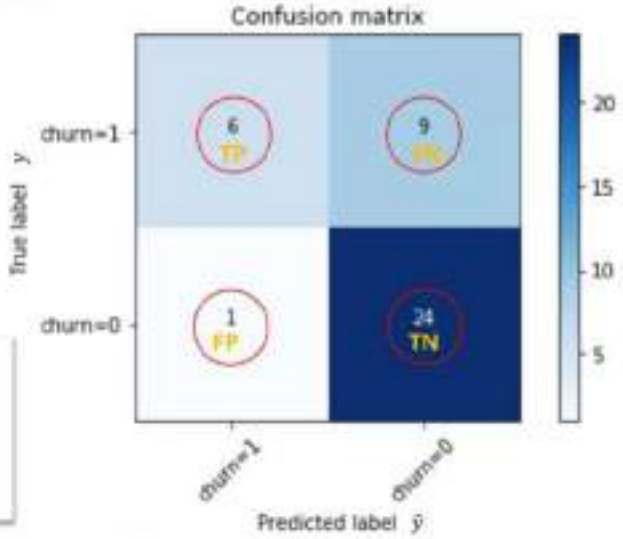
F1-score

- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2x (prc x rec) / (prc+rec)

F1-score: 0.00 ... 0.20 .... 0.55 .... 0.83 ... 1.00

Higher Accuracy

	precision	recall	f1-score
Churn = 0	0.73	0.96	0.83
Churn = 1	0.86	0.40	0.55



F1-score

It is a good way to show that a classifier has a good value for both recall and precision.

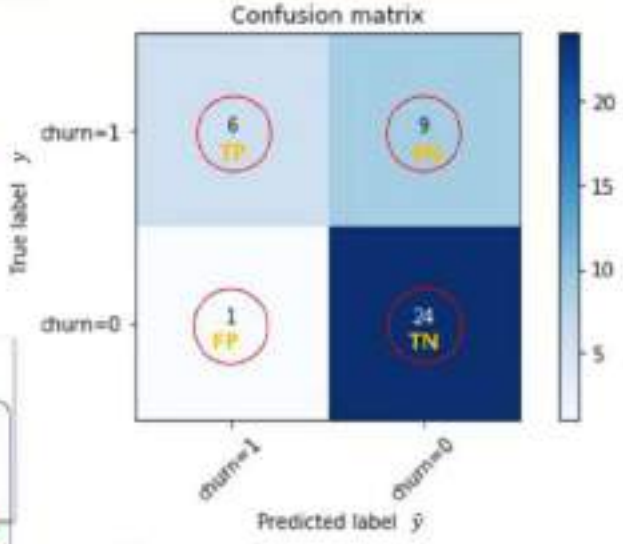
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2x (prc x rec) / (prc+rec)

F1-score: 0.00 ... 0.20 .... 0.55 .... 0.83 ... 1.00

Higher Accuracy

	precision	recall	f1-score
Churn = 0	0.73	0.96	0.83
Churn = 1	0.86	0.40	0.55

Avg Accuracy = 0.72



COGNITIVE CLASS

4

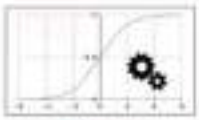
Log loss

Sometimes, the output of a classifier is the probability of a class label, instead of the label.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test



Predicted churn
1
0
0
0
1

Actual Labels y

$\hat{y}$ Predicted Labels

COGNITIVE CLASS

For example, in logistic regression, the output can be the probability of customer churn, i.e., yes (or equals to 1).

5

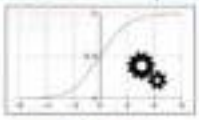
Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test



Predicted churn
0.91
0.13
0.04
0.23
0.43

Actual Labels y

$\hat{y}$ Predicted Probability

COGNITIVE CLASS

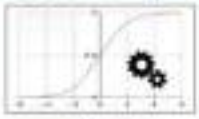
Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test



Predicted churn
0.91
0.13
0.04
0.23
0.43

Actual Labels y

$\hat{y}$ Predicted Probability

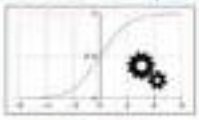
COGNITIVE CLASS

Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Test

Predicted churn
0.91
0.13
0.04
0.23
0.43

Actual Labels y

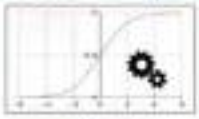
$\hat{y}$ Predicted Probability

Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Test

Predicted churn
0.91
0.13
0.04
0.23
0.43

Actual Labels y

$\hat{y}$ Predicted Probability

$$(y \times \log(\hat{y}) + (1 - y) \times \log(1 - \hat{y}))$$

Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test

Predicted churn	LogLoss
0.91	0.11
0.13	2.04
0.04	0.04
0.23	0.26
0.43	0.56

Actual Labels y

$\hat{y}$ Predicted Probability

$$(y \times \log(\hat{y}) + (1-y) \times \log(1-\hat{y}))$$

COGNITIVE CLASS

Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test

Predicted churn	LogLoss
0.91	0.11
0.13	2.04
0.04	0.04
0.23	0.26
0.43	0.56

Actual Labels y

$\hat{y}$ Predicted Probability

$$LogLoss = -\frac{1}{n}\sum(y \times \log(\hat{y}) + (1-y) \times \log(1-\hat{y}))$$

COGNITIVE CLASS

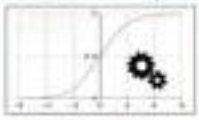
Log loss

Performance of a classifier where the predicted output is a probability value between 0 and 1.

Test set

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

Test



Predicted churn	LogLoss
0.91	0.11
0.13	2.04
0.04	0.04
0.23	0.26
0.43	0.56

$LogLoss = 0.60$

Actual Labels y

$\hat{y}$ Predicted Probability

$$LogLoss = -\frac{1}{n}\sum(y \times \log(\hat{y}) + (1-y) \times \log(1-\hat{y}))$$

LogLoss: 0.00 ... 0.35 .... 0.60 ... 1.00

Higher Accuracy

Intro to Decision Trees

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

What is a decision tree?

What is a decision tree?

The basic intuition behind a decision tree is to map out all possible decision paths in the form of a tree.

Narendra Nath Joshi

How to build a decision tree?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A
p15	Middle-age	F	Low	Normal	?

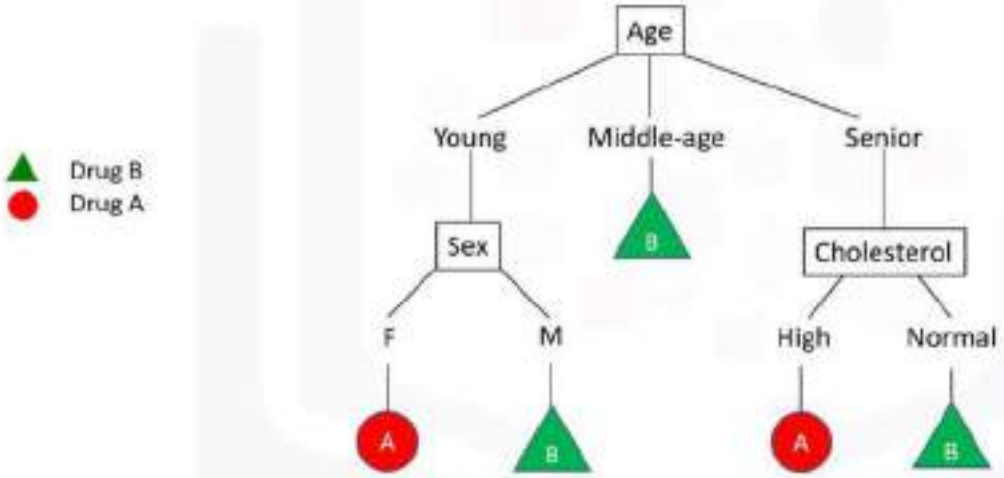
How to build a decision tree?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A
p15	Middle-age	F	Low	Normal	?

How to build a decision tree?

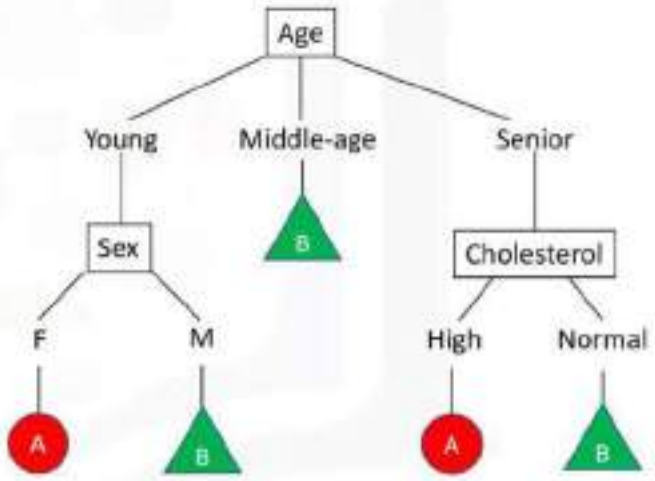
Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A
p15	Middle-age	F	Low	Normal	?

Building a decision tree with the training set



Decision tree learning algorithm

1. Choose an attribute from your dataset.
2. Calculate the significance of attribute in splitting of data.
3. Split data based on the value of the best attribute.
4. Go to step 1.



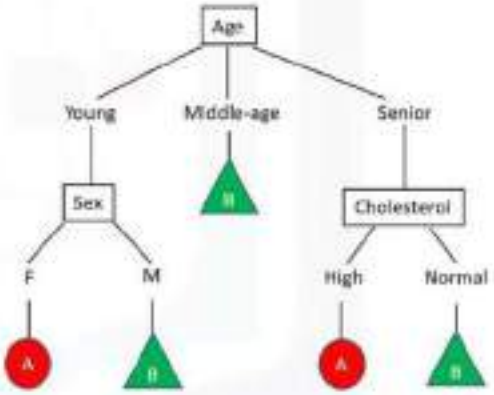
Building Decision Trees

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

How do you build a decision tree?

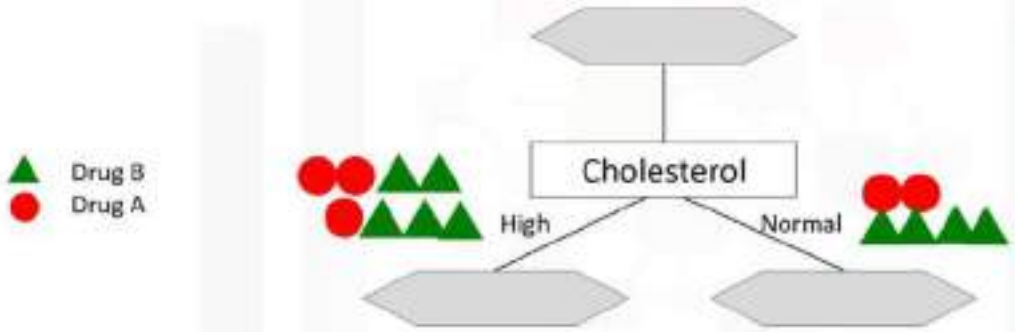
Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A
p15	Middle-age	F	Low	Normal	?



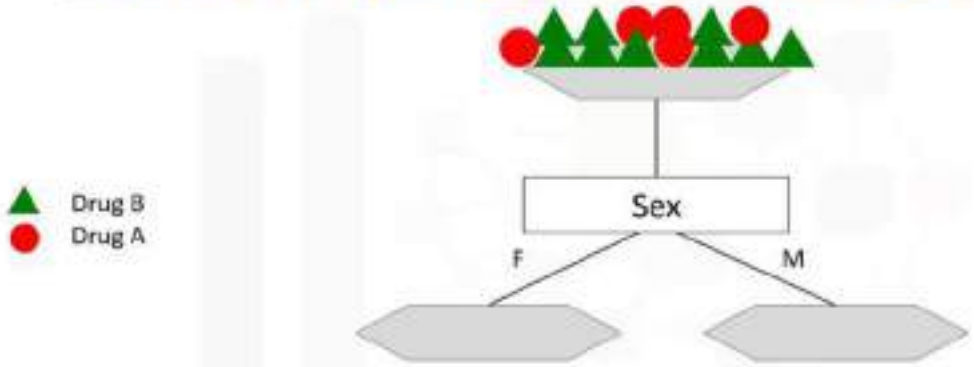
Which attribute is the best ?

Drug B
Drug A

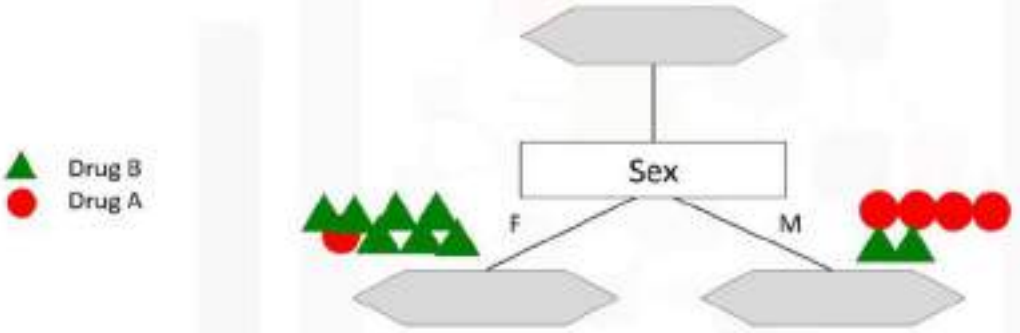
Which attribute is the best ?



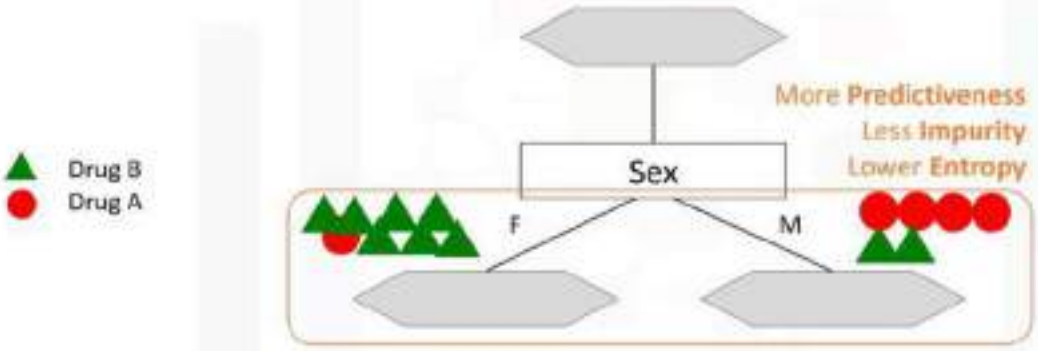
Which attribute is the best ?



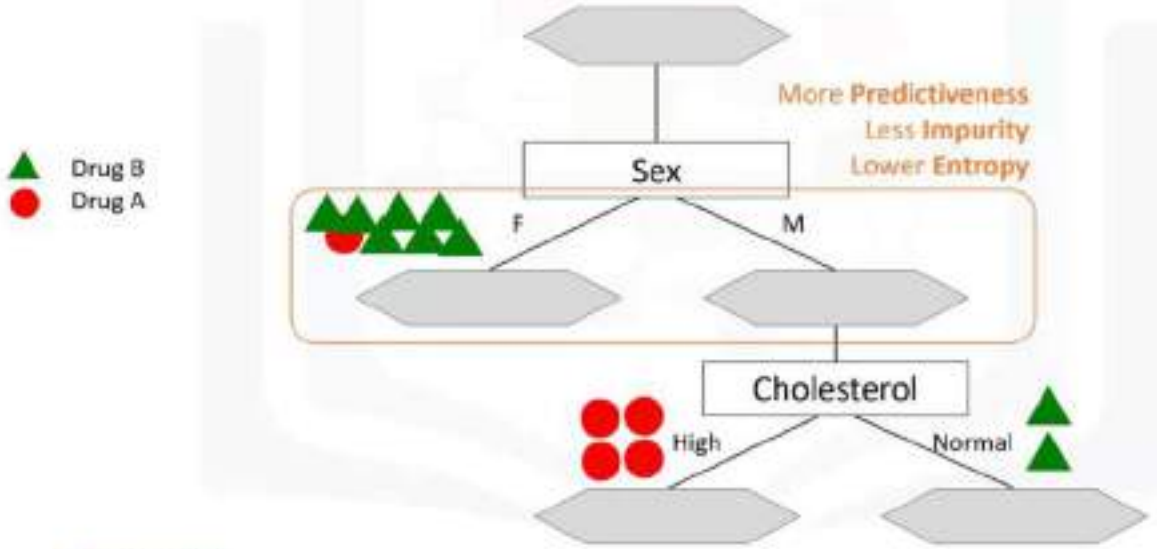
Which attribute is the best ?



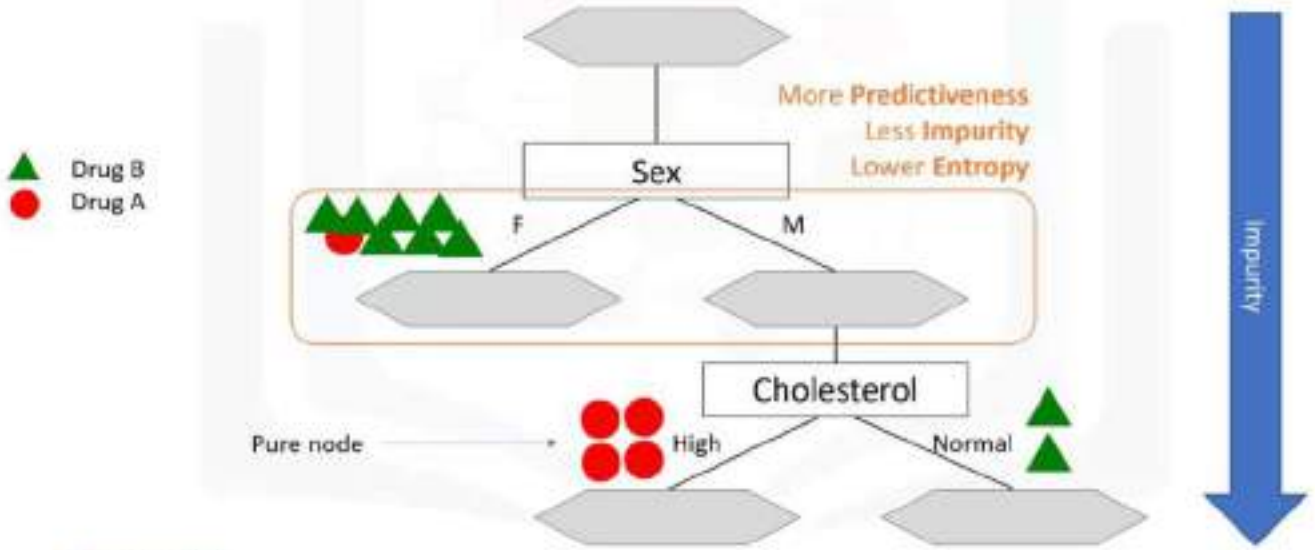
Which attribute is the best ?



Which attribute is the best ?

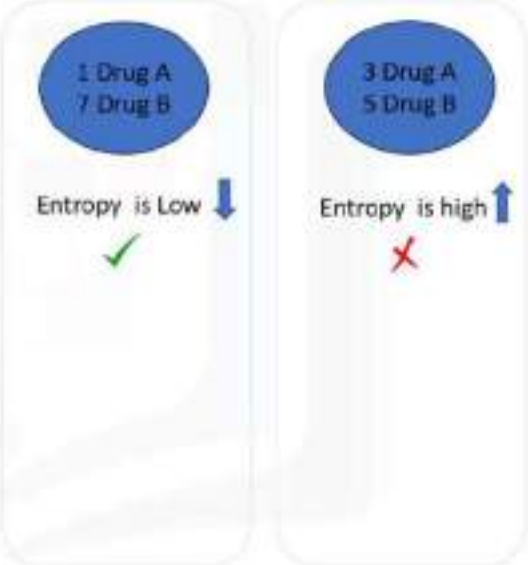


Which attribute is the best ?



Entropy

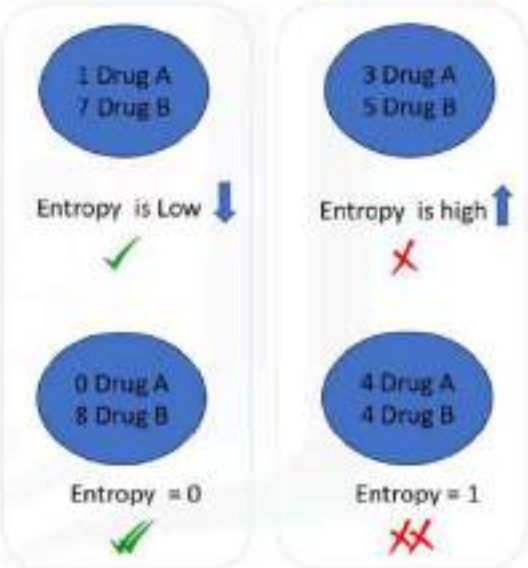
- Measure of randomness or uncertainty



The lower the Entropy, the less uniform the distribution, the purer the node.

Entropy

- Measure of randomness or uncertainty



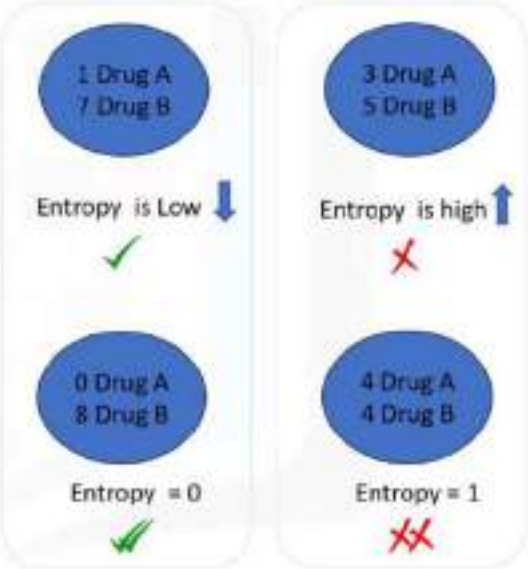
The lower the Entropy, the less uniform the distribution, the purer the node.

Entropy

- Measure of randomness or uncertainty

Entropy = - p(A)log(p(A)) - p(B)log(p(B))

The lower the Entropy, the less uniform the distribution, the purer the node.



Which attribute is the best one to use?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

Which attribute is the best one to use?

S: [9 B, 5 A]

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

Which attribute is the best one to use?

S: [9 B, 5 A]

$E = -p(B)\log(p(B)) - p(A)\log(p(A))$

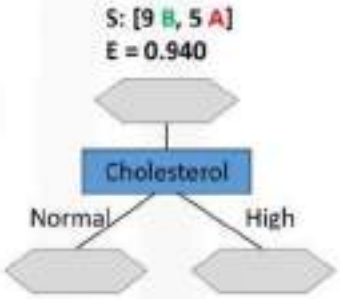
$E = -(9/14)\log(9/14) - (5/14)\log(5/14)$

E = 0.940

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

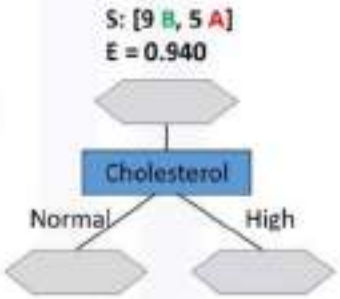
Is 'Cholesterol' the best attribute?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A



Is 'Cholesterol' the best attribute?

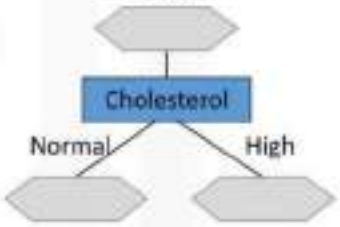
Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A



Is 'Cholesterol' the best attribute?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

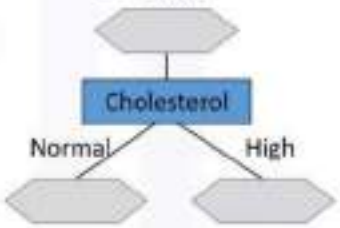
S: [9 B, 5 A]
E = 0.940



Is 'Cholesterol' the best attribute?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

S: [9 B, 5 A]
E = 0.940

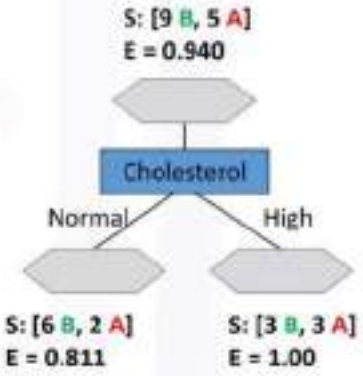


S: [6 B, 2 A]
E = 0.811

S: [3 B, 3 A]
E = 1.00

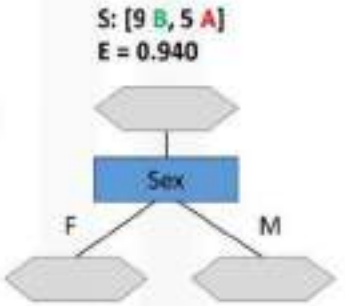
Is 'Cholesterol' the best attribute?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A



What about 'Sex'?

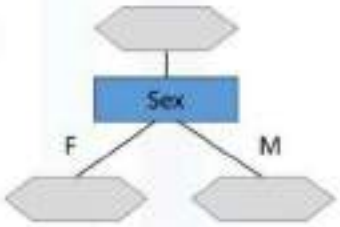
Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	Hiigh	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A



What about 'Sex'?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

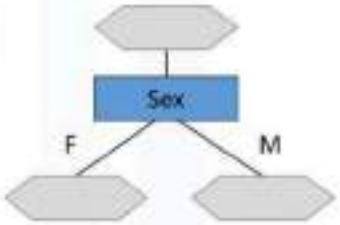
S: [9 B, 5 A]
E = 0.940



What about 'Sex'?

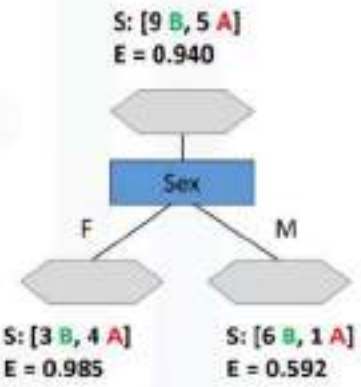
Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

S: [9 B, 5 A]
E = 0.940

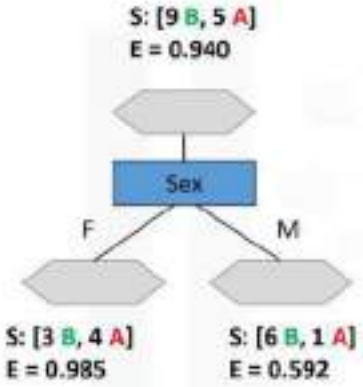


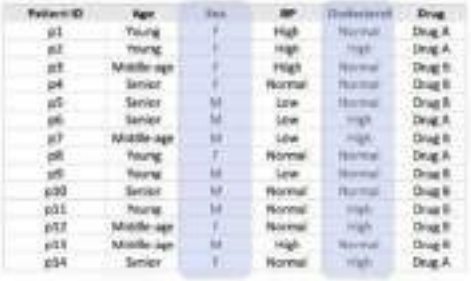
What about 'Sex'?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

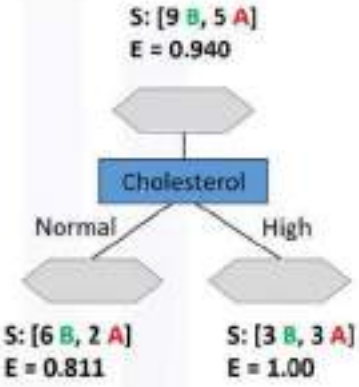


Which attribute is the best?





?



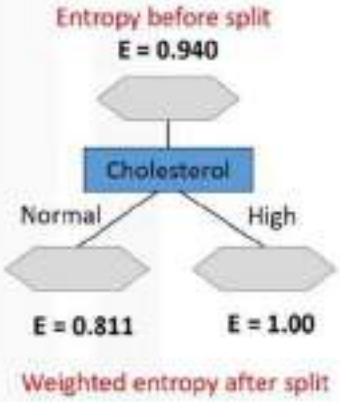
Which attribute is the best?

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

The tree with the higher Information Gain after splitting.

What is information gain?

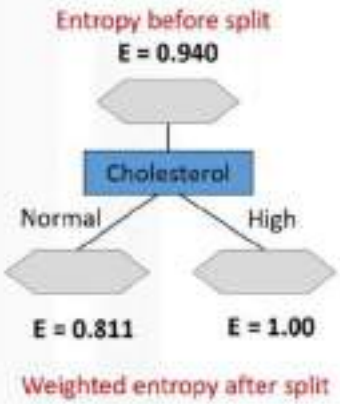
Information gain is the information that can increase the level of certainty after splitting.



What is information gain?

Information gain is the information that can increase the level of certainty after splitting.

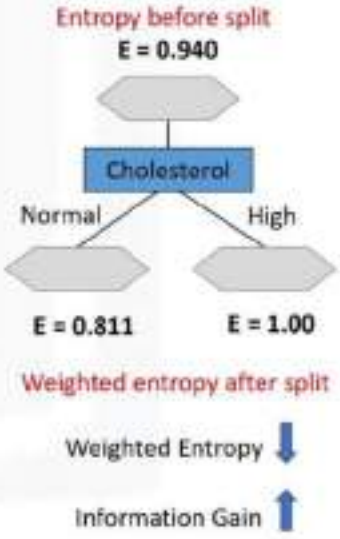
Information Gain = (Entropy before split) – (weighted entropy after split)



What is information gain?

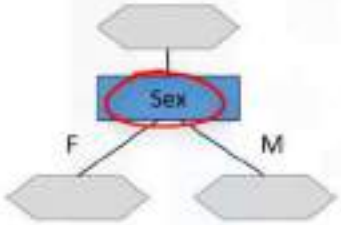
Information gain is the information that can increase the level of certainty after splitting.

Information Gain = (Entropy before split) – (weighted entropy after split)



Which attribute is the best?

S: [9 B, 5 A]
E = 0.940



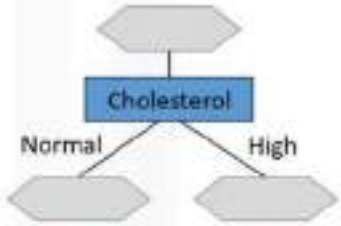
S: [3 B, 4 A]
E = 0.985

S: [6 B, 1 A]
E = 0.592

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

?

S: [9 B, 5 A]
E = 0.940



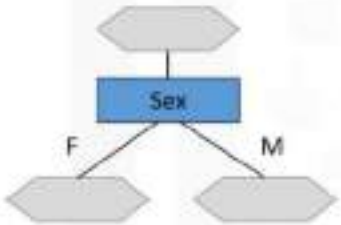
S: [6 B, 2 A]
E = 0.811

S: [3 B, 3 A]
E = 1.00

Gain (s, Sex)
= 0.940 – [(7/14)0.985 + (7/14)0.592]
=0.151

Which attribute is the best?

S: [9 B, 5 A]
E = 0.940



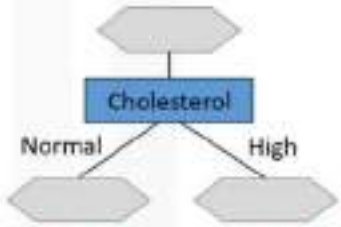
S: [3 B, 4 A]
E = 0.985

S: [6 B, 1 A]
E = 0.592

Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

?

S: [9 B, 5 A]
E = 0.940



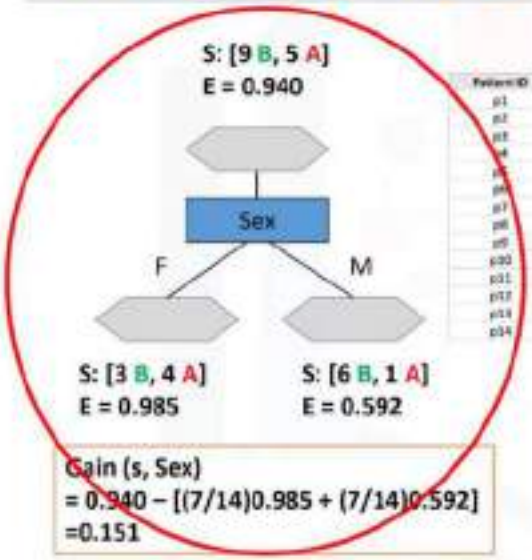
S: [6 B, 2 A]
E = 0.811

S: [3 B, 3 A]
E = 1.00

Gain (s, Sex)
= 0.940 – [(7/14)0.985 + (7/14)0.592]
=0.151

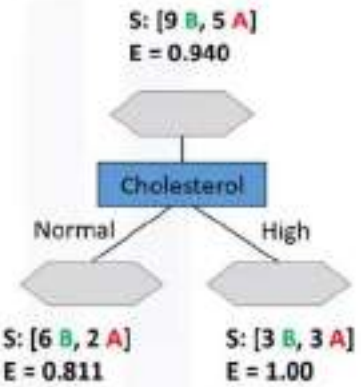
Gain (s, Cholesterol)
= 0.940 – [(8/14).811 + (6/14)1.0]
=0.048

Which attribute is the best?



Patient ID	Age	Sex	BP	Cholesterol	Drug
p1	Young	F	High	Normal	Drug A
p2	Young	F	High	High	Drug A
p3	Middle-age	F	High	Normal	Drug B
p4	Senior	F	Normal	Normal	Drug B
p5	Senior	M	Low	Normal	Drug B
p6	Senior	M	Low	High	Drug A
p7	Middle-age	M	Low	High	Drug B
p8	Young	F	Normal	Normal	Drug A
p9	Young	M	Low	Normal	Drug B
p10	Senior	M	Normal	Normal	Drug B
p11	Young	M	Normal	High	Drug B
p12	Middle-age	F	Normal	High	Drug B
p13	Middle-age	M	High	Normal	Drug B
p14	Senior	F	Normal	High	Drug A

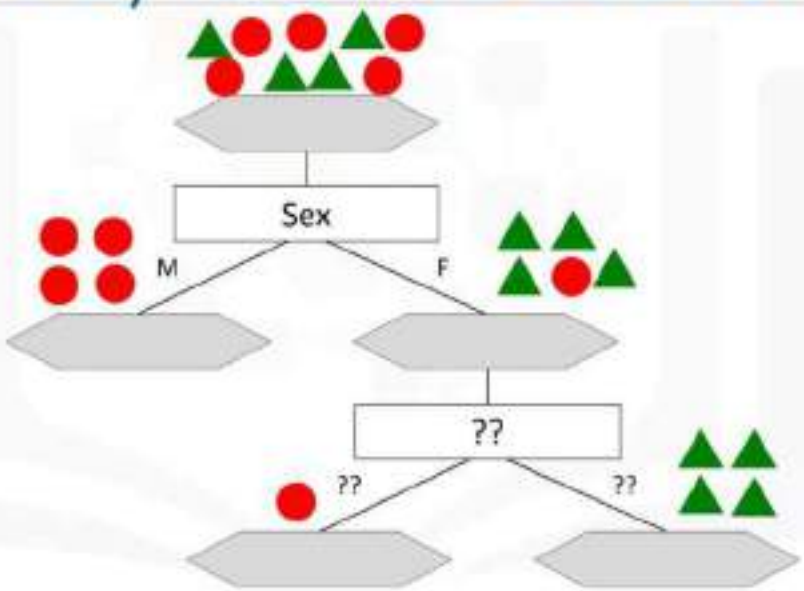
?



Gain (s, Sex)
= 0.940 – [(7/14)0.985 + (7/14)0.592]
=0.151

Gain (s, Cholesterol)
= 0.940 – [(8/14).811 + (6/14)1.0]
=0.048

Correct way to build a decision tree



Intro to Logistic Regression

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

What is logistic regression?

Logistic regression is a classification algorithm for categorical variables.

What is logistic regression?

Logistic regression is a classification algorithm for categorical variables.

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

What is logistic regression?

Logistic regression is a classification algorithm for categorical variables.

Independent variables

Dependent variable

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

COGNITIVE CLASS

2

What is logistic regression?

Logistic regression is a classification algorithm for categorical variables.

Independent variables

Dependent variable

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

What is logistic regression?

Logistic regression is a classification algorithm for categorical variables.

Independent variables

Dependent variable

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

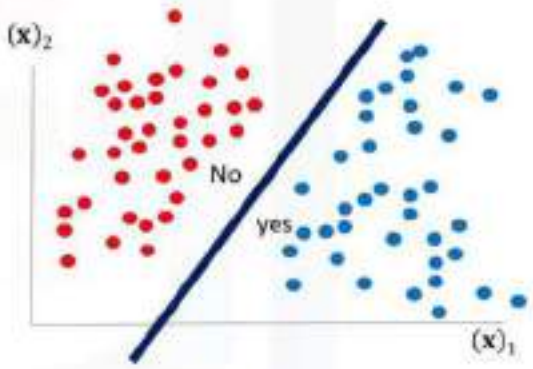
Categorical Variable

Continuous/Categorical variables

Logistic regression applications

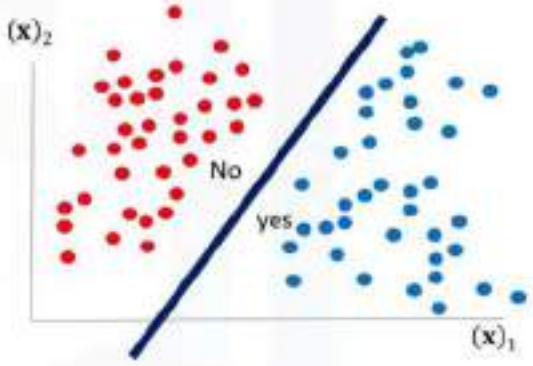
- Predicting the probability of a person having a heart attack
- Predicting the mortality in injured patients
- Predicting a customer's propensity to purchase a product or halt a subscription
- Predicting the probability of failure of a given process or product
- Predicting the likelihood of a homeowner defaulting on a mortgage

When is logistic regression suitable?



When is logistic regression suitable?

- If your data is binary
  - 0/1, YES/NO, True/False
- If you need probabilistic results
- When you need a linear decision boundary
- If you need to understand the impact of a feature



$\theta_0 + \theta_1 x_1 + \theta_2 x_2 > 0$

Building a model for customer churn

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No

Building a model for customer churn

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No

$\mathbf{x}$ y

Building a model for customer churn

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$\mathbf{x}$ y

Building a model for customer churn

$\mathbf{x}$ y

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$X \in \mathbb{R}^{m \times n}$
$y \in \{0,1\}$

Building a model for customer churn

$\mathbf{x}$ y

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$X \in \mathbb{R}^{m \times n}$
$y \in \{0,1\}$

$\boxed{\hat{y} = \mathbf{P(y=1|x)}}$

Building a model for customer churn

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$\mathbf{x}$ y

$X \in \mathbb{R}^{m \times n}$

$y \in \{0,1\}$

$\hat{y} = P(y=1|x)$

$P(y=0|x) = 1 - P(y=1|x)$

Logistic Regression vs Linear Regression

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

Model of customer churn data

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	No

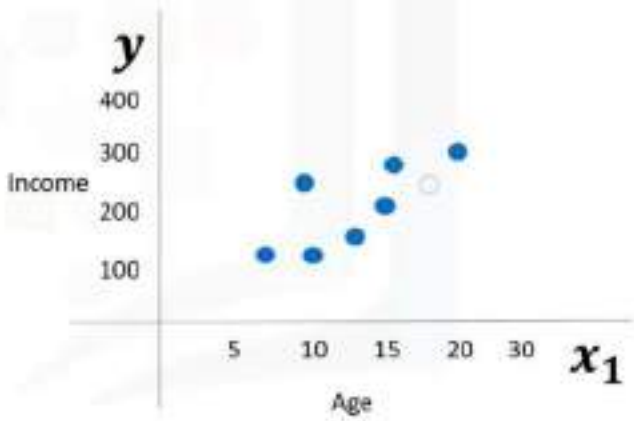
$\mathbf{x}$ y

$\hat{y} = P(y=1|x)$

© 2018 IBM Corporation

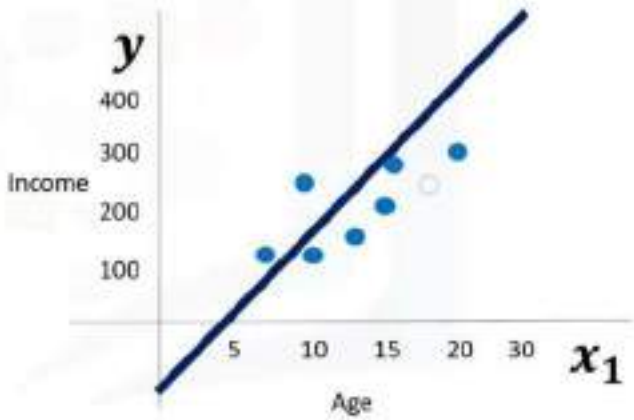
Predicting customer income

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



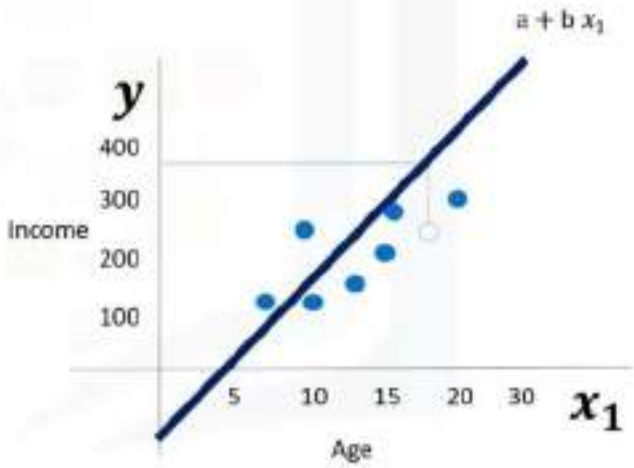
Predicting customer income

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Predicting customer income

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

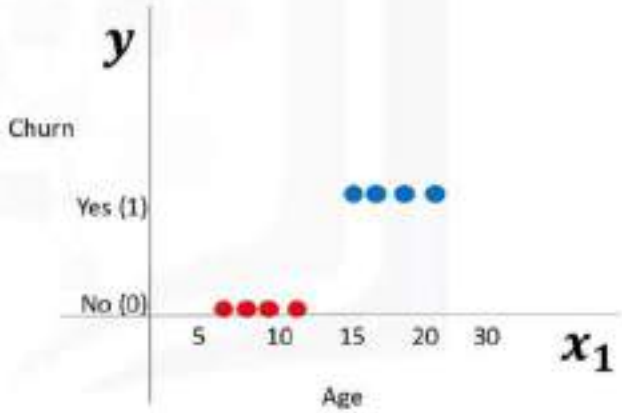


Predicting churn using linear regression

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

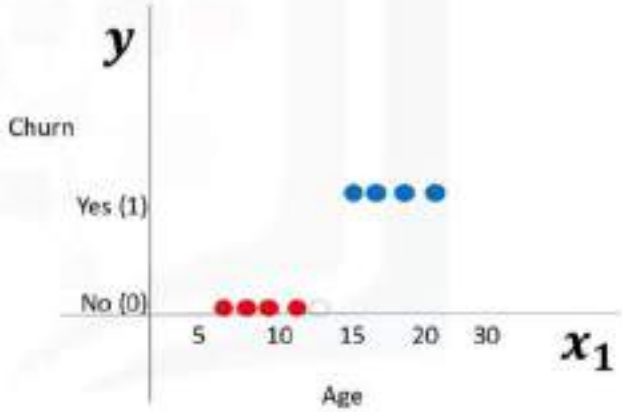
Predicting churn using linear regression

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0

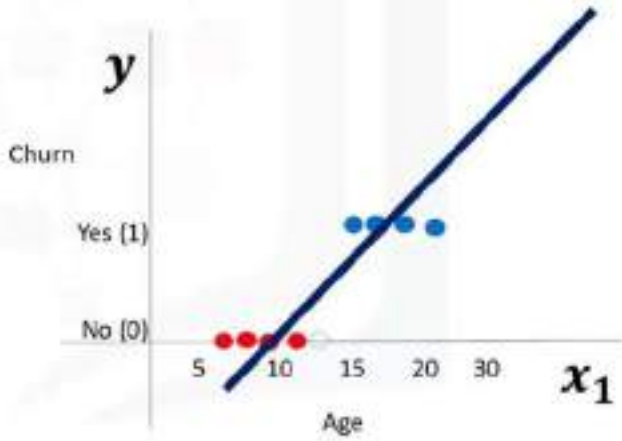


Predicting churn using linear regression

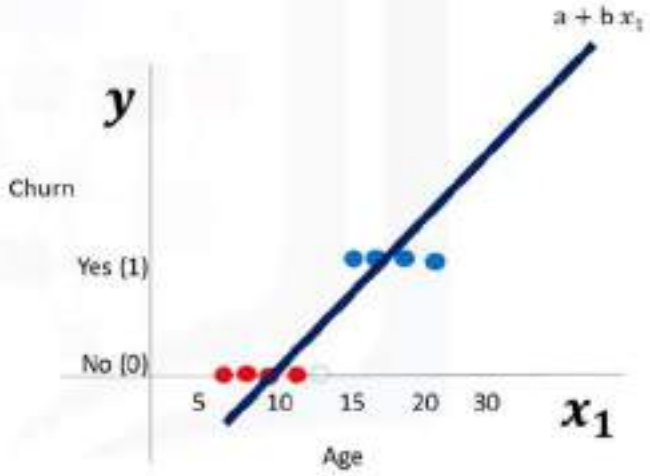
	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Predicting churn using linear regression



Predicting churn using linear regression

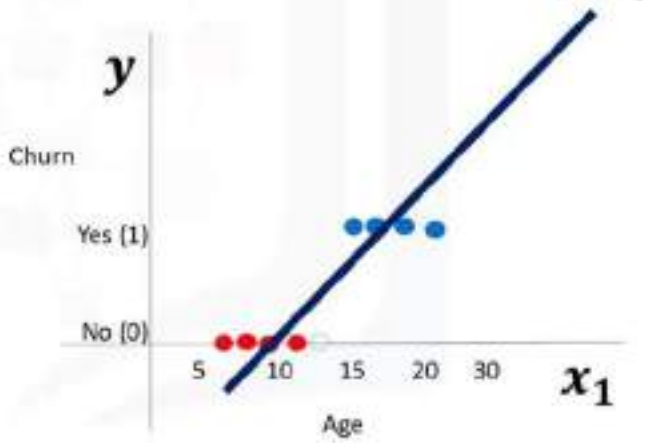


Predicting churn using linear regression

$\theta^T = [\theta_0, \theta_1]$

$\theta_0 + \theta_1 x_1$

$a + b x_1$



Predicting churn using linear regression

$\theta^T = [\theta_0, \theta_1]$

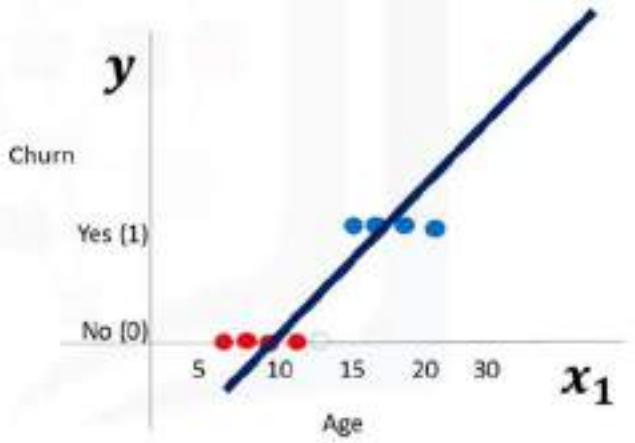
$\theta_0 + \theta_1 x_1$

$a + b\,x_1$

$\theta^T X = \theta_0 + \theta_1 x_1$

$\theta^T X = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots$

$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$ $\quad X = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \ldots \end{bmatrix}$



Predicting churn using linear regression

$\theta^T = [\theta_0, \theta_1]$

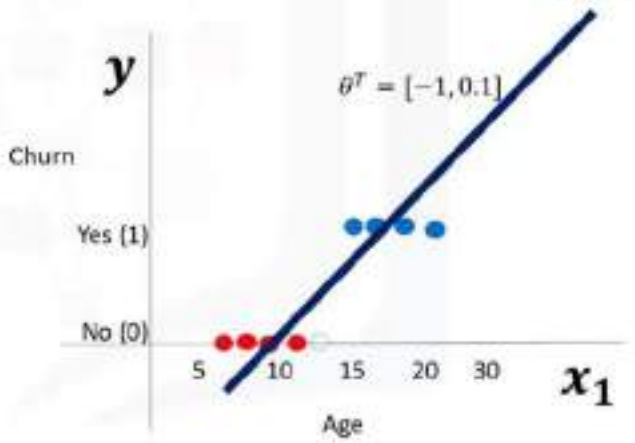
$\theta_0 + \theta_1 x_1$

$a + b\,x_1$

$\theta^T X = \theta_0 + \theta_1 x_1$

$\theta^T X = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots$

$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$ $\quad X = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \ldots \end{bmatrix}$



Predicting churn using linear regression

$\theta^T = [\theta_0, \theta_1]$

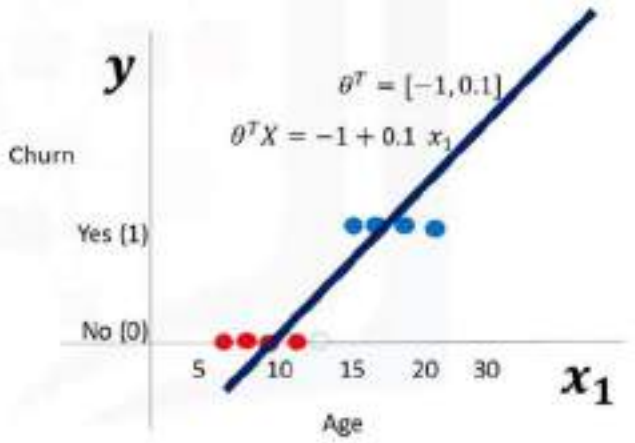
$\theta_0 + \theta_1 x_1$

$a + b\,x_1$

$\theta^T X = \theta_0 + \theta_1 x_1$

$\theta^T X = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots$

$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$ $\quad X = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \ldots \end{bmatrix}$

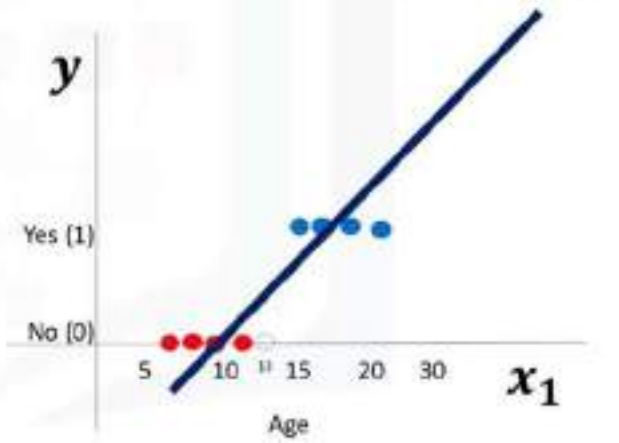


Linear regression in classification problems?

$\theta^T X = \theta_0 + \theta_1 x_1$

$\theta^T X = -1 + 0.1\ .\ x_1$

$p_1 = [13]$ → $\theta^T X = -1 + 0.1\ .\ x_1$
$= -1 + 0.1 \times 13$
$= 0.3$

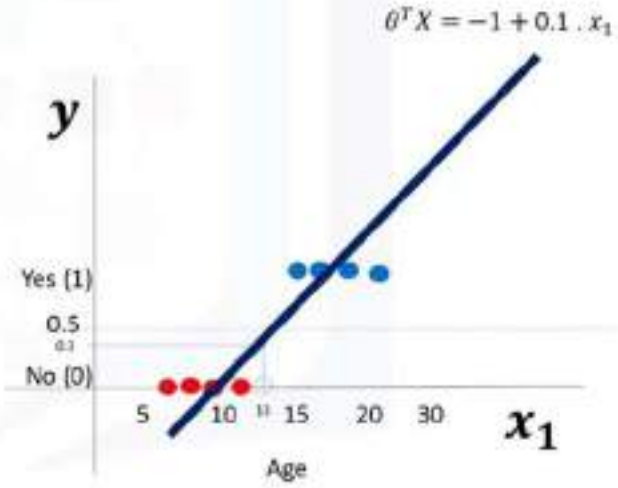


Linear regression in classification problems?

$\theta^T X = \theta_0 + \theta_1 x_1$

$p_1 = [13]$ → $\theta^T X = -1 + 0.1 \,.\, x_1$
$= -1 + 0.1 \times 13$
$= 0.3$

$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$



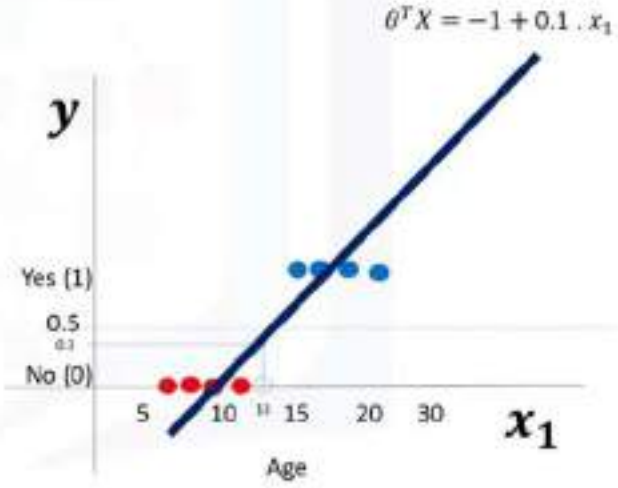
Linear regression in classification problems?

$\theta^T X = \theta_0 + \theta_1 x_1$

$p_1 = [13]$ → $\theta^T X = -1 + 0.1 \,.\, x_1$
$= -1 + 0.1 \times 13$
$= 0.3$

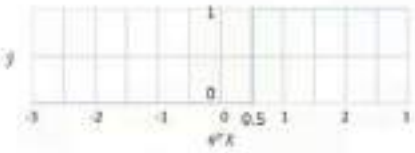
$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$

$\theta^T X = 0.3$
$\theta^T X < 0.5$ → Class 0



The problem with using linear regression

$\theta^T X = \theta_0 + \theta_1 x_1 + \cdots$

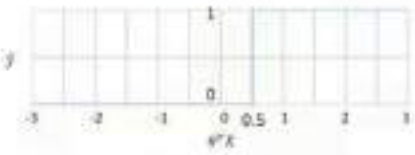


$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$

© 2018 IBM Corporation

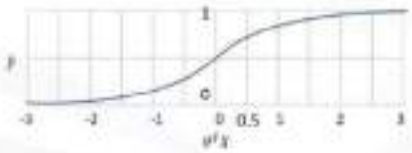
The problem with using linear regression

$\theta^T X = \theta_0 + \theta_1 x_1 + \cdots$



$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$

$\sigma(\theta^T X) = \sigma(\theta_0 + \theta_1 x_1 + \cdots)$



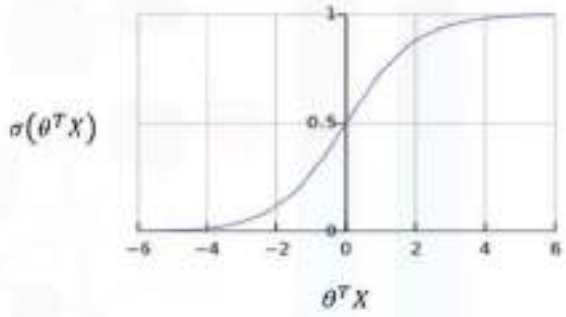
$$\hat{y} = \sigma(\theta^T X)$$

P(y=1|x)

© 2018 IBM Corporation

Sigmoid function in logistic regression

- Logistic Function

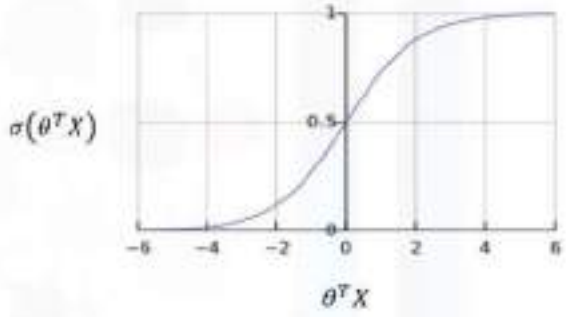


© 2018 IBM Corporation

Sigmoid function in logistic regression

- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T X}}$$



© 2018 IBM Corporation

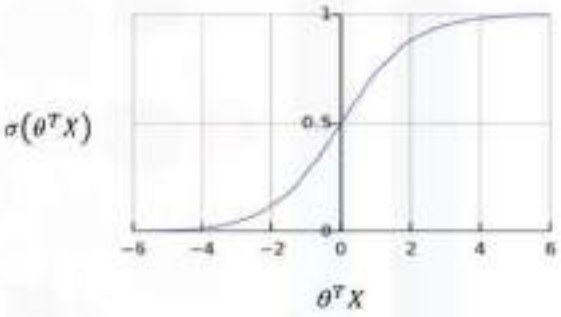
Sigmoid function in logistic regression

- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T X}}$$

$\sigma(\theta^T X) = 1$

$\sigma(\theta^T X) = 0$



© 2018 IBM Corporation

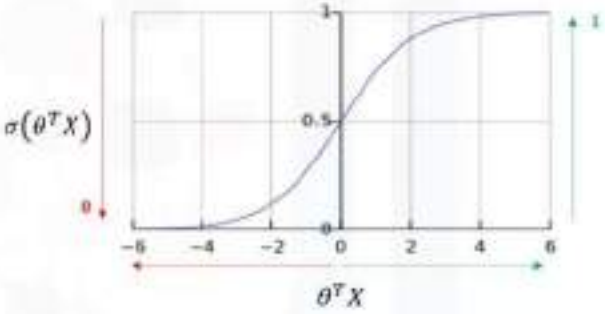
Sigmoid function in logistic regression

- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T X}}$$

$\sigma(\theta^T X) = 1$

$\sigma(\theta^T X) = 0$



© 2018 IBM Corporation

Sigmoid function in logistic regression

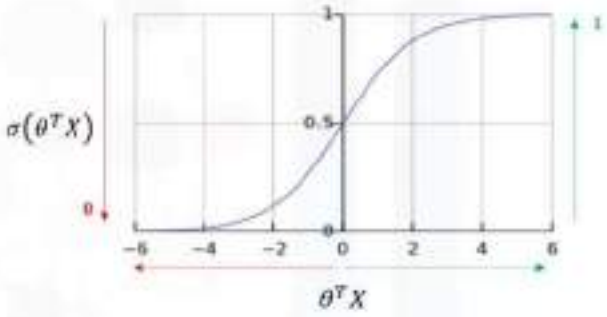
- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T x}}$$

$\sigma(\theta^T X) = 1$

$\sigma(\theta^T X) = 0$

$[0, 1]$



© 2018 IBM Corporation

Sigmoid function in logistic regression

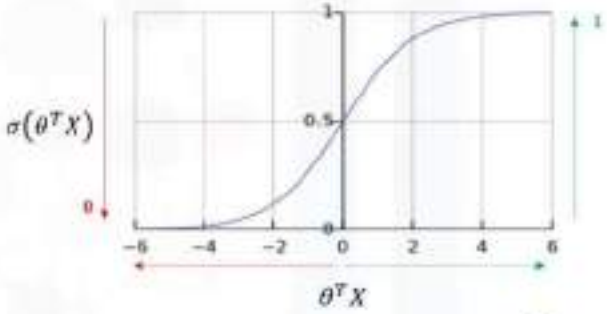
- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T x}}$$

$\sigma(\theta^T X) = 1$

$\sigma(\theta^T X) = 0$

$[0, 1]$



P(y=1|x)

© 2018 IBM Corporation

Sigmoid function in logistic regression

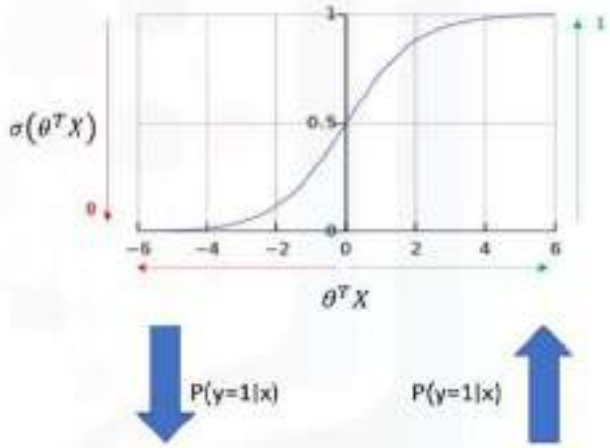
- Logistic Function

$$\sigma(\theta^T X) = \frac{1}{1 + e^{-\theta^T X}}$$

$\sigma(\theta^T X) = 1$

$\sigma(\theta^T X) = 0$

$[0, 1]$



© 2018 IBM Corporation

Clarification of the customer churn model

What is the output of our model?

Clarification of the customer churn model

What is the output of our model?

- P(Y=1|X)
- P(y=0|X) = 1 - P(y=1|x)

- P(Churn=1|income,age) = 0.8
- P(Churn=0|income,age) = 1 - 0.8 = 0.2

$$\sigma(\theta^T X) \longrightarrow \text{P(y=1|x)}$$

Clarification of the customer churn model

What is the output of our model?

- P(Y=1|X)
- P(y=0|X) = 1 - P(y=1|x)

- P(Churn=1|income,age) = 0.8
- P(Churn=0|income,age) = 1 - 0.8 = 0.2

$$\sigma(\theta^T X) \longrightarrow \text{P(y=1|x)}$$

$$1 - \sigma(\theta^T X) \longrightarrow \text{P(y=0|x)}$$

The training process

$\sigma(\theta^T X) \longrightarrow$ P(y=1|x)

1. Initialize θ.
2. Calculate $\hat{y} = \sigma(\theta^T X)$ for a customer.
3. Compare the output of $\hat{y}$ with actual output of customer, y, and record it as error.
4. Calculate the error for all customers.
5. Change the θ to reduce the cost.
6. Go back to step 2.

$\theta = [-1,2]$

$\hat{y} = \sigma([-1,2] \times [2,5]) = 0.7$

Error = 1-0.7 = 0.3

$Cost = J(\theta)$

θ_{new}

Logistic Regression - Training

Saeed Aghabozorgi

© IBM Corporation. All rights reserved.

General cost function

$\sigma(\theta^T X) \rightarrow$ **P(y=1|x)**

- Change the weight -> Reduce the cost
- Cost function

General cost function

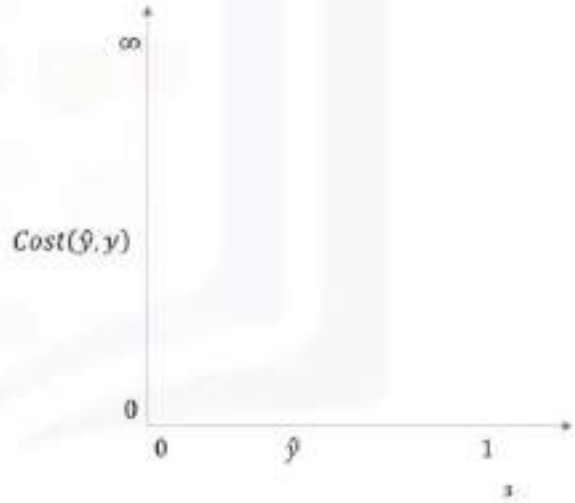
$\sigma(\theta^T X) \rightarrow$ **P(y=1|x)**

- Change the weight -> Reduce the cost
- Cost function

$$Cost(\vartheta, y) = \frac{1}{2}\left(\sigma(\theta^T X) - y\right)^2$$

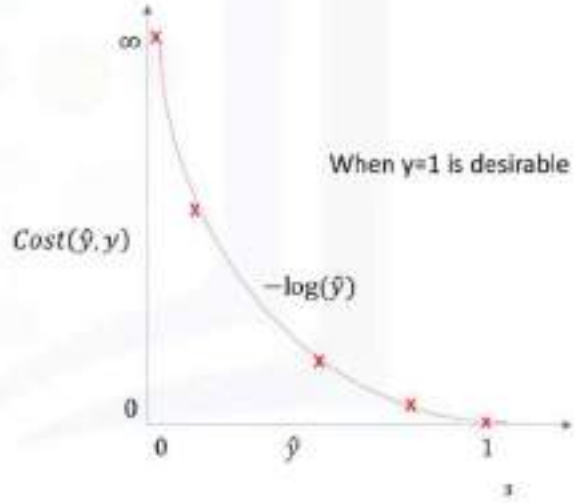
$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(\vartheta, y)$$

Plotting the cost function of the model



Plotting the cost function of the model

- Model $\hat{y}$
- Actual Value y=1 or 0
- If Y=1, and $\hat{y}$=1 → cost = 0
- If Y=1, and $\hat{y}$=0 → cost = large



Logistic regression cost function

- So, we will replace cost function with:

$$Cost(\hat{y}, y) = \frac{1}{2}\left(\sigma\left(\theta^T X\right) - y\right)^2$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(\hat{y}, y)$$

Logistic regression cost function

- So, we will replace cost function with:

$$Cost(\hat{y}, y) = \frac{1}{2}\left(\sigma\left(\theta^T X\right) - y\right)^2$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(\hat{y}, y)$$

$$Cost(\hat{y}, y) = \begin{cases} -\log(\hat{y}) & if\ y = 1 \end{cases}$$

Logistic regression cost function

- So, we will replace cost function with:

$$Cost(\hat{y}, y) = \frac{1}{2}\left(\sigma\left(\theta^T X\right) - y\right)^2$$

$$Cost(\hat{y}, y) = \begin{cases} -\log(\hat{y}) & if\ y = 1 \\ -\log(1 - \hat{y}) & if\ y = 0 \end{cases}$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(\hat{y}, y)$$

Logistic regression cost function

- So, we will replace cost function with:

$$Cost(\hat{y}, y) = \frac{1}{2}\left(\sigma\left(\theta^T X\right) - y\right)^2$$

$$Cost(\hat{y}, y) = \begin{cases} -\log(\hat{y}) & if\ y = 1 \\ -\log(1 - \hat{y}) & if\ y = 0 \end{cases}$$

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost(\hat{y}, y)$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log\left(\hat{y}^i\right) + (1 - y^i)\log\left(1 - \hat{y}^i\right)$$

Minimizing the cost function of the model

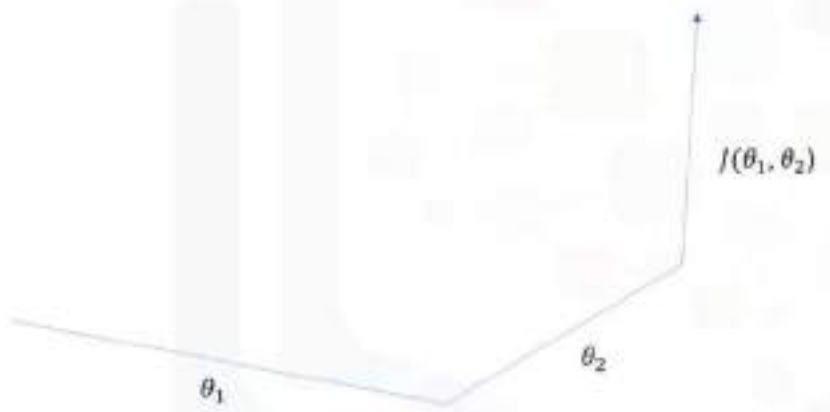
- How to find the best parameters for our model?
  - Minimize the cost function
- How to minimize the cost function?
  - Using Gradient Descent
- What is gradient descent?
  - A technique to use the derivative of a cost function to change the parameter values, in order to minimize the cost

Using gradient descent to minimize the cost

$$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

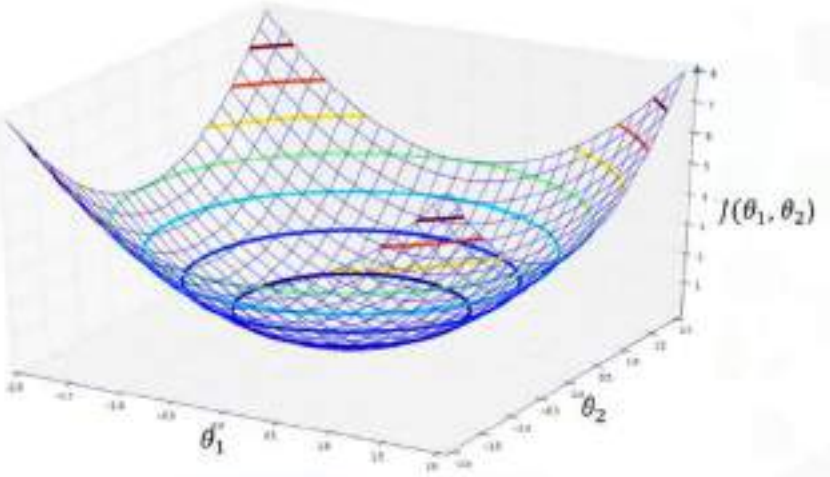
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

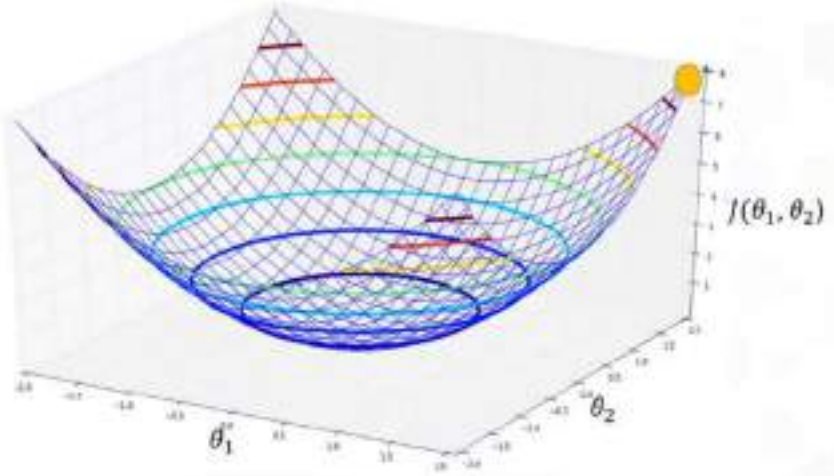
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

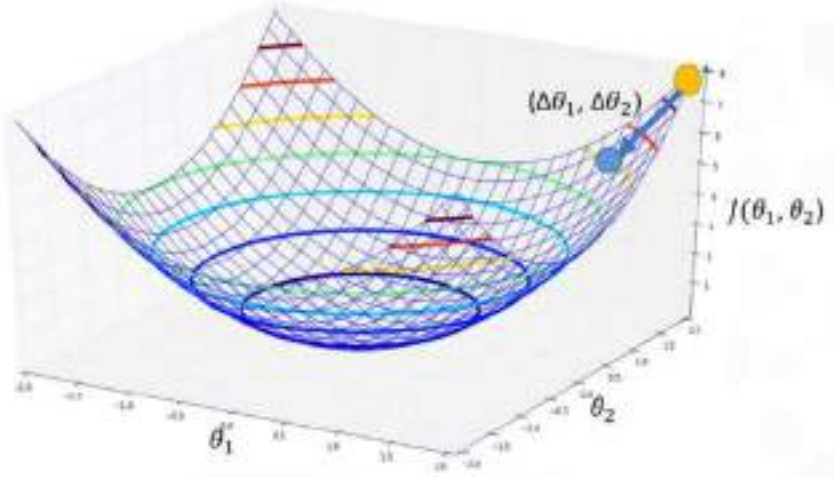
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

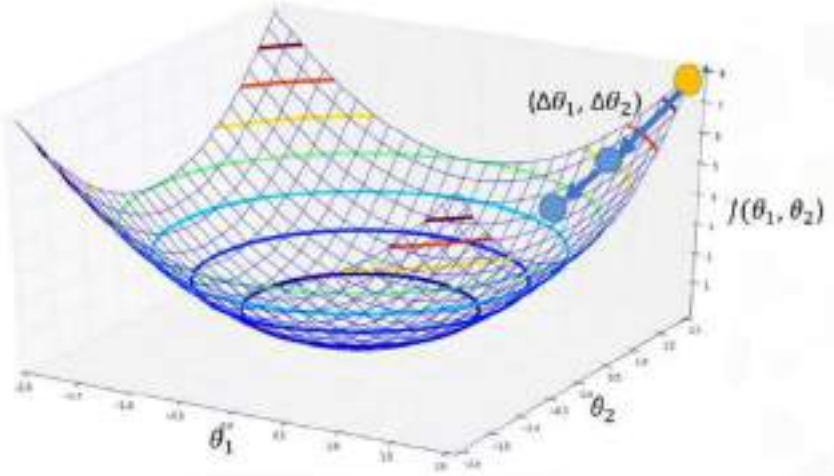
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

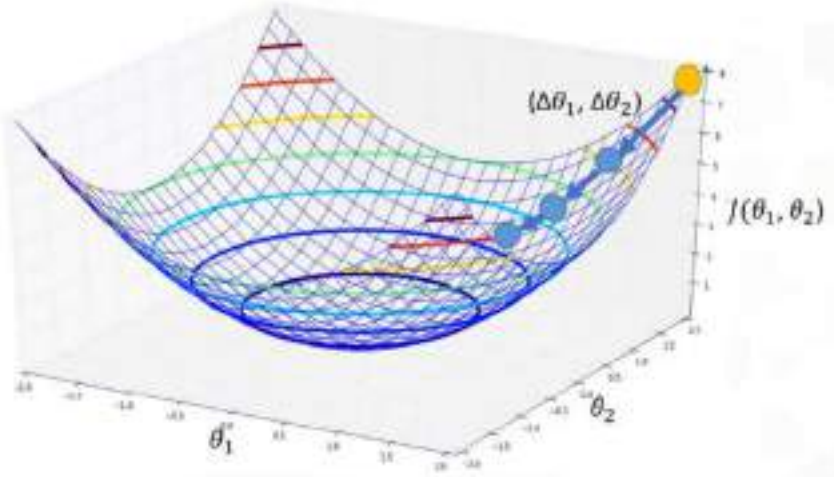
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

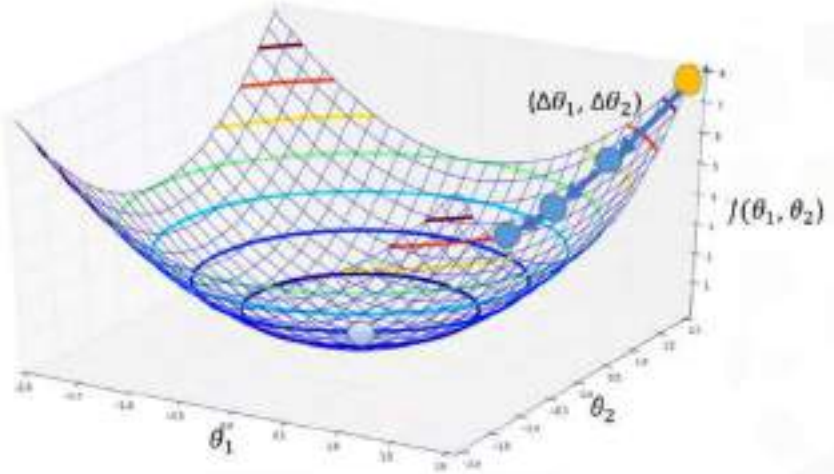
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

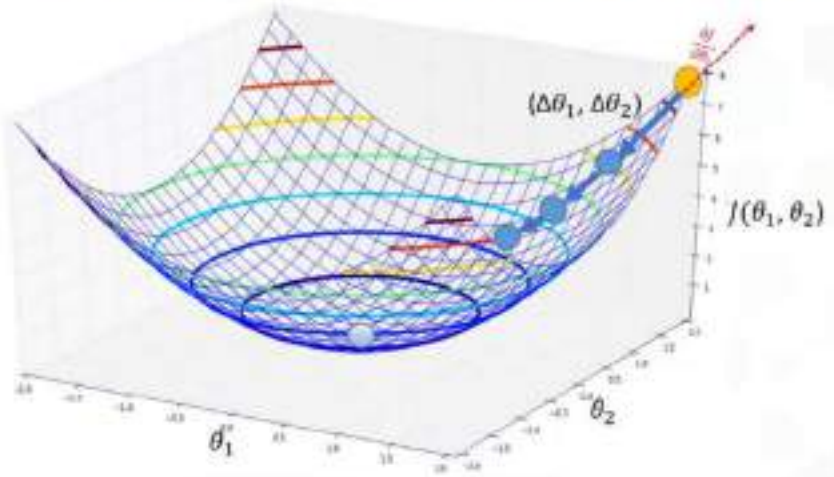
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

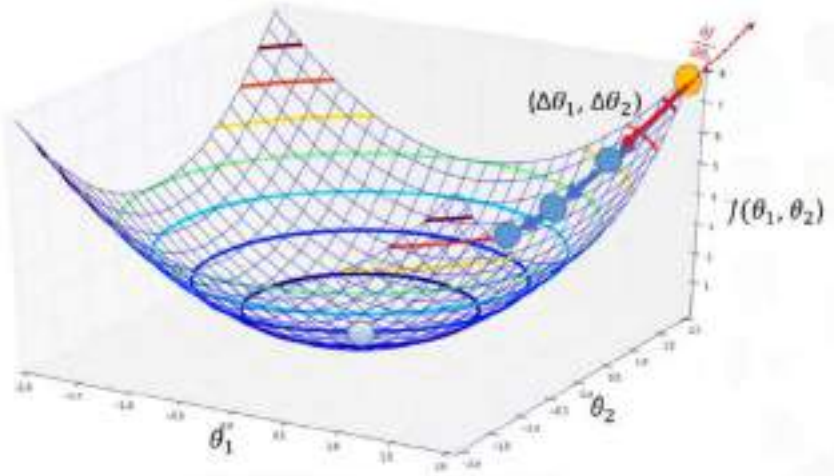
Using gradient descent to minimize the cost



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Using gradient descent to minimize the cost

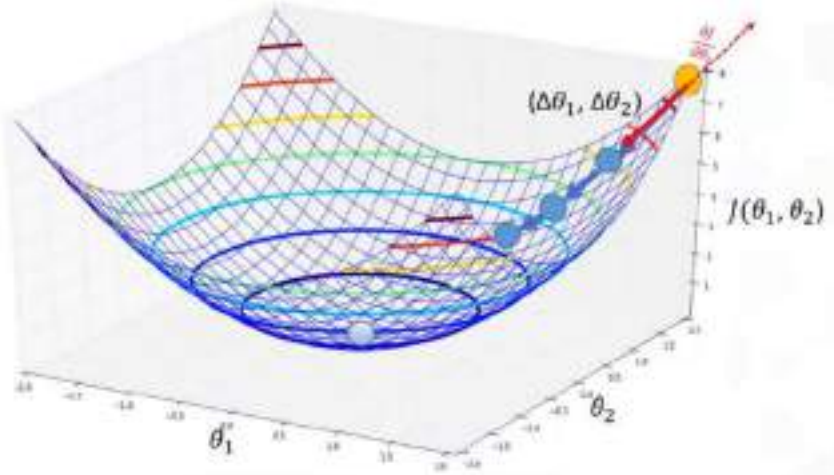


$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Using gradient descent to minimize the cost

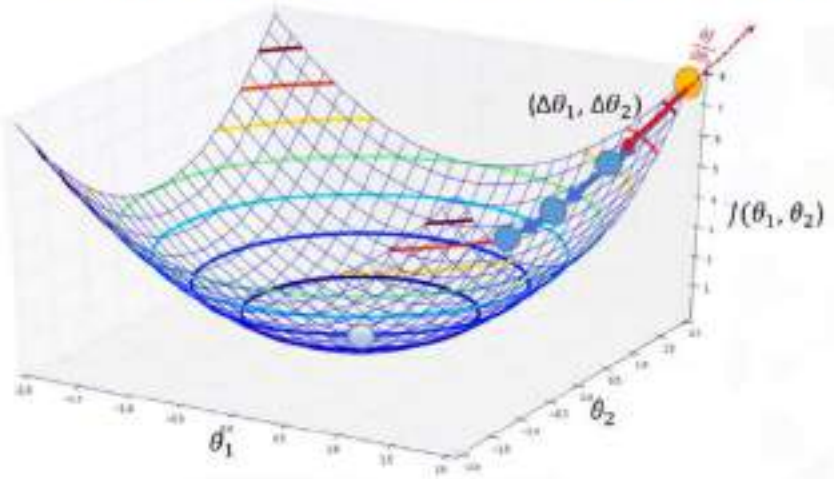
$$\frac{\partial J}{\partial \theta_1} = -\frac{1}{m}\sum_{i=1}^{m}(y^i - \hat{y}^i)\,\hat{y}^i(1-\hat{y}^i)\,x_1$$



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Using gradient descent to minimize the cost



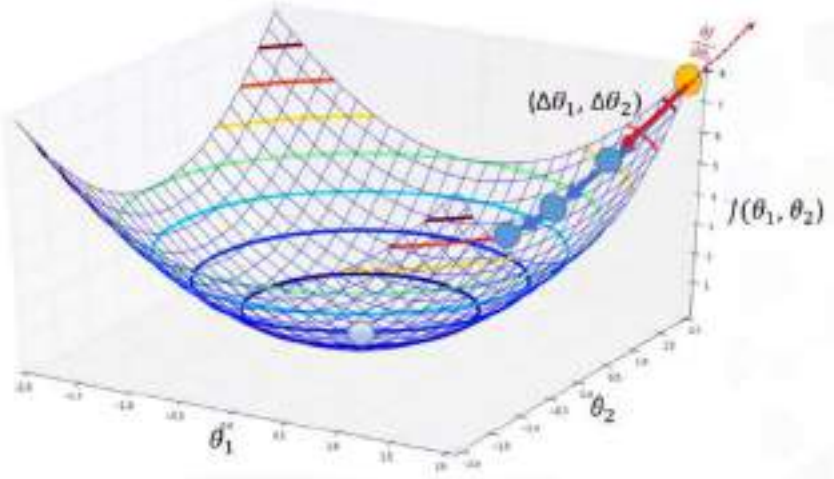
$$\frac{\partial J}{\partial \theta_1} = -\frac{1}{m}\sum_{i=1}^{m}(y^i - \hat{y}^i)\,\hat{y}^i(1-\hat{y}^i)\,x_1$$

$$\nabla J = \begin{bmatrix} \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \\ \frac{\partial J}{\partial \theta_3} \\ \cdots \\ \cdots \\ \frac{\partial J}{\partial \theta_k} \end{bmatrix}$$

$$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Using gradient descent to minimize the cost



$$\frac{\partial J}{\partial \theta_1} = -\frac{1}{m}\sum_{i=1}^{m}(y^i - \hat{y}^i)\,\hat{y}^i(1-\hat{y}^i)\,x_1$$

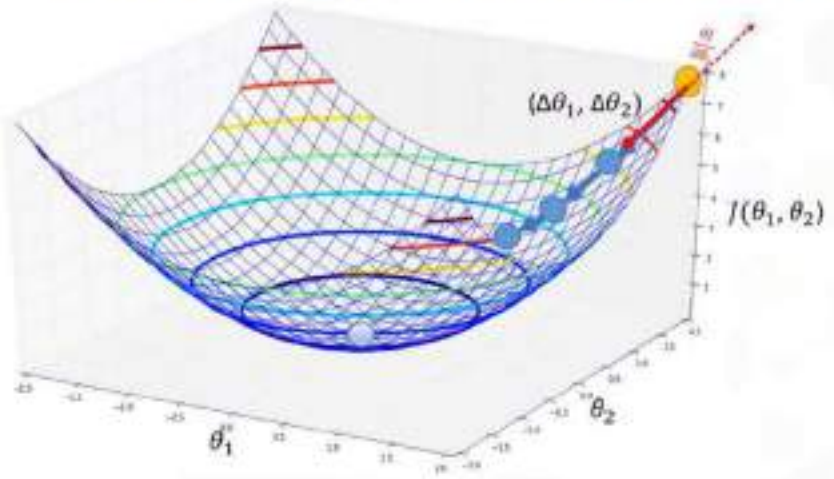
$$\nabla J = \begin{bmatrix} \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \\ \frac{\partial J}{\partial \theta_3} \\ \cdots \\ \cdots \\ \frac{\partial J}{\partial \theta_k} \end{bmatrix}$$

$$New\theta = old\theta - \nabla J$$

$$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Using gradient descent to minimize the cost



$$\frac{\partial J}{\partial \theta_1} = -\frac{1}{m}\sum_{i=1}^{m}(y^i - \hat{y}^i)\,\hat{y}^i(1-\hat{y}^i)\,x_1$$

$$\nabla J = \begin{bmatrix} \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \\ \frac{\partial J}{\partial \theta_3} \\ \cdots \\ \cdots \\ \frac{\partial J}{\partial \theta_k} \end{bmatrix}$$

$$New\theta = old\theta - \eta\,\nabla J$$

$$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Training algorithm recap

1. initialize the parameters randomly.
2. Feed the cost function with training set, and calculate the error.
3. Calculate the gradient of cost function.
4. Update weights with new values.
5. Go to step 2 until cost is small enough.
6. Predict the new customer X.

$$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

$$\nabla J = [\frac{\partial J}{\partial \theta_1}, \frac{\partial J}{\partial \theta_2}, \frac{\partial J}{\partial \theta_3}, \ldots, \frac{\partial J}{\partial \theta_k}]$$

$$\theta_{new} = \theta_{pre} - \eta \nabla J$$

$$\mathbf{P(y=1|x)} = \sigma(\theta^T X)$$

https://www.coursera.org/professional-certificates/ibm-data-science

coursera

This lecture was only a reduced version of the **Data Analysis in Python** course Offered by **IBM** on **Coursera**

Hacettepe Üniversitesi
Bilgisayar Mühendisliği Bölümü

BBM467 - Data Intensive Applications

Foundations of Clusters

Dr. Fuat Akal
akal@hacettepe.edu.tr

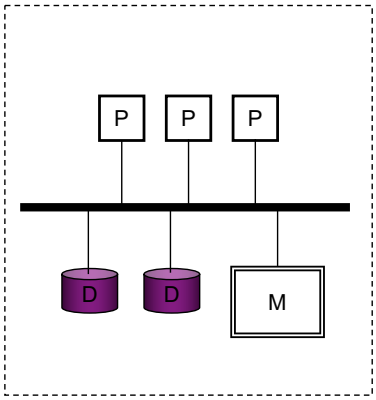
Foundations of Data[base] Clusters

- Database Clusters
- Hardware Architectures
- Data Design Schemes
- Replication Schemes
- Query Parallelism
- Logical Cluster Organization
- Replication Management

Database Clusters

- A cluster of computers can be thought as a single computing resource.
  - It utilizes multiple machines to provide a more powerful computing environment through a single system image.
- There are two types clusters
  - *high availability clusters* (HA)
  - *high performance computing clusters* (HPC)

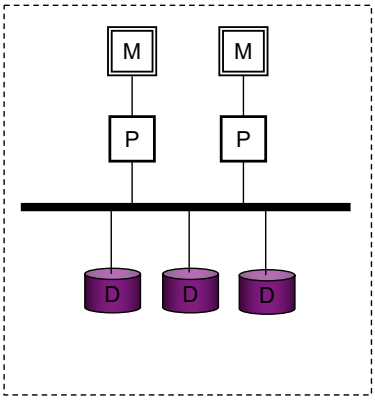
Hardware Architectures: Shared Memory



- All processors have access to the main memory and the disk, respectively.
- The processors are tightly coupled inside the same box and interconnected with a special switch.
- The interprocess communication is done by using a shared memory.
- The shared-memory approach presents simplicity and allows for load balancing as well as inter-query parallelism which comes for free.
- However, it is too expensive since it requires a special interconnect among the processors.
- Its performance and scalability are limited with the available memory and communication bandwidths.

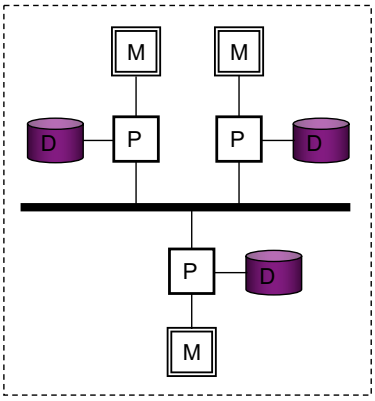
Hardware Architectures: Shared Disk

- In the shared-disk approach, all processors have their own memory, but they share disks.
- The interprocess communication occurs over a common high-speed bus.
- Provides high availability. All data is still accessible even when a node fails.
- Since each node has its own data cache, cache coherency must be maintained, e.g. by means of a lock manager, which results in reduced performance.
- Shared-disk systems have limited scalability due to bandwidth of the high-speed bus and potential bottlenecks of shared hardware.



Hardware Architectures: Shared Nothing

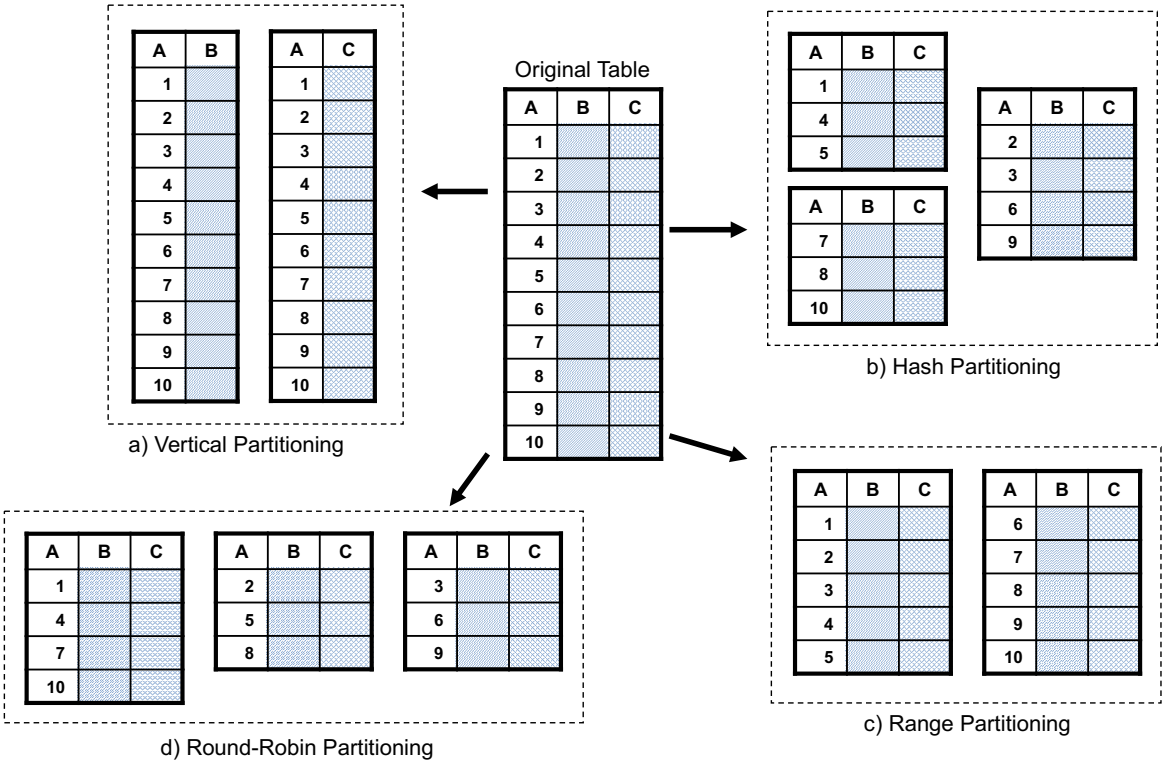
- In a shared-nothing architecture, each node is a complete stand-alone computer with its own memory and disk.
- The nodes are connected via switch or LAN. But, they do not share anything.
- The main advantages of such systems are very good scalability and high availability.
- However, the management of data is complicated and the programming with this model is harder due to importance of data partitioning and allocation.



Partitioning Schemes

- **Vertical Partitioning:** Vertical partitioning divides the columns of a table into separate tables.
  - Vertical partitioning makes projections and joins easier and helps optimizing access to the cache by reducing size of the tuples. However, access to the whole table may be required anyway, when executing queries.
- **Horizontal Partitioning:** Horizontal partitioning divides a table along its tuples. Its basic advantage is to allow parallel scans or projects.
  - The *hash partitioning* is based on a hash function that distributes the tuples according to a hashing key.
    - useful for parallel exact match queries and hash-join operations.
    - not appropriate for range queries and operations on other than partitioning keys.
  - The *range partitioning* is made based on value intervals of partitioning keys.
    - utilizes evaluations of range queries.
    - the performance of the range partitioning depends on the interval size.
  - The *round robin* partitioning technique distributes the tuples on each of the partitions. This approach is also called *striping*. The number of logically con-secutive tuples forms a striping unit.
    - The relative size of the striping unit directly affects the performance.
    - Small striping units result in more I/O parallelism for scans and long range queries.
    - Larger striping units, on the other hand, may cause latency to complete scans.

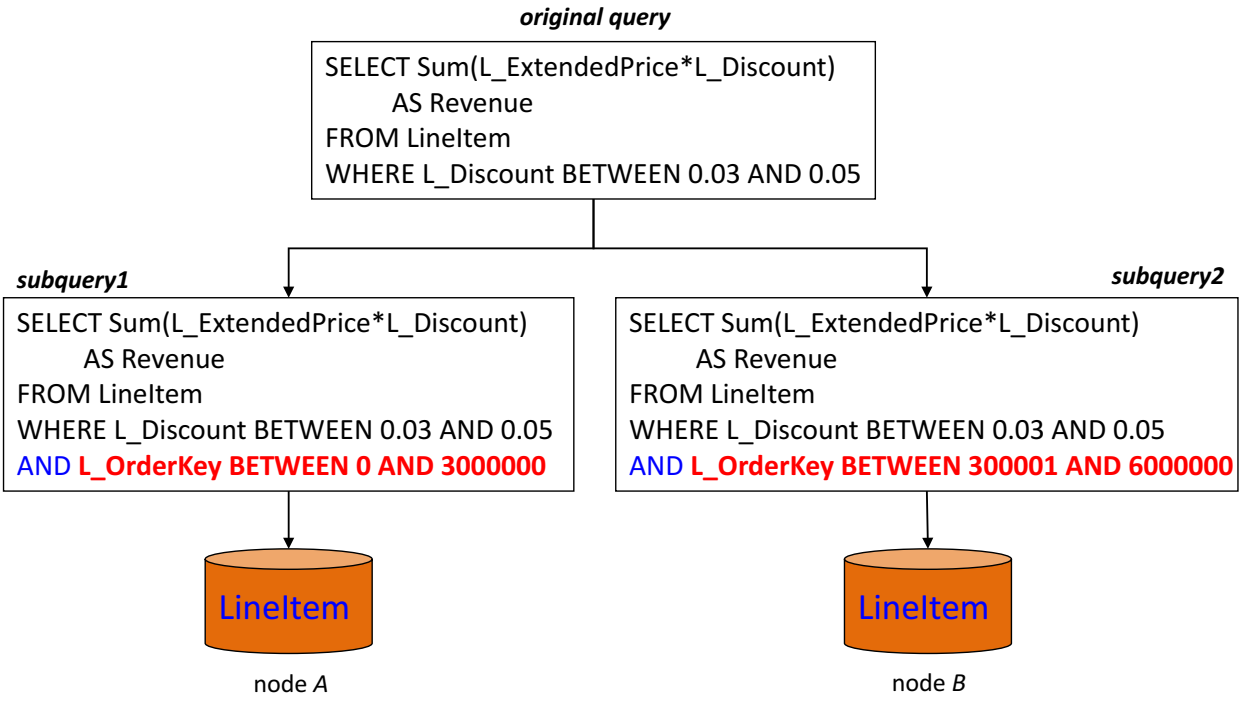
Partitioning Schemes



Virtual Partitioning

- Virtual partitioning, also called query partitioning, assumes that all tables are fully replicated on each cluster node.
- In this approach, a query is decomposed into subqueries which access small pieces of data by appending range predicates to the where clause of that query.
- Each subquery then deals with only a small part of the data.

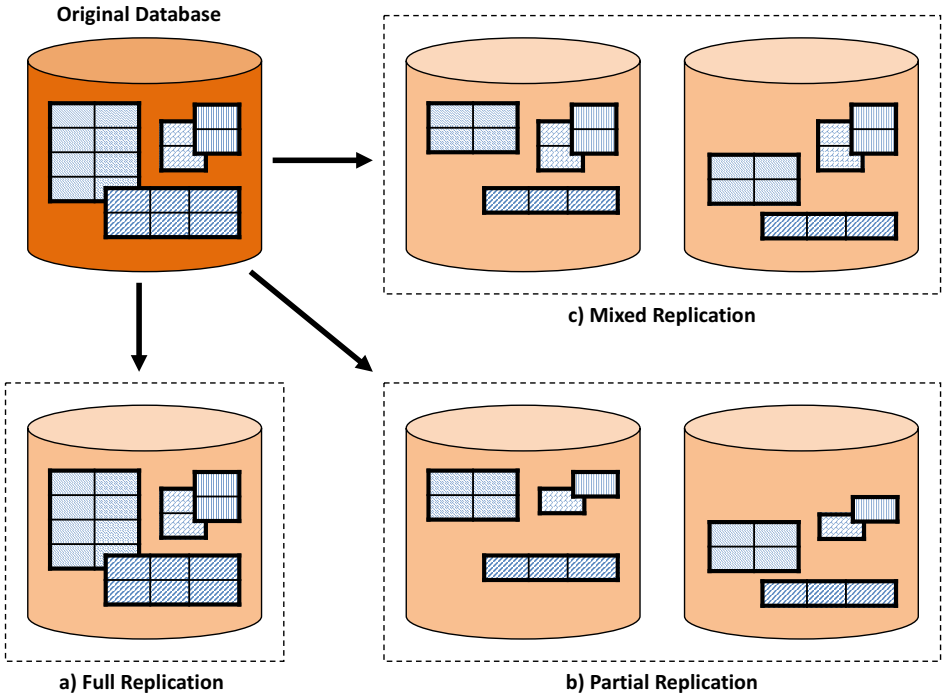
Virtual Partitioning (Example)



Replication Schemes

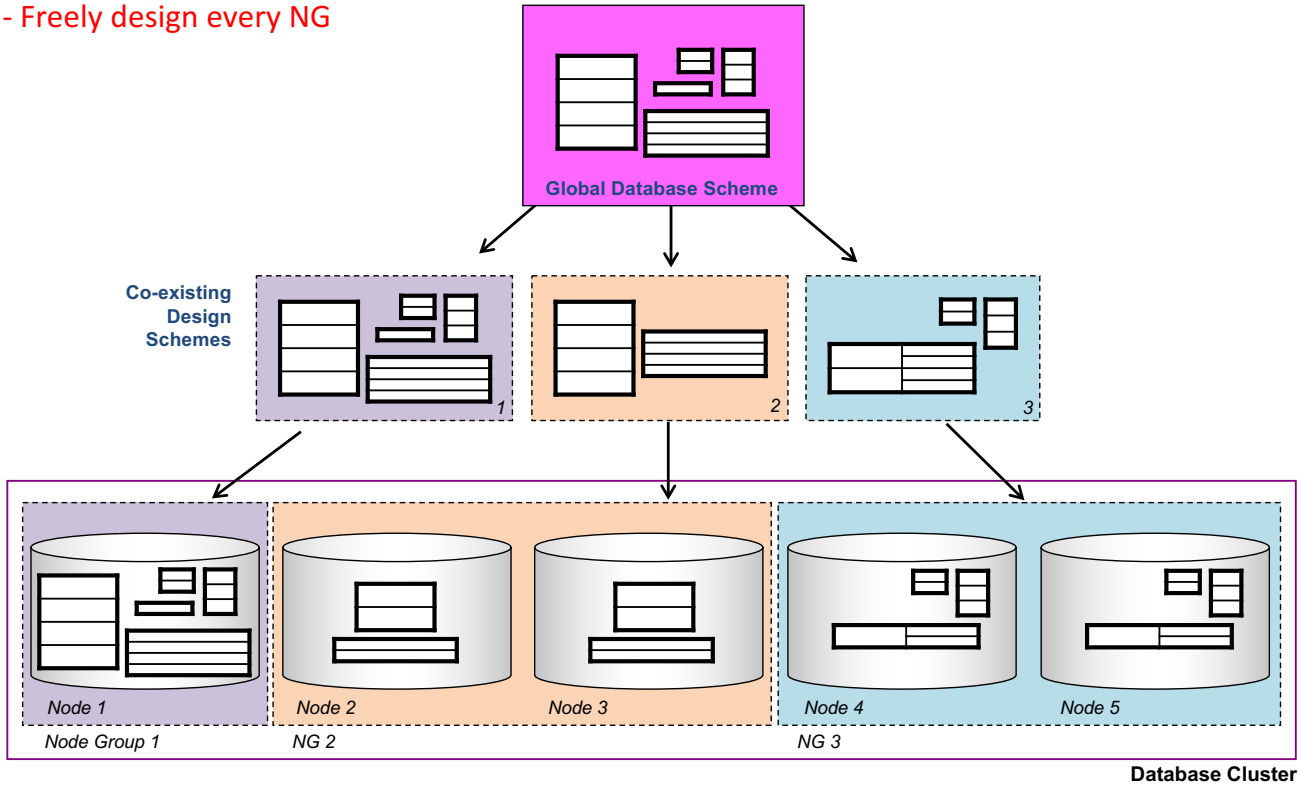
- **Full Replication:** Tables are duplicated on each cluster node. That is, each node holds an exact copy of the original database.
- **Partial Replication:** Partial replication means that only parts of original database are replicated on the different cluster nodes.
- **Mixed Replication:** Both full and partial replication at the same time.

Replication Schemes



Mixed Data Design

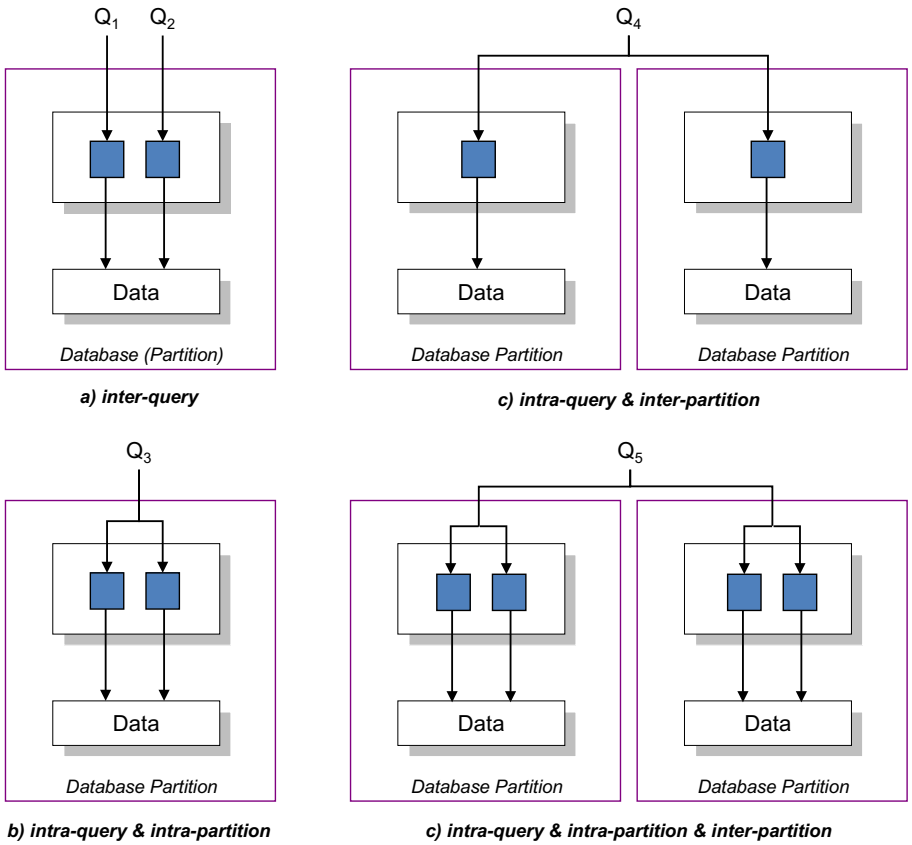
- Organize as node groups (NG)
- Freely design every NG



Query Parallelism in a Cluster

- **inter-query parallelism:** The capability of the database management system to accept queries from multiple users simultaneously. Each query is executed independently of the others.

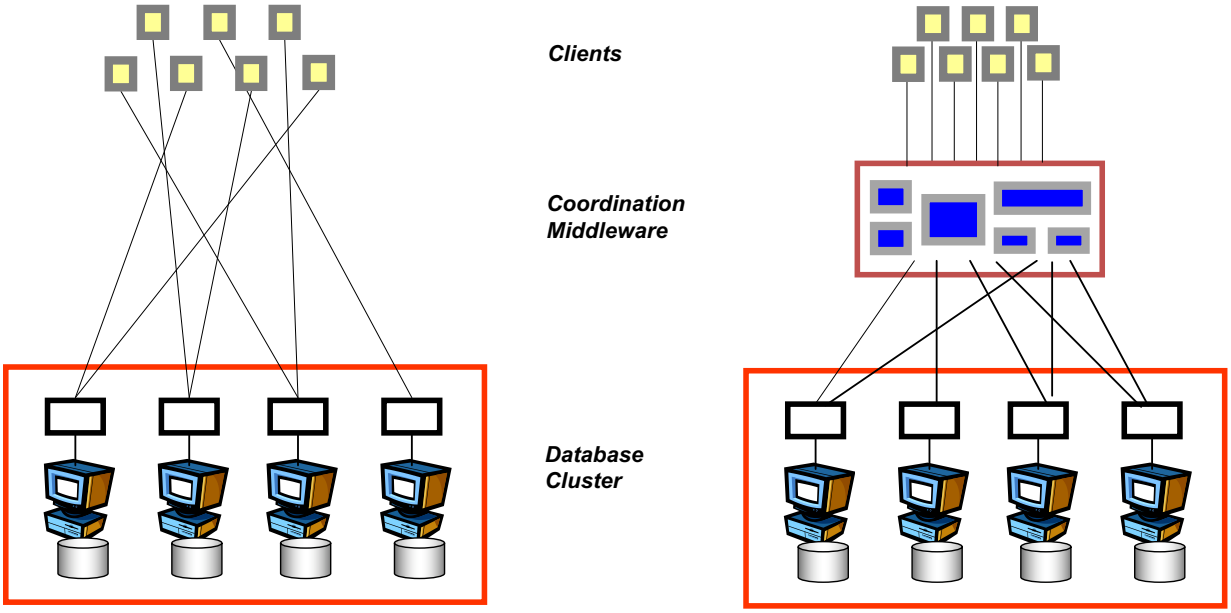
- **intra-query parallelism:** Achieved by decomposing queries into subqueries and evaluating them simultaneously.
  - *inter-partition*, *intra-partition* and *hybrid* parallelism



Logical Cluster Organization

- **Flat Cluster Architecture:** Allows any cluster node to be accessible by clients.
  - Forms a federated database of distinct databases running on independent servers.
    - Connected by a LAN, no resource sharing, such as disks.
  - Provides high availability and simple design.
    - Replication is difficult to implement with this model.
- **Middleware Based Cluster Architecture:** A client can only interact with the cluster through a coordination middleware.
  - The middleware is responsible for scheduling and routing of the clients requests.
  - The middleware has the knowledge about underlying cluster.
    - It can be used to ensure correct executions of concurrent updates and reads.
    - It also allows to improve overall throughput by choosing better components, e.g. with less load to perform client requests.
  - It is subject to single point of failure.
    - If the middleware fails, the cluster will become useless.
  - The middleware must be decentralized to improve scalability.

Logical Cluster Organization



a) flat architecture

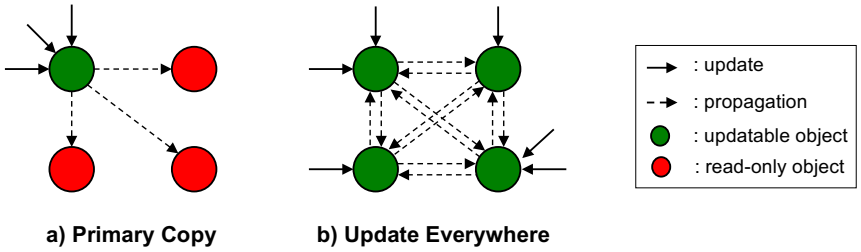
b) middleware-based architecture

Replication Management

- Replication is an essential technique to improve availability and scalability by fully or partially duplicating data objects among the nodes of a distributed system.
- Replication management is responsible for the maintenance of replicas and ensures consistency of multiple copies of the same data object residing on different nodes.
- That is, replication management is not simply copying data objects onto different nodes of a distributed system.

Synchronization of Updates

- There are two possibilities for the location of updates:
  - Updates can either be centralized on one primary copy
  - Or, be distributed on (a subset of) all replicas (update everywhere).

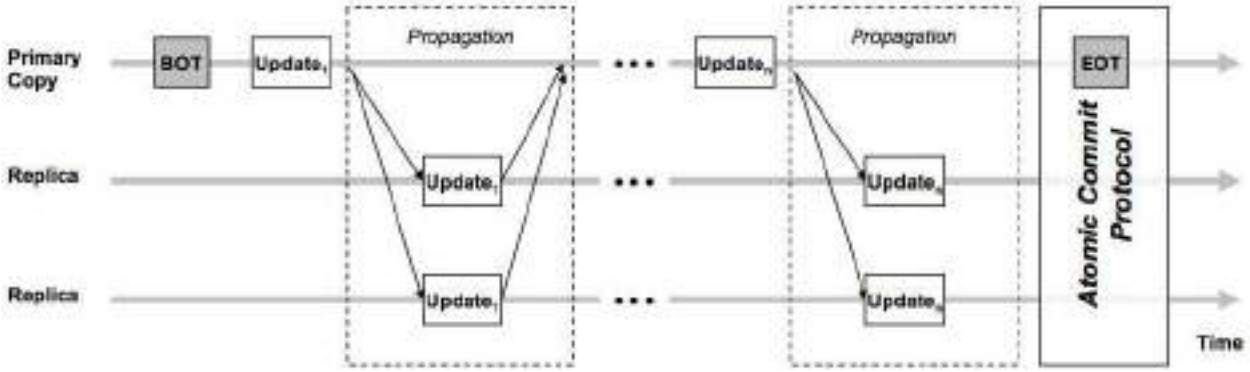


- Synchronization of updates can be done in two ways: *eager* and *lazy*

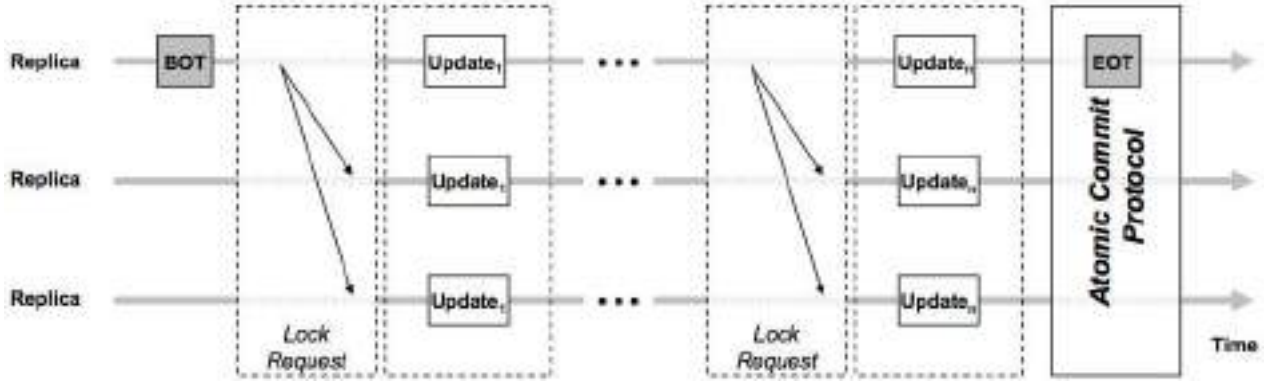
Synchronization of Updates

- Eager (or synchronous) replication.
  - All copies of an object are synchronized within the same database transaction.
  - Allows early detection of conflicts and presents a simple solution to provide consistency.
  - Has drawbacks regarding performance and due to the high communication overhead among the replicas and the high probability of deadlocks.
- Lazy (or asynchronous) replication.
  - Replica maintenance is decoupled from the original database transaction.
  - The transactions keeping the replicas up-to-date and consistent run as separate and independent database transactions after the original transaction has committed.
  - Compared to eager replication approaches, lazy approaches require additional efforts to guarantee serializable executions.

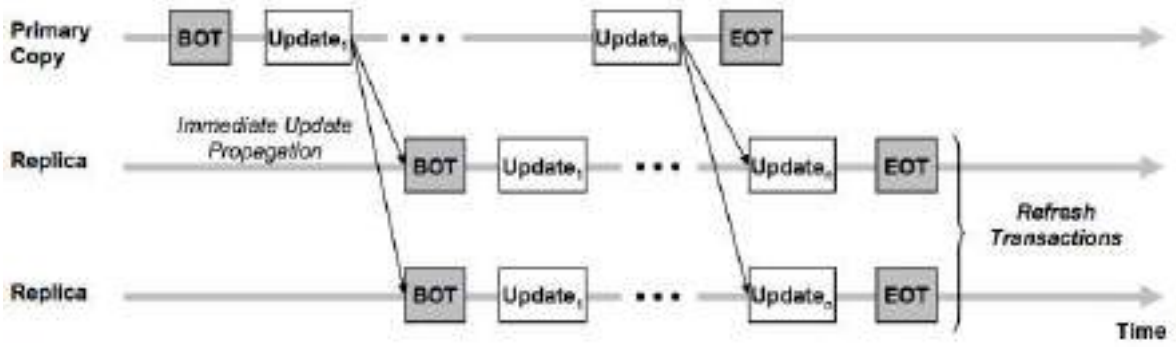
Eager Primary Copy Replication



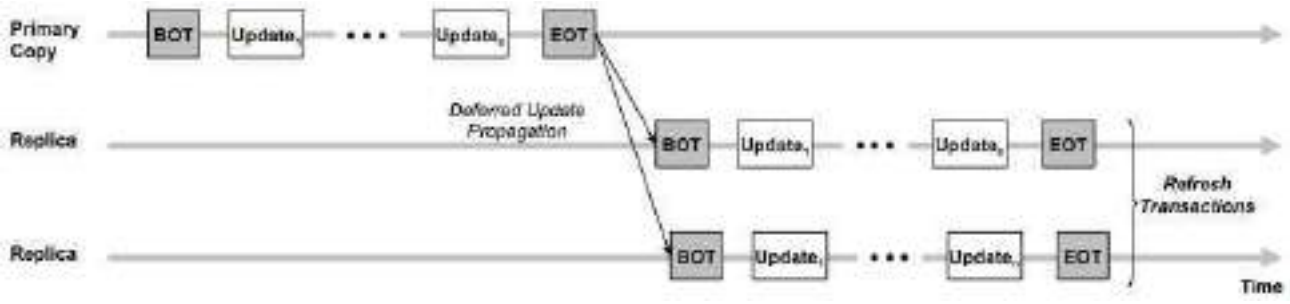
Eager Update Everywhere Replication



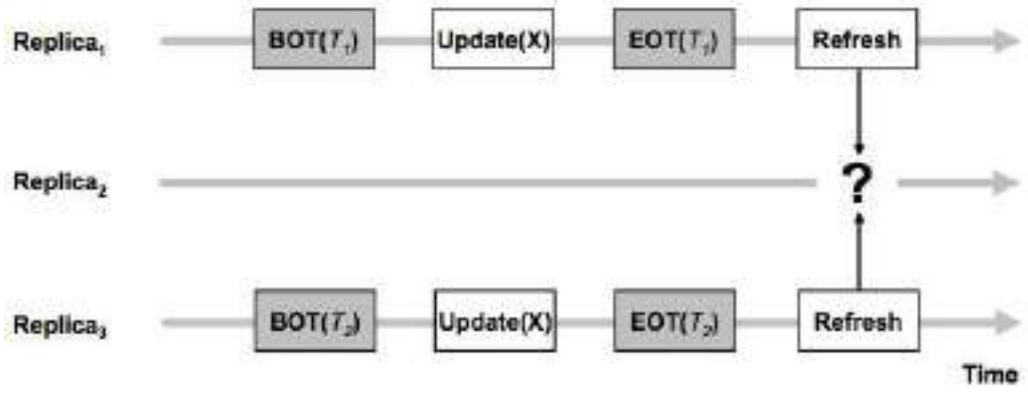
Lazy Primary Copy Replication with Immediate Updates



Lazy Primary Copy Replication with Deferred Updates



Lazy Update Everywhere Replication



Hacettepe Üniversitesi
Bilgisayar Mühendisliği Bölümü

BBM467 - Data Intensive Applications

Hadoop

Dr. Fuat Akal
akal@hacettepe.edu.tr

Problem

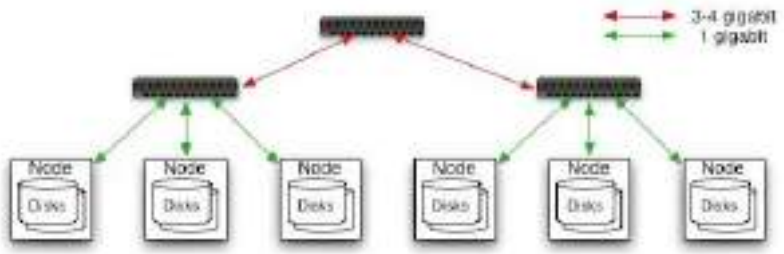
- How do you scale up applications?
  - Run jobs processing 100's of terabytes of data
  - Takes days to read on 1 computer
- Need lots of cheap computers
  - Fixes speed problem (15 minutes on 1000 computers), but…
  - Reliability problems
    - In large clusters, computers fail every day
      - 1 Computer, 1 failure in a year
      - 365 computer, 1 failure a day ☺
    - Cluster size is not fixed
- Need common infrastructure
  - Must be efficient and reliable

Solution: Hadoop

- Open Source Apache Project
  - An open-source software framework for storage and large scale processing of data-sets on clusters of commodity hardware.
- Hadoop Core includes:
  - Distributed File System - distributes data
  - Map/Reduce - distributes application
- Runs on
  - Linux, Mac OS/X, Windows, and Solaris
  - Commodity hardware

Recall: Commodity Hardware Cluster

- Typically in 2 level architecture
  - Nodes are commodity PCs
  - 40 nodes/rack
  - Uplink from rack is 8 gigabit
  - Rack-internal is 1 gigabit



What is Hadoop?

- Apache top level project, open-source implementation of frameworks for reliable, scalable, distributed computing and data storage.
- It is a flexible and highly-available architecture for large scale computation and data processing on a network of commodity hardware.

What is Hadoop Good for?

- Data-intensive text processing
- Assembly of large genomes
- Graph mining
- Machine learning and data mining
- Large scale social network analysis
- …

Who Uses Hadoop?

http://wiki.apache.org/hadoop/PoweredBy

Little Hadoop History

- Hadoop was created by Doug Cutting and Mike Cafarella in 2005.
- Cutting, who was working at Yahoo! at the time, named it after his son's toy elephant.
  - "Being a guy in the software business, we're always looking for names," Cutting said. "I'd been saving it for the right time."

Example Hadoop Cluster

- ~20,000 machines running Hadoop
- Largest clusters are currently 2000 nodes
- Several petabytes of user data
- Hundreds of thousands of jobs every month

Numbers are probably outdated and growing.

YAHOO! DEVELOPER NETWORK

MY PROJECTS TOOLS & SERVICES RESOURCES

YDN / Hadoop and Distributed Computing at Yahoo! / Hadoop Sorts a Petabyte in 16.25 Hours and a Terabyte in 62.5

Yahoo! Developer Network Blog

« Previous | Main | Next »

MAY 11, 2009

Hadoop Sorts a Petabyte in 16.25 Hours and a Terabyte in 62 Seconds

We used **Apache Hadoop** to compete in **Jim Gray's Sort** benchmark. Jim's Gray's sort benchmark consists of a set of many related benchmarks, each with their own rules. All of the sort benchmarks measure the time to sort different numbers of 100 byte records. The first 10 bytes of each record is the key and the rest is the value. The **minute sort** must finish end to end in less than a minute. The **Gray sort** must sort more than 100 terabytes and must run for at least an hour. The best times we observed were:

Bytes	Nodes	Maps	Reduces	Replication	Time
500,000,000,000	1406	8000	2600	1	59 seconds
1,000,000,000,000	1460	8000	2700	1	62 seconds
100,000,000,000,000	3452	190,000	10,000	2	173 minutes
1,000,000,000,000,000	3658	80,000	20,000	2	975 minutes

Distributed Systems

- Allows developers to use multiple machines for a single task
- Programming on a distributed system is much more complex
  - Synchronizing data exchanges
  - Managing a finite bandwidth
  - Controlling computation timing is complicated
- Distributed systems must be designed with the expectation of failure

"A distributed system is one that *prevents you* from *working* because of the failure of a machine that *you* had never heard of." *Leslie Lamport*

Partial Failures

- Failure of a single component must not cause the failure of the entire system only a degradation of the application performance
- Failure should not result in the loss of any data
- If a component fails, it should be able to recover without restarting the entire system
- Component failure or recovery during a job must not affect the final output

Scalability

- Increasing resources should increase load capacity
- Increasing the load on the system should result in a graceful decline in performance for all jobs
  - Not system failure

Hadoop

- Based on work done by Google in the early 2000s
  - "The Google File System" in 2003
  - "MapReduce: Simplified Data Processing on Large Clusters" in 2004
- The core idea was to distribute the data as it is initially stored
  - Each node can then perform computation on the data it stores without moving the data for the initial processing

Hadoop Ecosystem

Ambari™: A web-based tool for provisioning, managing, and monitoring Apache Hadoop.
Avro™: A data serialization system.
Cassandra™: A scalable multi-master database with no single points of failure.
Chukwa™: A data collection system for managing large distributed systems.
HBase™: A scalable, distributed database that supports structured data storage for large tables.
Hive™: A data warehouse infrastructure that provides data summarization and ad hoc querying.
Mahout™: A Scalable machine learning and data mining library.
Pig™: A high-level data-flow language and execution framework for parallel computation.
ZooKeeper™: A high-performance coordination service for distributed applications.

Hadoop Components

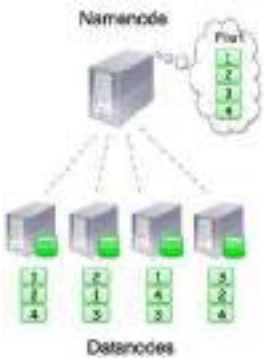
- Hadoop Distributed file system (HDFS)
  - Single namespace for entire cluster
  - Replicates data for fault-tolerance
- MapReduce framework
  - Executes user jobs specified as “map” and “reduce” functions
  - Manages work distribution & fault-tolerance

HDFS

- The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware.
  - HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware.
  - HDFS provides high throughput access to application data and is suitable for applications that have large data sets.

HDFS Pursues a master-slave Model

- NameNode
  - Executes file system namespace operations like opening, closing, and renaming files and directories.
  - Determines the mapping of blocks to DataNodes.
- DataNodes
  - Manage attached storage.
  - Internally, a file is split into one or more blocks and these blocks are stored in a set of DataNodes.
  - The DataNodes are responsible for serving read and write requests from the file system's clients.
  - The DataNodes also perform block creation, deletion, and replication upon instruction from the NameNode.



Block Placement

- Files are split into fixed sized blocks and stored on data nodes (Default 64MB)
- Data blocks are replicated for fault tolerance and fast access (Default is 3)
- Where to put a given block by default?
  - **First copy** is written to the node creating the file (write affinity)
  - **Second copy** is written to a data node within the same rack
  - **Third copy** is written to a data node in a different rack

Replica Selection

- To minimize global bandwidth consumption and read latency, HDFS tries to satisfy a read request from a replica that is **closest to the reader**.
- If there exists a **replica on the same rack** as the reader node, then that replica is preferred to satisfy the read request.
- If an HDFS cluster spans multiple data centers, then a replica that is **resident in the local data center** is preferred over any remote replica.

Disk Failures

- Each DataNode sends a Heartbeat message to the NameNode periodically.
- The NameNode marks DataNodes without recent Heartbeats as dead and does not forward any new IO requests to them.
- Any data that was registered to a dead DataNode is not available to HDFS any more.
- DataNode death may cause the replication factor of some blocks to fall below their specified value.
- The NameNode constantly tracks which blocks need to be replicated and initiates replication whenever necessary.

Re-replication

- The necessity for re-replication may arise due to many reasons:
  - a DataNode may become unavailable.
  - a replica may become corrupted.
  - a hard disk on a DataNode may fail.
  - the replication factor of a file may be increased.

Cluster Rebalancing

- The HDFS architecture is compatible with data rebalancing schemes.
- A scheme might automatically move data from one DataNode to another if the free space on a DataNode falls below a certain threshold.
- In the event of a sudden high demand for a particular file, a scheme might dynamically create additional replicas and rebalance other data in the cluster.
  - These types of data rebalancing schemes are not yet implemented.

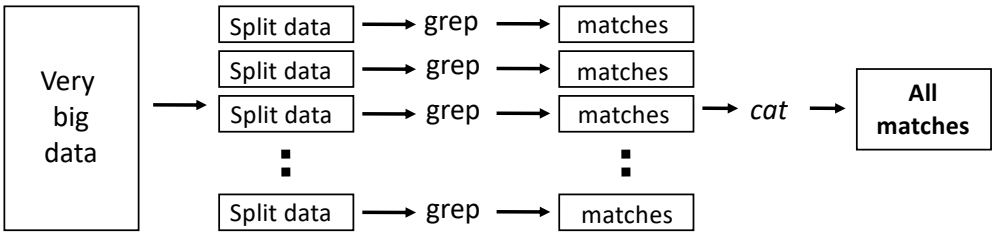
The Workflow

- Load data into the Cluster (HDFS writes)
- Analyze the data (MapReduce)
- Store results in the Cluster (HDFS)
- Read the results from the Cluster (HDFS reads)

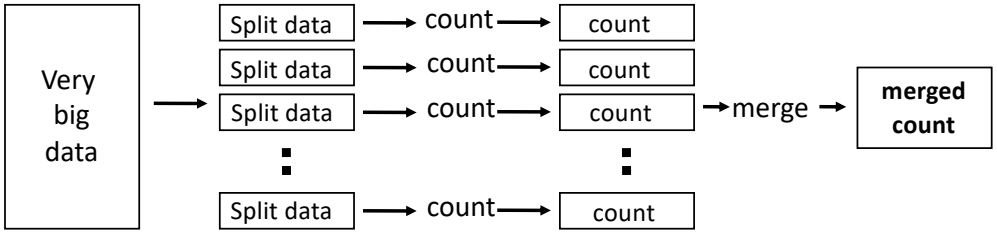
Motivation for MapReduce

- Scalability to large data volumes:
  - Scan 100 TB on 1 node @ 50 MB/s = 24 days
  - Scan on 1000-node cluster = 35 minutes
- Cost-efficiency
  - Commodity nodes (cheap, but unreliable)
  - Commodity network
  - Automatic fault-tolerance (fewer admins)
  - Easy to use (fewer programmers)
- MapReduce Architecture provides
  - Automatic Parallelization and Distribution
  - Fault Tolerance
  - I/O Scheduling
  - Monitoring and Status Updates

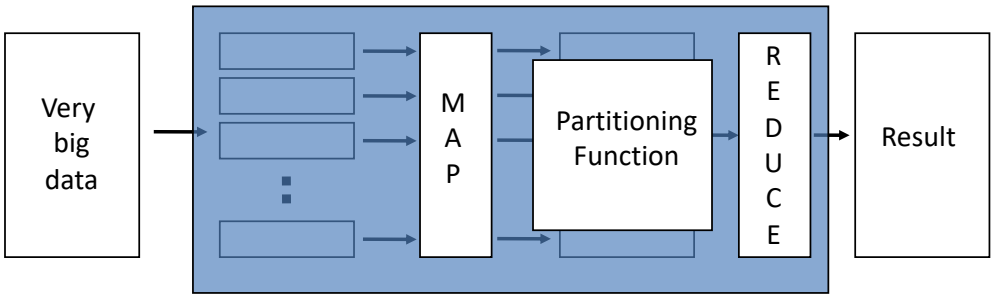
Distributed Grep



Distributed Word Count



Map + Reduce

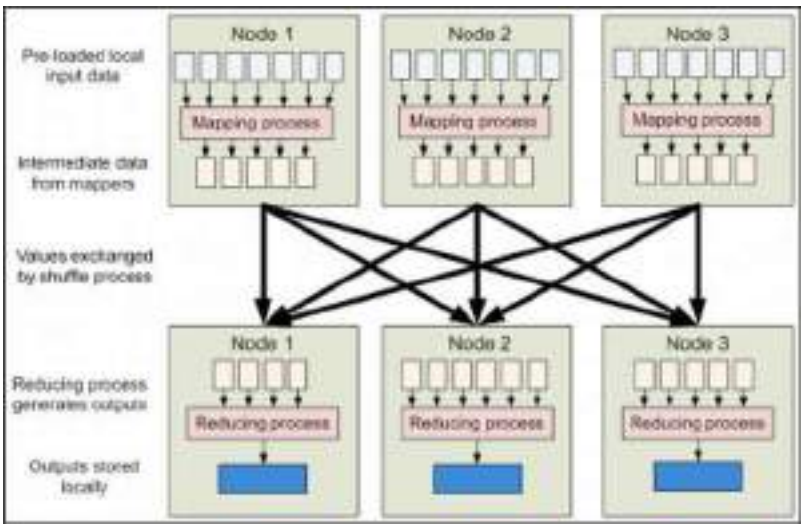


- Map:
  - Accepts *input* key/value pair
  - Emits *intermediate* key/value pair
- Reduce :
  - Accepts *intermediate* key/value* pair
  - Emits *output* key/value pair

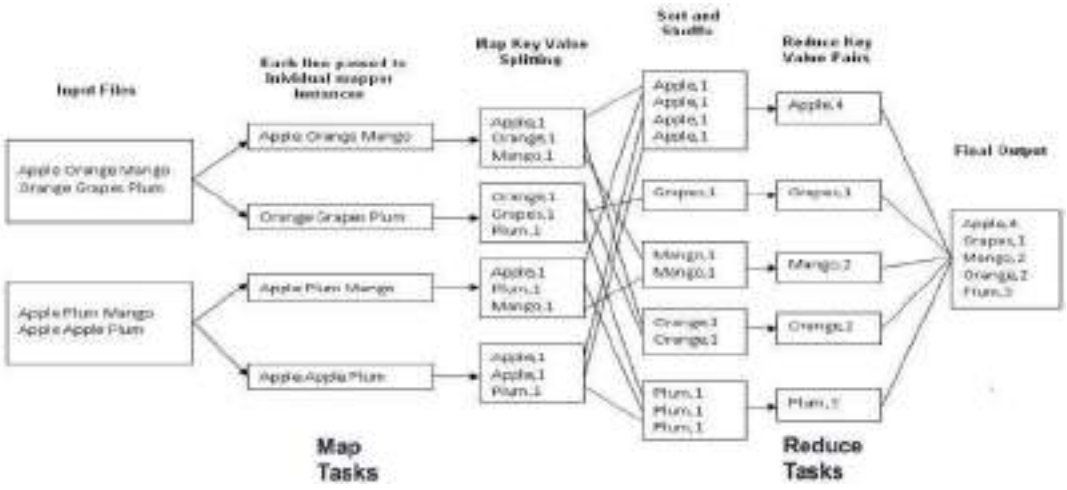
MapReduce Model

- Two primitive operations
  - **map**: $(k_1, v_1) \rightarrow list(k_2, v_2)$
  - **reduce**: $(k_2, list(v_2)) \rightarrow list(k_3, v_3)$
- Each map operation processes one input key/value pair and produces a set of key/value pairs
- Each reduce operation
  - Merges all intermediate values (produced by map ops) for a particular key
  - Produce final key/value pairs
- Operations are organized into tasks
  - Map tasks: apply map operation to a set of key/value pairs
  - Reduce tasks: apply reduce operation to intermediate key/value pairs
  - Each MapReduce job comprises a set of map and reduce (optional) tasks.

MapReduce Model



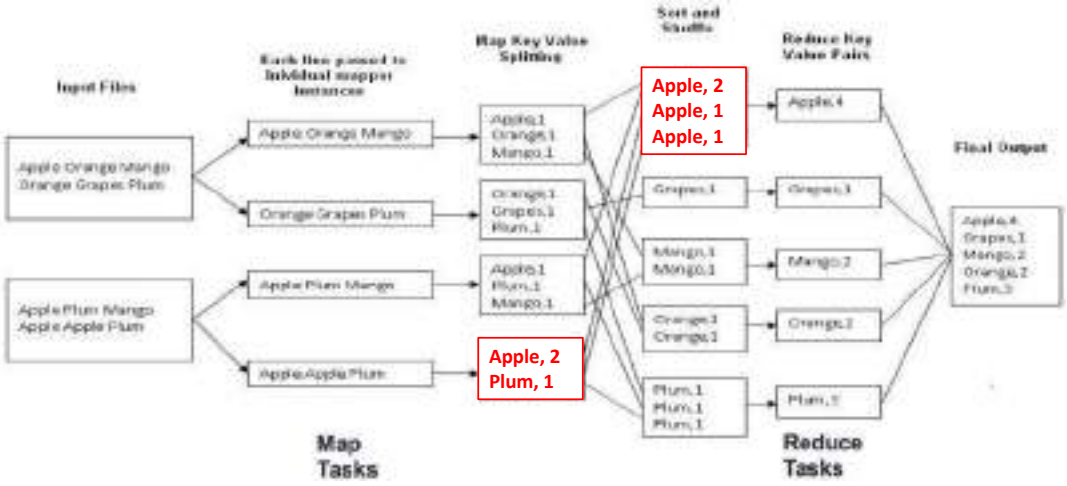
Word Count Example



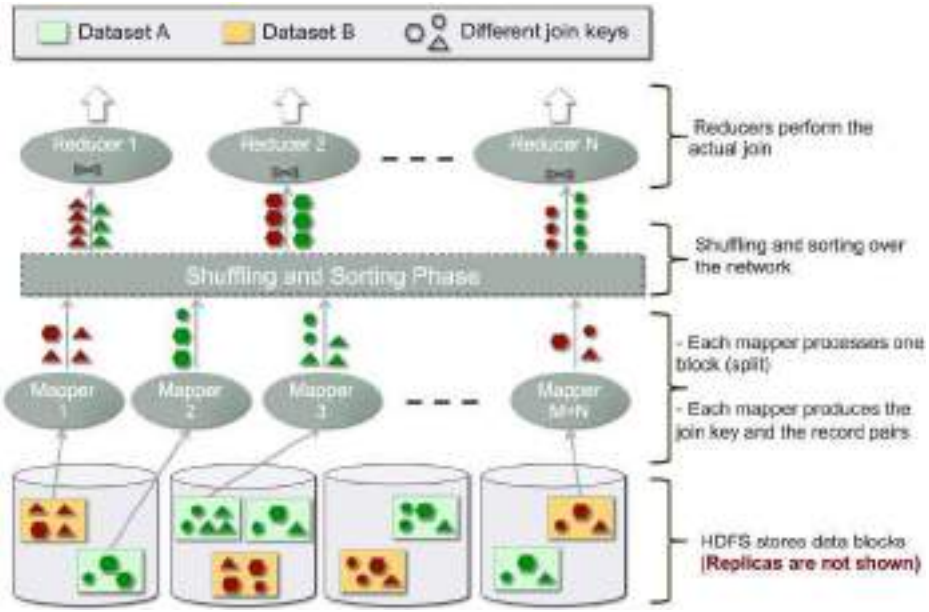
An Optimization: The Combiner

- Local aggregation function for repeated keys produced by same map
- For associative operators like sum, count, max
- Decreases size of intermediate data

Word Count Example (Combiner)



Joining Two Large Datasets



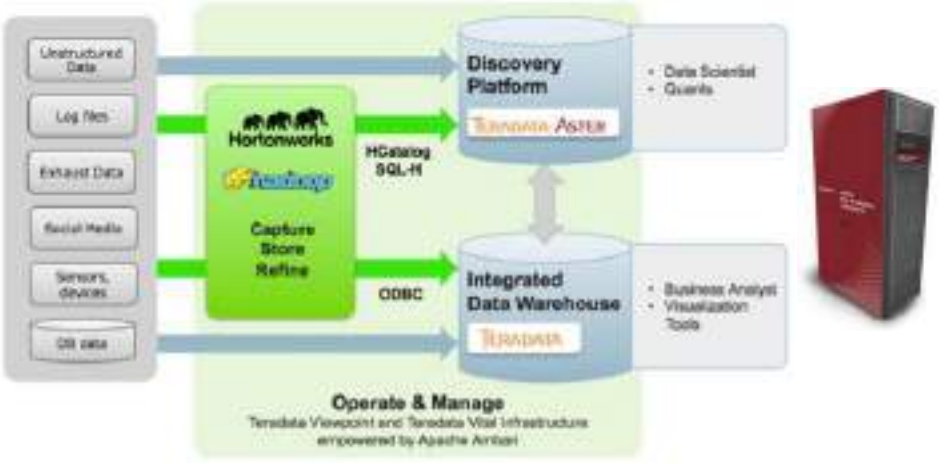
Fault Tolerance

- What if a task fails (map or reduce)?
  - Jobtracker re-schedules the task
- What if a datanode fails?
  - Both namenode and jobtracker detect the failure
  - All tasks on the failed node are re-scheduled
  - Namenode replicates the users' data to another node
- If a task is going slowly (straggler):
  - Launch second copy of task on another node
  - Take the output of whichever copy finishes first, and kill the other one

Oracle Big Data Appliance

http://www.cloudera.com

Teradata Aster Big Analytics Appliance



http://hortonworks.com/

Good Starting Point for Hadoop

- Hortonworks provides a sandbox which is provided as a self-contained virtual machine.
- No data center, no cloud service and no internet connection needed!

http://hortonworks.com/products/hortonworks-sandbox/

Apache Spark and Scalable Machine Learning

BBM467 Data Intensive Applications
Dr. Fuat Akal
akal@hacettepe.edu.tr

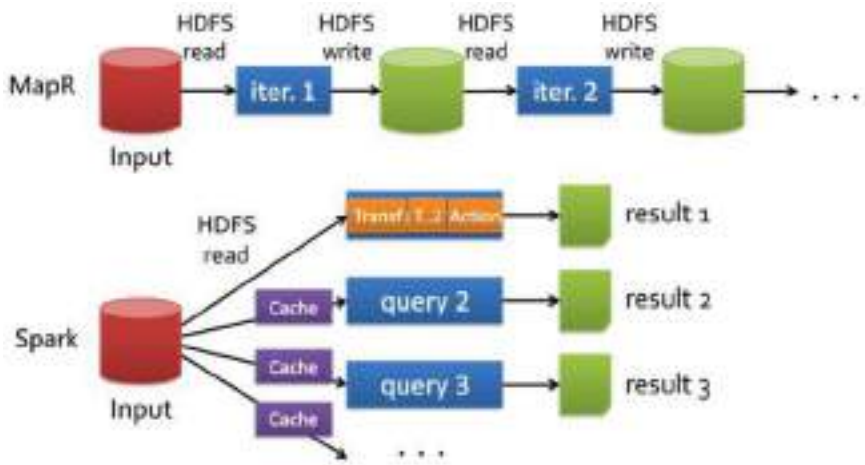
What is Apache Spark?

- Apache Spark is a cluster computing platform designed to be *fast* and *general- purpose*.
  - Spark extends the popular MapReduce model to efficiently support more types of computations, including interactive queries and stream processing.
    - One of the main features Spark offers for speed is the ability to run computations in memory.
  - Spark is designed to cover a wide range of workloads that pre- viously required separate distributed systems, including batch applications, iterative algorithms, interactive queries, and streaming.

Why Spark?

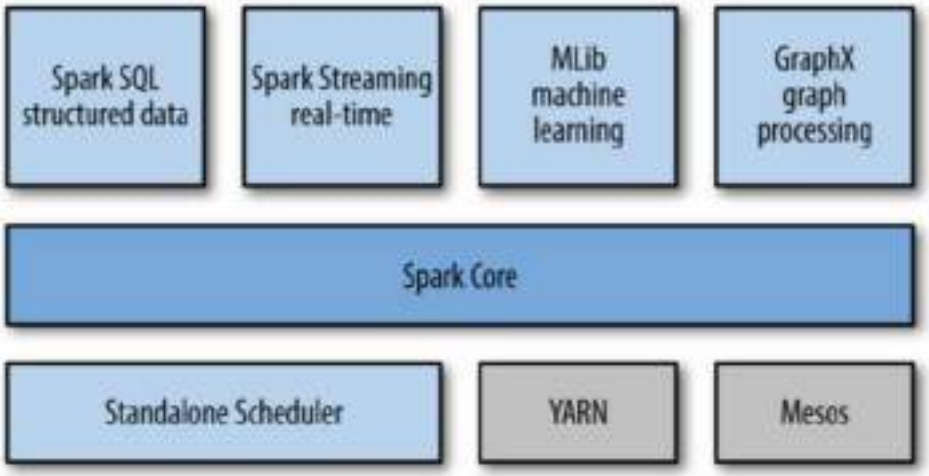
- Current popular programming models for clusters transform data flowing from stable storage to stable storage
- Acyclic data flow is a powerful abstraction, but is not efficient for applications that repeatedly reuse a *working set* of data

Spark vs. MapReduce



I/O and serialization can take **90%** of the time

The Spark Stack



Who is Using Spark?

- Data Science Tasks
  - While there is no standard definition, a *data scientist* is somebody whose main task is to analyze and model data.
  - Spark supports the different tasks of data science with a number of components.
  - Spark enables data scientists to tackle problems with larger data sizes than they could before with tools like R or Pandas.
- Data Processing Applications
  - Spark provides a simple way to parallelize engineering applications across clusters, and hides the complexity of distributed systems programming, network communication, and fault tolerance.

Functional Programming (FP)

- Mathematical concept of "Lambda Calculus"
- 1st implementation was LISP in the 50s
- Haskell is part of every computer science curriculum since the 90s
- Scala most recent representative
- Scala, Python, R and Java also rudimentary support FP

https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

Functions

```
var output = function (fn,x,y,z) { return fn(x,y,z); };

var sum = function (a,b,c )  { return (a+b+c); };

alert( output (sum,1,2,3) );
```

https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

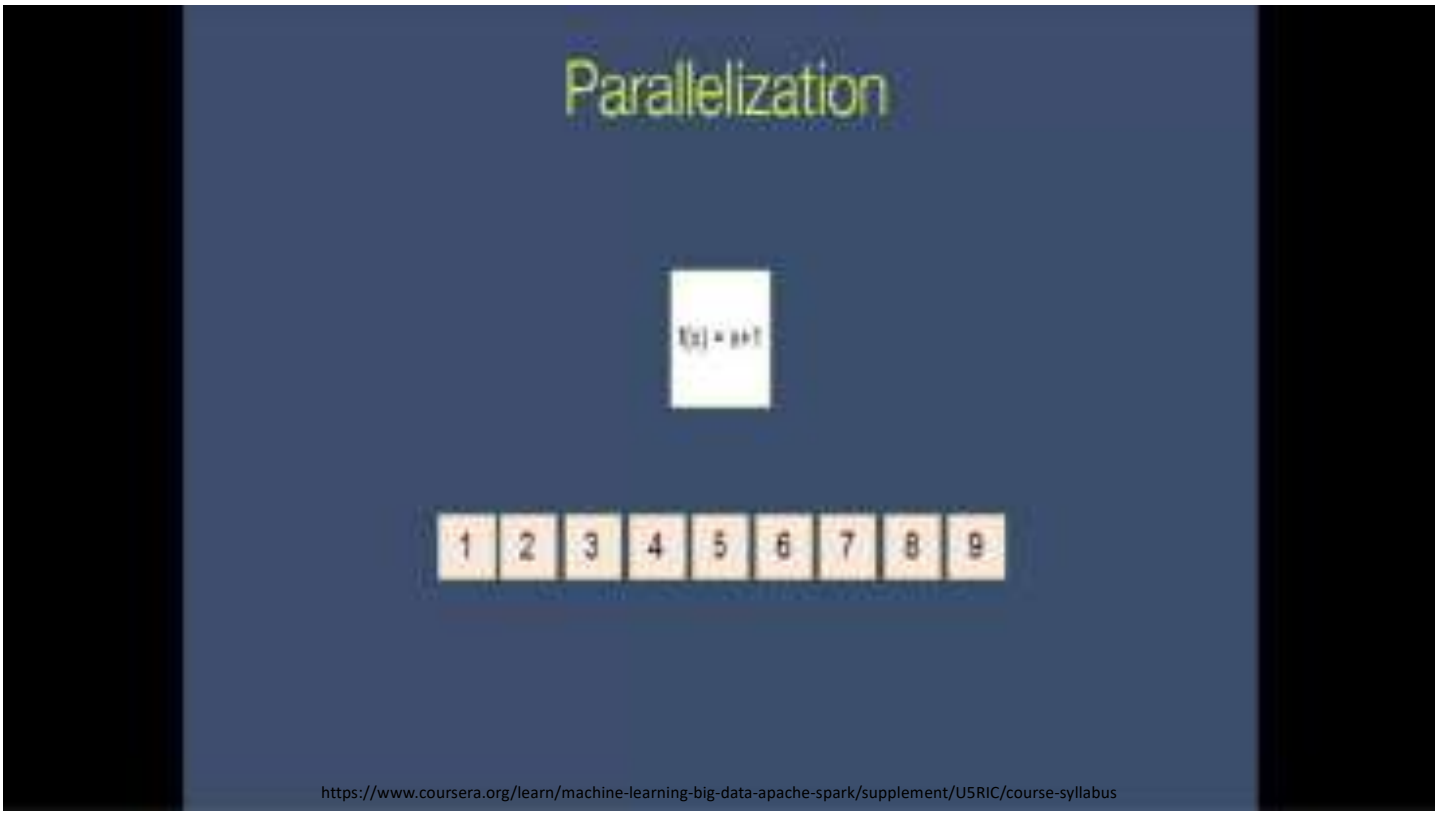
Functions

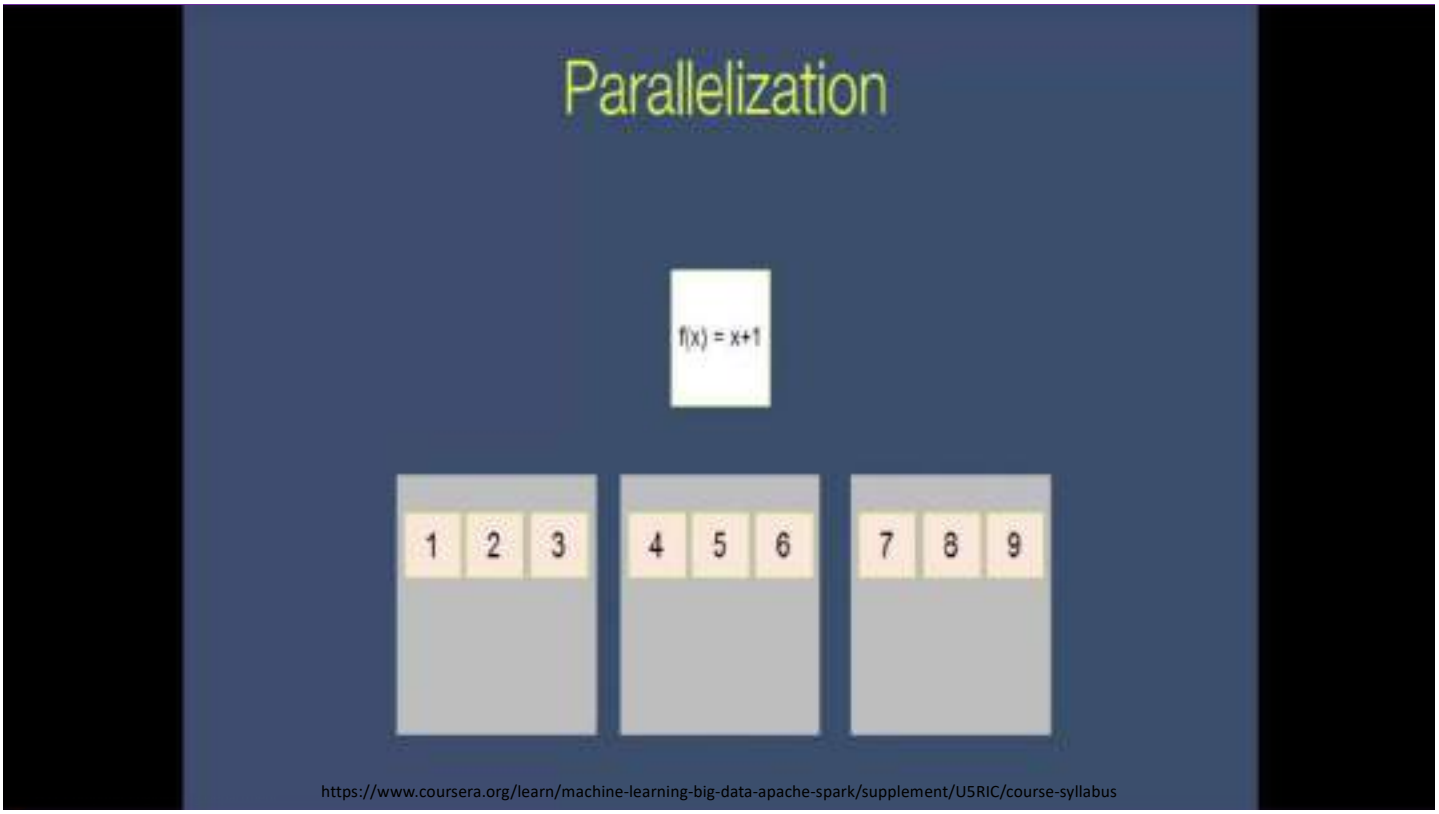
https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

Functions

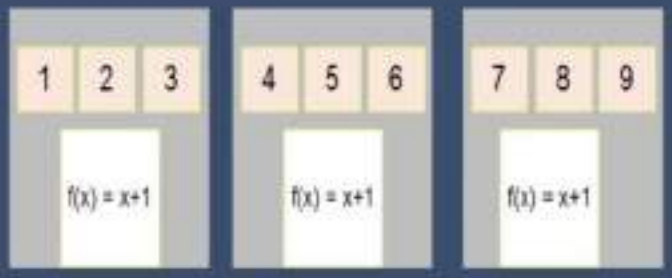
```
list = [1,2,3,4]
newList = list()
for i in list:
    newList.append(i+1)
```

https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus


Parallelization
f(x) = x+1
1 2 3 4 5 6 7 8 9
https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

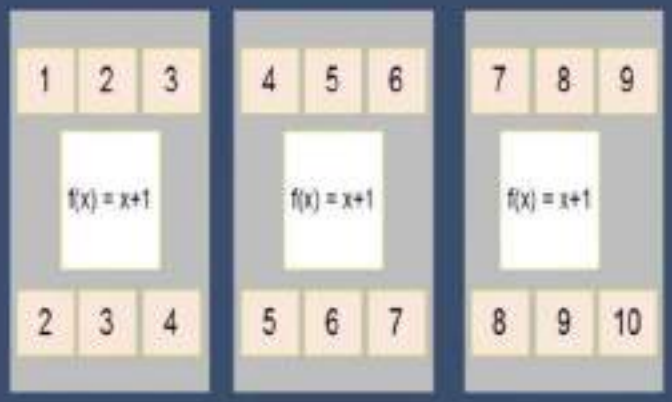

Parallelization
f(x) = x+1
1 2 3
4 5 6
7 8 9
https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

Parallelization

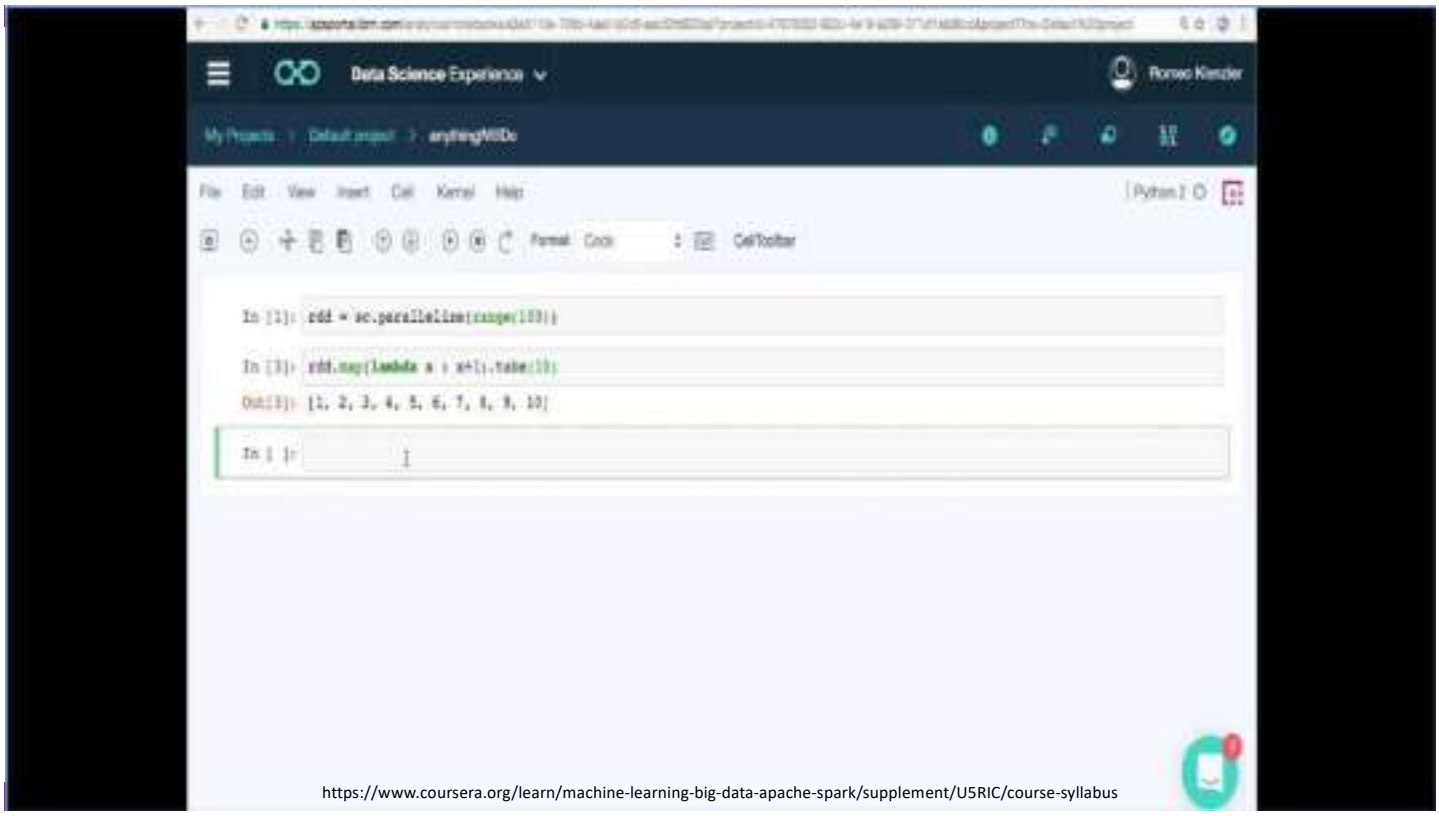


https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

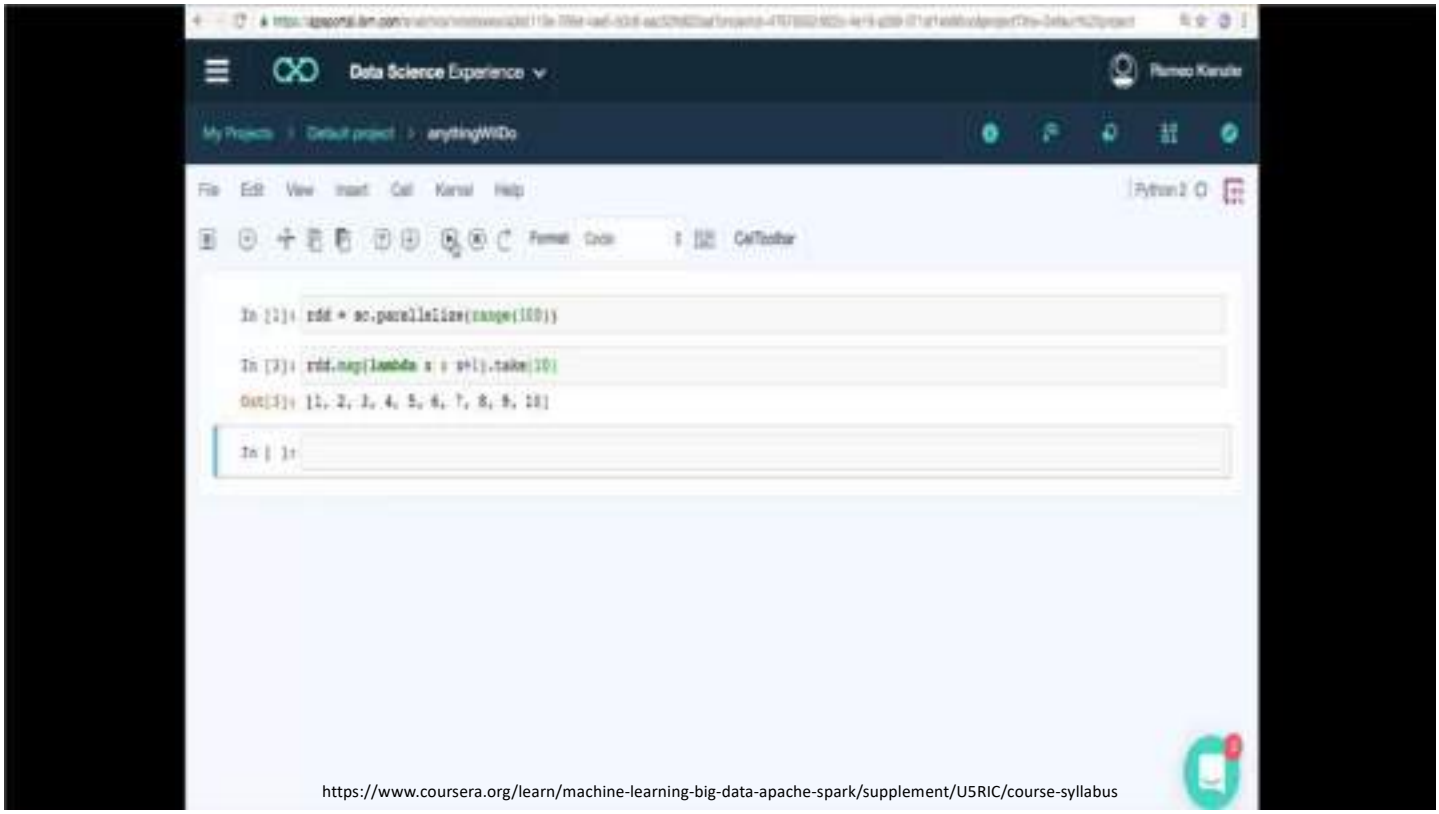
Parallelization



https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus



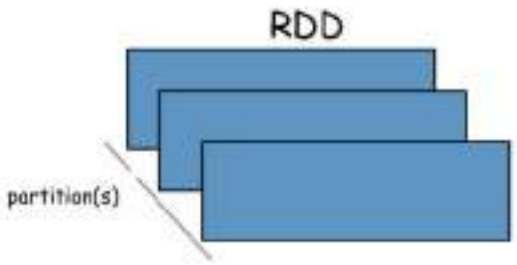
https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus



https://www.coursera.org/learn/machine-learning-big-data-apache-spark/supplement/U5RIC/course-syllabus

Resilient Distributed Datasets (RDDs)

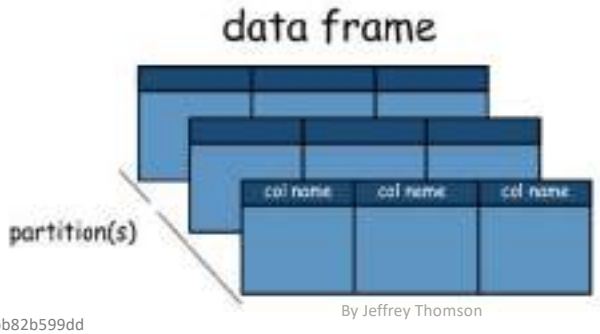
- In Spark, we express our computation through operations on distributed collections that are automatically parallelized across the cluster.
- These collections are called ***resilient distributed datasets***, or **RDD**s.
- RDDs are Spark's fundamental abstraction for distributed data and computation.



By Jeffrey Thomson

The Dataframe

- Since Spark 2.0.0 a DataFrame is a *Dataset* organized into named columns.
- It is conceptually equivalent to a table in a relational database or a DataFrame in R/Python, but with richer optimizations under the hood.
- It can be distirbuted across a cluster

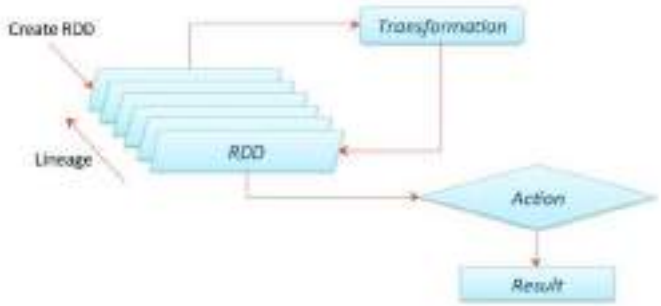


By Jeffrey Thomson

https://towardsdatascience.com/your-first-apache-spark-ml-model-d2bb82b599dd

Transformations and Actions on RDDs

- Transformations are lazy, which means that they do not compute their results right away.
- Transformations are only computed when an **action** requires a result to be returned to the driver program.

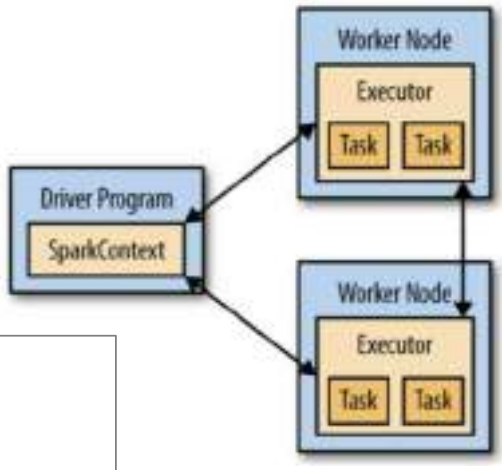


- You may also *persist* an RDD in memory using the persist (or cache) method, in which case Spark will keep the elements around on the cluster for much faster access the next time you query it.

Check this out! https://spark.apache.org/docs/latest/rdd-programming-guide.html#transformations

Components for Distributed Execution in Spark

- At a high level, every Spark application consists of a *driver program* that launches various parallel operations on a cluster.
- The driver program contains your application's main function and defines distributed datasets on the cluster, then applies operations to them.



Example. Python line count

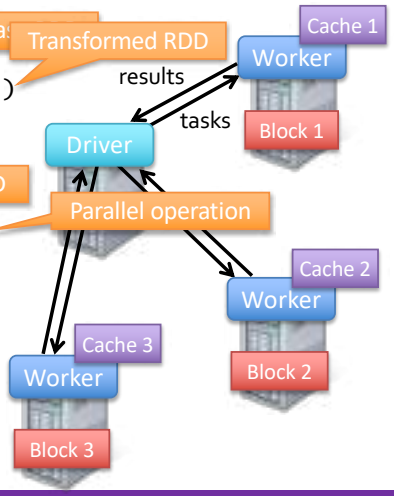
```
>>> lines = sc.textFile("README.md") # Create an RDD called lines
>>> lines.count() # Count the number of items in this RDD
127
>>> lines.first() # First item in this RDD, i.e. first line of README.md u'# Apache Spark'
```

Example: Log Mining

Load error messages from a log into memory, then interactively search for various patterns.

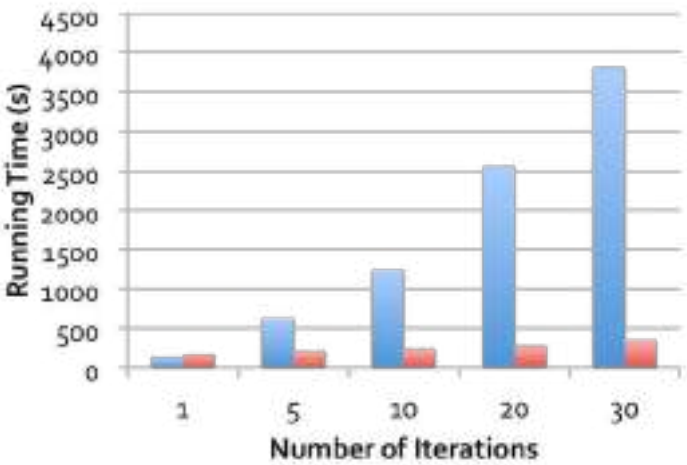
```
lines = spark.textFile("hdfs://...")
errors = lines.filter(_.startsWith("ERROR"))
messages = errors.map(_.split('\t')(2))
cachedMsgs = messages.cache()

cachedMsgs.filter(_.contains("foo")).count
cachedMsgs.filter(_.contains("bar")).count
. . .
```



Result: full-text search of Wikipedia in <1 sec (vs. 20 sec for on-disk data)

Logistic Regression Performance



127 s / iteration

Hadoop

Spark

first iteration 174 s
further iterations 6 s

Hacettepe University
Computer Engineering Department

BBM467 - Data Intensive Applications

Semistructured Data

Dr. Fuat Akal
akal@hacettepe.edu.tr

Variety?

- Learn how to deal with any kind of data
  - data comes in same format from different sources
  - data comes in different (electronic) formats
  - data comes from different media (paper, people, ...)
- **Goal: Use data in ways you did not plan before**
  - enable questions that you did not anticipate

Semi-structured Data

- Make it a design principle
  - data first
  - purpose second
- Deal with the complexities of this approach
  - how do you represent such data?
  - how do you access such data?
  - how can you even understand such data?
  - how can you optimize access to that data

What Needs to be Done?

- How to represent data?
  - decouple data from system that generated it
  - standard data formats to serialize data
    - e.g., XML, JSON, ...
- How to access serialized data?
  - semi-structured data models
  - standard APIs to read & manipulate data
    - e.g., DOM, SAX or Xpath

Example: Health Records

```
<patient name = "Malcolm X">
   <hospital name = "Triemli">
      <disease> Cardiac Arrest </disease>
      <medication> Beta Blocker </medication>
   </hospital>
   <doctor> <treatment>
         <disease> Cardiac Arrest </disease>
         <app dosis="2ml"> Diovan </app>
      </treatment> </doctor>
</patient>
```


Other Reasons to Decouple Data and Schema

- Machine to machine communication
  - Web services
- Legacy applications
  - turn off application, keep the data
  - reduces vendor lock-in (e.g., Oracle, SAP)
  - solves Y2K problems

What is XML?

- Extensible Markup Language
- XML is a syntax to serialize data
  - based on UNICODE (universal)
  - General: any application, any platform, ...
- XML is a family of standards
  - XML schema
  - Web services, WS policies, WS security, …
  - Semantic Web: RDF, RDF Schema, OWL, …
  - Processing: XPath, XSLT, XQuery
  - **all you need to build enterprise Web apps**

HTML

```
<h1> Bibliography </h1>
<p> <i> Foundations of Databases </i>
      Abiteboul, Hull, Vianu
      <br> Addison Wesley, 1995
<p> <i> Data on the Web </i>
      Abiteboul, Buneman, Suciu
      <br> Morgan Kaufmann, 1999
```

XML

```
<bibliography>
    <book>  <title> Foundations… </title>
            <author> Abiteboul </author>
            <author> Hull </author>
            <author> Vianu </author>
            <publisher> Addison Wesley </publisher>
            <year> 1995 </year>
    </book>
    …
</bibliography>
```

HTML describes presentation
XML describes content

XML Terminology

- tags: book, title, author, …
- start tag: <book>, end tag: </book>
- elements: <book>…<book>, <author>…</author>
- elements are nested
- empty element: <red></red> abbrv. <red/>
- an XML document: single *root element*

well formed XML document: if it has matching tags

XML Attributes

```
<book price = “55” currency = “USD”>
  <title> Foundations of Databases </title>
  <author> Abiteboul </author>
   …
  <year> 1995 </year>
</book>
```

Attributes are alternative ways to represent data

XML Namespaces

- In XML, element names are defined by the developer.
- This often results in a conflict when trying to mix XML documents from different XML applications.

XML Namespaces

- This XML carries HTML table information:

```
<table>
  <tr>
    <td>Apples</td>
    <td>Bananas</td>
  </tr>
</table>
```

- This XML carries information about a table (a piece of furniture):

```
<table>
  <name>African Coffee Table</name>
  <width>80</width>
  <length>120</length>
</table>
```


XML Namespaces

- Name conflicts in XML can easily be avoided using a name prefix.
- This XML carries information about an HTML table, and a piece of furniture:

```
<h:table>
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>

<f:table>
  <f:name>African Coffee Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>
```

XML Namespaces

- http://www.w3.org/TR/REC-xml-names
- name ::= [prefix:]localpart

```
<book xmlns:isbn="www.isbn-org.org/def">
    <title> … </title>
    <number> 15 </number>
    <isbn:number> …. </isbn:number>
</book>
```


XML Namespaces

- syntactic: <number> , <isbn:number>
- semantic: provide URL for "shared" schema

```
<tag  xmlns:mystyle = "http://…">
        …
        <mystyle:title> … </mystyle:title>
        <mystyle:number> …
  </tag>
```

defined here

XML vs. Relational

name	phone
John	3634
Sue	6343
Dick	6363

```
<row name = "John"  phone = "3634"/>
<row name = "Sue"   phone = " 6343"/>
<row name = "Dick"  phone = "6363"/>
```


XML vs. Relational

- Structural Differences
  - Tree vs. Table
  - Heterogeneous vs. Homegeneous
  - Optional vs. Strict typing
  - Unnormalized vs. Normalized data
- Some commonalities
  - Logical and physical data independence
  - Declarative semantics
  - Generic data model

Reasons for XML Success

- XML is a general data representation format
- XML is human readable
- XML is machine readable
- XML is internationalized (UNICODE)
- XML is platform independent
- XML is vendor independent
- XML is endorsed by the World Wide Web Consortium
- XML is not a new technology
- XML is not *only* a data representation format, it's a full infrastructure of technologies

What is XML Good for?

- Extensibility and structured nature of XML allows it to be used for communication between different systems

- From one source of XML-based information you can format and distribute it via a multitude of different channels

XML and It's Future

- **XML is everywhere.**
- The XML standard has been developed quickly and a large number of software vendors have adopted it.
- XML might be the most common tool for all data manipulation and data transmission.
- XML can be used to create new languages.

How does XML Work?

- XML tags are not predefined. You must **define your own tags**
- XML uses a **Document Type Definition** (DTD) or an **XML Schema** to describe the data
- XML with a DTD or XML Schema is designed to be **self-descriptive**

XML Validation

XML Document is well-formed, if the rules below are obeyed:

- Every start-tag must have a matching end-tag.
- Tags cannot overlap. Proper nesting is required.
- XML documents can only have one root element.
- XML is case sensitive.
- XML preserves white space within text.
- Elements may contain attributes. If an attribute is present, it must have a value, even if it is an empty string "".

XML Validation - 2

- Element names must obey the following XML naming conventions:
  - Names must start with letters or the "_" character. Names cannot start with numbers of punctuation characters.
  - After the first character, numbers and punctuation characters are allowed.
  - Names cannot contain spaces.
  - Names should not contain the ":" character as it is a "reserved" character.
  - Names cannot start with the letters "xml" in any combination of case.
  - The element name must come directly after the "<" without any spaces between them.

Document Type Definition

- *which* tags and attributes are allowed,
- *where* they can be placed, and
- whether or not they can be *nested* within a given document.

Internal vs External DTD declaration

Internal:

<!DOCTYPE foo [<!ENTITY greeting "hello">]>

External, public:

<!DOCTYPE html PUBLIC "//W3C//DTD HTML 4.01//EN" >

All HTML 4.01 documents are expected to conform to one of three SGML DTDs. The public identifiers of these DTDs are constant and are as follows:

-//W3C//DTD HTML 4.01//EN http://www.w3.org/TR/html4/strict.dtd

-//W3C//DTD HTML 4.01 Transitional//EN http://www.w3.org/TR/html4/loose.dtd

-//W3C//DTD HTML 4.01 Frameset//EN http://www.w3.org/TR/html4/frameset.dtd

Internal DTD

```
<?xml version="1.0"?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to (#PCDATA)>
  <!ELEMENT from (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body (#PCDATA)>
  ]>
  <note>
     <to>Tove</to>
      <from>Jani</from>
      <heading>Reminder</heading>
      <body>Don't forget me this weekend!</body>
  </note>
```

CDATA - The value is character data
PCDATA - Parsed Character Data

PCDATA vs CDATA

- All text in an XML document will be parsed by the parser.
  - Because XML elements can contain other elements

```
<name><first>Bill</first><last>Gates</last></name>
```

- The text inside a CDATA section will be ignored by the parser.
  - Some text, like JavaScript code, contains a lot of "<" or "&" characters which are illegal in XML elements. To avoid errors script code can be defined as CDATA.

PCDATA vs CDATA

- In the example below, everything inside the CDATA section is ignored by the parser.

```
<script>
<![CDATA[
function matchwo(a,b)
{
if (a < b && a < 0) then
  {
  return 1;
  }
else
  {
  return 0;
  }
}
]]>
</script>
```


External DTD

```
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

>> *saved as file note.dtd*

Valid XML Document

- A "**Valid**" XML document is a "Well Formed" XML document, which also conforms to the rules of a Document Type Definition (DTD):

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE note SYSTEM ”note.dtd">
<note> <to>Tom</to> <from>Jane</from> <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```


Validate an XML Document

- If you want to check your syntax with the DTD specification together, you have to ‘publish’ your DTD.
- This means, saving your DTD on a server with a public URI, and with public access rights.
  - http://yourserver.uk/yourdirectory/file.dtd
- Than you can refer it from your XML file as:

```
<!DOCTYPE yourrootelement SYSTEM “http://yourserver.uk/yourdirectory/file.dtd">
```

  - Check Google for online XML validators

External DTD (Example Document)

```
<!--inform the XML processor that an external DTD is referenced-->
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
   <to>Tove</to>
   <from>Jani</from>
   <heading>Reminder</heading>
   <body>Don't forget me this weekend!</body>
</note>
```


XML Schema (XSD)

- XML Schema is an XML based alternative to DTD.
- W3C supports an alternative to DTD called **XML Schema**: http://www.w3.org/XML/Schema
- XML Schemas will likely be used in most Web applications as a replacement for DTDs.
  - XML Schemas are extensible to future additions
  - XML Schemas are richer and more powerful than DTDs
  - XML Schemas are written in XML
  - XML Schemas support data types
  - XML Schemas support namespaces

XML Schema Example

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="note">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="to" type="xs:string"/>
      <xs:element name="from" type="xs:string"/>
      <xs:element name="heading" type="xs:string"/>
      <xs:element name="body" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>
```

Saved as note.xsd

Reference to XML Schema

```
<?xml version="1.0"?>

<note
xmlns="http://www.w3schools.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.w3schools.com note.xsd">
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

Declaring Simple Types in XML Schema

- Here are some XML elements:

```
<lastname>Refsnes</lastname>
<age>36</age>
<dateborn>1970-03-27</dateborn>
```

- And here are the corresponding simple element definitions:

```
<xs:element name="lastname" type="xs:string"/>
<xs:element name="age" type="xs:integer"/>
<xs:element name="dateborn" type="xs:date"/>
```


More on Internet

- http://www.w3schools.com/schema/default.asp

- Simple XSLT Demo
  http://www.w3schools.com/xmL/xsl_intro.asp

XPATH

- XPath is a syntax for defining parts of an XML document
- XPath uses path expressions to navigate in XML documents
- XPath contains a library of standard functions
- XPath is a major element in XSLT
- XPath is a W3C recommendation, thus a Standard (16. November 1999)

XPATH Terminology

- Nodes
- Atomic values
- Items (atomic values or nodes)
- Relationships of nodes
  - Parent
  - Children
  - Siblings
  - Ancestors
  - Descendants

XPATH Nodes

- In XPath, there are seven kinds of nodes:
  - element, attribute, text, namespace, processing-instruction, comment, and document nodes.
- XML documents are treated as trees of nodes.
  - The topmost element of the tree is called the root element.

XPATH Nodes

```
<?xml version="1.0" encoding="ISO-8859-1"?>
 <bookstore>
   <book>
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
   </book>
 </bookstore>
```

Document (root) node

Element node

Attribute node

Atomic Values

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<bookstore>
  <book>
   <title lang="en">Harry Potter</title>
   <author>J K. Rowling</author>
   <year>2005</year>
   <price>29.99</price>
  </book>
</bookstore>
```

nodes with no children or parent

Selecting Nodes

Expression	Description
nodename	Selects all child nodes with this name
/	Selects from the root node
//	Selects nodes in the document from the current node down that match the selection no matter where they are
.	Selects the current node
..	Selects the parent of the current node
@	Selects attributes

Path Expression	Result
bookstore	Selects all the bookstore elements
/bookstore	Selects the root element bookstore **Note:** If the path starts with a slash (/) it always represents an absolute path to an element!
bookstore/book	Selects all book elements that are children of bookstore
//book	Selects all book elements no matter where they are in the document
bookstore//book	Selects all book elements that are descendant of the bookstore element, no matter where they are under the bookstore element
//@lang	Selects all attributes that are named lang

Predicates

- Predicates are used to find a specific node or a node that contains a specific value.
- Predicates are always embedded in square brackets.

Predicates

Path Expression	Result
/bookstore/book[1]	Selects the first book element that is the child of the bookstore element
/bookstore/book[last()]	Selects the last book element that is the child of the bookstore element
/bookstore/book[last()-1]	Selects the last but one book element that is the child of the bookstore element
/bookstore/book[position() <3]	Selects the first two book elements that are children of the bookstore element

Predicates

//title[@lang]	Selects all the title elements that have an attribute named lang
//title[@lang='en']	Selects all the title elements that have an attribute named lang with a value of 'en'
/bookstore/book[price >35.00]	Selects all the book elements of the bookstore element that have a price element with a value greater than 35.00
/bookstore/book[price >35.00]/title	Selects all the title elements of the book elements of the bookstore element that have a price element with a value greater than 35.00

Selecting Unknown Nodes

Wildcard	Description
*	Matches any element node
@*	Matches any attribute node
node()	Matches any node of any kind

Selecting Several Paths

Path Expression	Result
//book/title | //book/price	Selects all the title as well as price elements of all book elements
//title | //price	Selects all the title as well as price elements in the document
/bookstore/book/title | //price	Selects all the title elements of the book element of the bookstore element as well as all the price elements in the document

```
<Course_Catalog>
 <Department Code="CS">
   <Title>Computer Science</Title>
   <Chair>
    <Professor>
     <First_Name>Jennifer</First_Name>
     <Last_Name>Widom</Last_Name>
    </Professor>
   </Chair>
   <Course Number="CS106A" Enrollment="1070">
    <Title>Programming Methodology</Title>
    <Description>Introduction to the programming.</Description>
    <Instructors>
     <Lecturer>
      <First_Name>Jerry</First_Name>
      <Middle_Initial>R.</Middle_Initial>
      <Last_Name>Cain</Last_Name>
     </Lecturer>
     <Professor>
      <First_Name>Eric</First_Name>
      <Last_Name>Roberts</Last_Name>
     </Professor>
    </Instructors>
    <Prerequisites>
     <Prereq>CS106A</Prereq>
    </Prerequisites>
   </Course>
 </Department>
</Course_Catalog>
```

Bütün bölüm başkanlarının soyadlarını döndüren ifade?

//Chair/Professor/Last_Name

500'den fazla kayıtlı öğrenci bulunan derslerin başlıklarını (*Title*) döndüren ifade?

//Course[@Enrollment > 500]/Title


```
<Course_Catalog>
 <Department Code="CS">
   <Title>Computer Science</Title>
   <Chair>
    <Professor>
     <First_Name>Jennifer</First_Name>
     <Last_Name>Widom</Last_Name>
    </Professor>
   </Chair>
   <Course Number="CS106A" Enrollment="1070">
    <Title>Programming Methodology</Title>
    <Description>Introduction to the programming.</Description>
    <Instructors>
     <Lecturer>
      <First_Name>Jerry</First_Name>
      <Middle_Initial>R.</Middle_Initial>
      <Last_Name>Cain</Last_Name>
     </Lecturer>
     <Professor>
      <First_Name>Eric</First_Name>
      <Last_Name>Roberts</Last_Name>
     </Professor>
    </Instructors>
    <Prerequisites>
     <Prereq>CS106A</Prereq>
    </Prerequisites>
   </Course>
 </Department>
</Course_Catalog>
```


"CS106B" dersinin alınmasını ön şart olarak gören en az bir dersi açan bölümlerin adını (*Title*) döndüren ifade?

//Department[Course/Prerequisites/Prereq = "CS106B"]/Title

```
<Course_Catalog>
 <Department Code="CS">
   <Title>Computer Science</Title>
   <Chair>
    <Professor>
     <First_Name>Jennifer</First_Name>
     <Last_Name>Widom</Last_Name>
    </Professor>
   </Chair>
   <Course Number="CS106A" Enrollment="1070">
    <Title>Programming Methodology</Title>
    <Description>Introduction to the programming.</Description>
    <Instructors>
     <Lecturer>
      <First_Name>Jerry</First_Name>
      <Middle_Initial>R.</Middle_Initial>
      <Last_Name>Cain</Last_Name>
     </Lecturer>
     <Professor>
      <First_Name>Eric</First_Name>
      <Last_Name>Roberts</Last_Name>
     </Professor>
    </Instructors>
    <Prerequisites>
     <Prereq>CS106A</Prereq>
    </Prerequisites>
   </Course>
 </Department>
</Course_Catalog>
```

Tanımında (*Description*) "*system*" kelimesi geçen ve 100'den fazla öğrencinin kayıtlı olduğu dersleri veren hocaların soyadlarını döndüren ifade?

//Course[contains(Description, "system") and (@Enrollment > 100)]//Last_Name

More on Internet

- http://www.w3schools.com/xpath/default.asp
- XPATH Demo (Google "try xpathonline")
  - http://www.xpathtester.com/xpath

XQUERY

- XQuery is ***the*** language for querying XML data
- XQuery for XML is like SQL for databases
- XQuery is built on XPath expressions
- XQuery is defined by the W3C
- XQuery is supported by all the major database engines (IBM, Oracle, Microsoft, etc.)

Path Expression vs FLWOR

- Select all the title elements under the book elements that are under the bookstore element that have a price element with a value that is higher than 30.
  - `doc("books.xml")/bookstore/book[price>30]/title`
- **FLWOR** is an acronym for "*For, Let, Where, Order by, Return*".
  -
```
for $x in doc("books.xml")/bookstore/book
where $x/price>30
return $x/title
```

FLWOR

- The **for** clause selects all book elements under the bookstore element into a variable called $x.
- The **where** clause selects only book elements with a price element with a value greater than 30.
- The **order by** clause defines the sort-order. Will be sort by the title element.
- The **return** clause specifies what should be returned.

FLWOR Allows Sorting

- ```
  for $x in doc("books.xml")/bookstore/book
  where $x/price>30
  order by $x/title
  return $x/title
  ```
  ```

XQUERY Gives You a Choice

- Path Expressions:
  - If you just want to copy certain elements and attributes as is
- FLWOR Expressions:
  - Allow sorting of results
  - Allow adding elements/attributes to results

Adding Elements/Attributes to the Result

```
<html>
<body>

<h1>Bookstore</h1>

<ul>
{
for $x in doc("books.xml")/bookstore/book
order by $x/title
return
<li>{data($x/title)}. Category: {data($x/@category)}</li>
}
</ul>
</body>
</html>
```

Added HTML elements to the result.

More on Internet

- http://www.w3schools.com/xquery/default.asp
- XQUERY Demo (Google "try xquery online")
  - http://basex.org/products/live-demo/

Acknowledgement - 1

- The course material used for this class is mostly taken and/or adopted* from the course materials of the Big Data class given by Nesime Tatbul and Donald Kossmann at ETH Zurich (http://www.systems.ethz.ch/).

(*) Original course material is reduced somehow to fit the needs of BBM467. Therefore, original slides were not used as they are.

Acknowledgement - 2

- Some material used for this lecture is taken and/or adopted from the course materials of Dr. Alexandra I. Cristea (http://www.dcs.warwick.ac.uk/~acristea/courses/CS253/).

Graph Databases

BBM467 Data Intensive Applications
Dr. Fuat Akal
akal@hacettepe.edu.tr

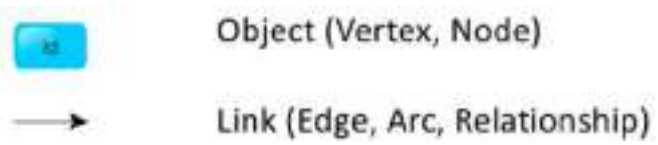
Everyone is Talking About Graphs

- Facebook Open Graph
- Google Knowledge Graph
- Twitter Interest Graph
- …

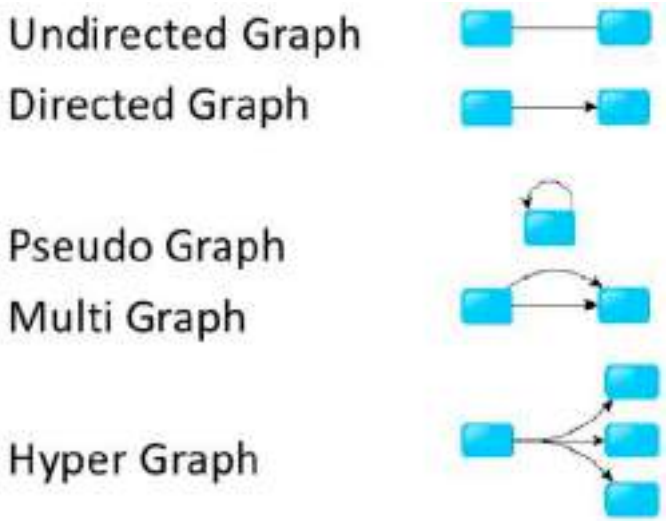
How things are connected!

What is a Graph?

- Formally, a graph is just a collection of *vertices* and *edges*.
- Or, in less intimidating language, a set of *nodes* and the *relationships* that connect them.
- Graphs represent entities as nodes and the ways in which those entities relate to the world as relationships.



Kinds of Graphs



Kinds of Graphs

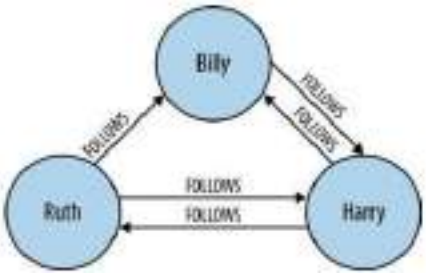
Weighted Graph

Labeled Graph

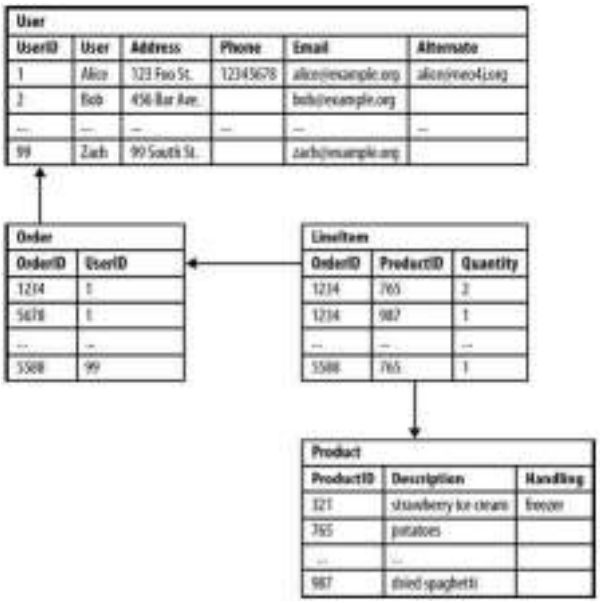
Property Graph

Graphs are Everywhere

- Graphs are extremely useful in understanding a wide diversity of datasets in fields such as science, government, and business.
- For example, Twitter's data is easily represented as a graph.



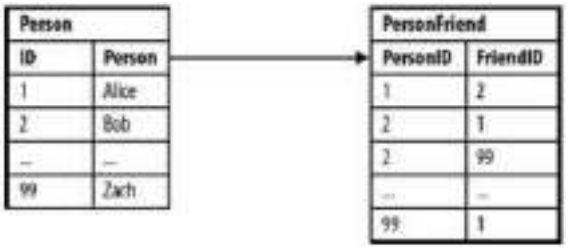
Relational Databases Lack Relationships



Relational Databases Lack Relationships

- Foreign key constraints add additional development and maintenance overhead *just to make the database work*.
- Several *expensive* joins are needed just to discover what a customer bought.
- Reciprocal queries are even more costly. “What products did a customer buy?” is relatively cheap compared to “which customers bought this product?”, which is the basis of recommendation systems.

Recording Relationships in RDBMs




```
Bob's friends

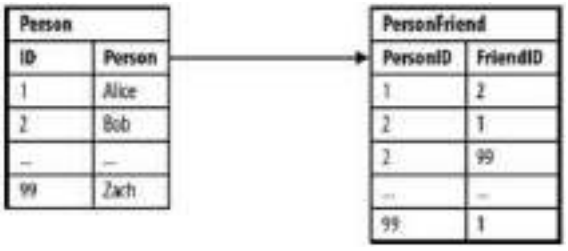
SELECT p1.Person
FROM Person p1 JOIN PersonFriend
         ON PersonFriend.FriendID =
p1.ID
JOIN Person p2
         ON PersonFriend.PersonID =
p2.ID
WHERE p2.Person = 'Bob'
```

```
Who is friends with Bob?

SELECT p1.Person
FROM Person p1 JOIN PersonFriend
         ON PersonFriend.PersonID =
p1.ID
JOIN Person p2
         ON PersonFriend.FriendID =
p2.ID
WHERE p2.Person = 'Bob'
```


Recording Relationships in RDBMs

Things might get complicated!




```
Alice's friends-of-friends

SELECT p1.Person AS PERSON, p2.Person AS FRIEND_OF_FRIEND
FROM PersonFriend pf1
JOIN Person p1
         ON pf1.PersonID = p1.ID
JOIN PersonFriend pf2
         ON pf2.PersonID = pf1.FriendID
JOIN Person p2
         ON pf2.FriendID = p2.ID
WHERE p1.Person = 'Alice' AND pf2.FriendID <> p1.ID
```

Graphs Embrace Relationships

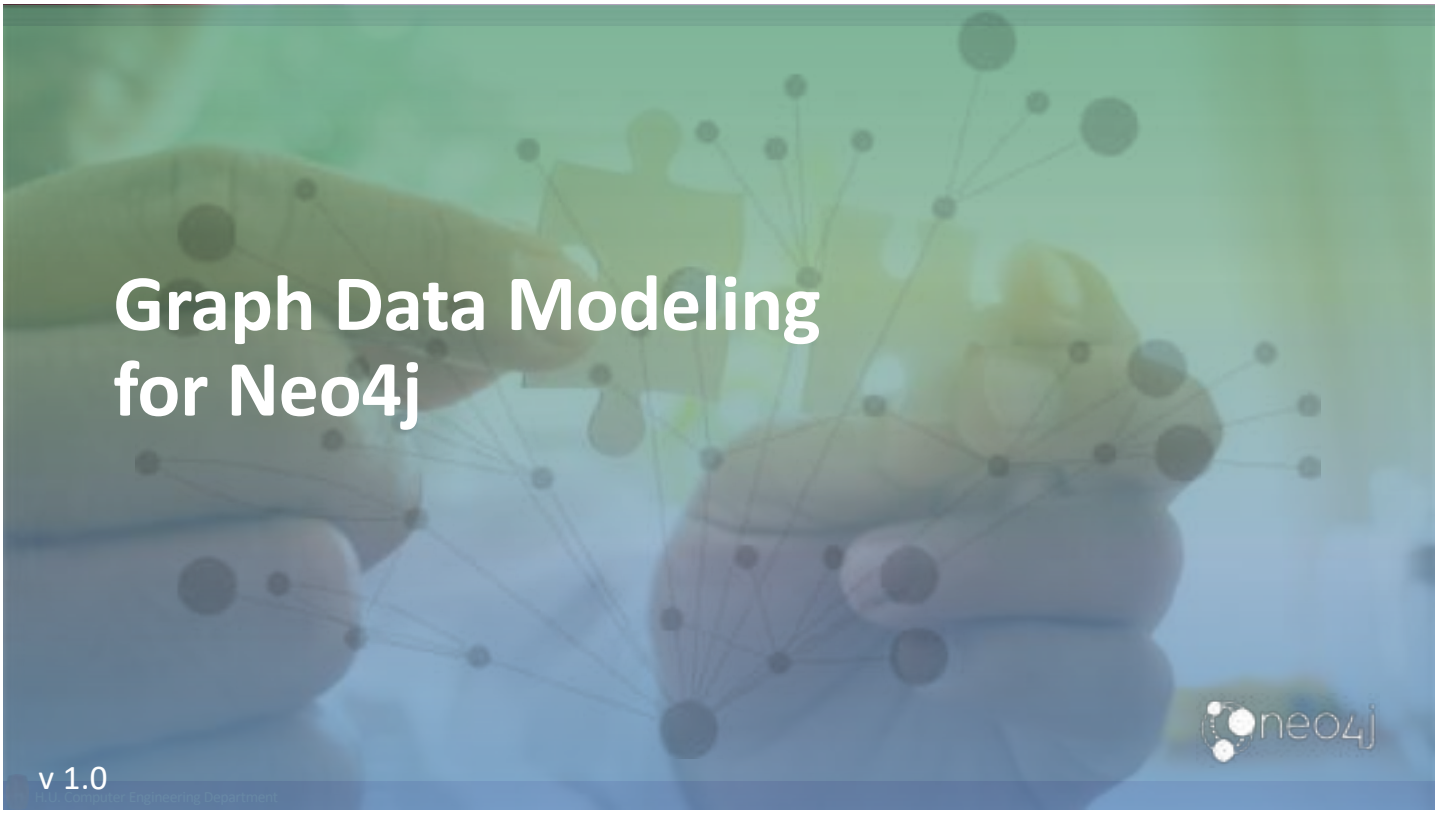
- SQL/NOSQL examples have dealt with *implicitly* connected data.
  - Users infer semantic dependencies between entities, but the data models—and the databases themselves—are blind to these connections.
- We want a cohesive picture of the whole, including the connections between elements.
- In contrast to the SQL/NOSQL data stores we looked at before, in the graph world, connected data is *stored as connected data*.

What is a Graph Database?

- A graph database is NOT for charts & diagrams, or vector artwork ☺
- It is for storing data that is structured as a graph.
  - *"A relational database may tell you the average age of everyone in this place, but a graph database will tell you who is most likely to buy you a beer."*

Neo4J

- Neo4J is a NOSQL Graph Database
- Reliable with ACID transactions
- Scalable: 32 Billion nodes
- http://neo4j.org



Neo4j is a property graph data model

The components of a Neo4j property graph include:

- Nodes (Entities)
- Relationships (Connections between Entities)
- Properties
- Labels

You create graphs to answer important questions for your domain.

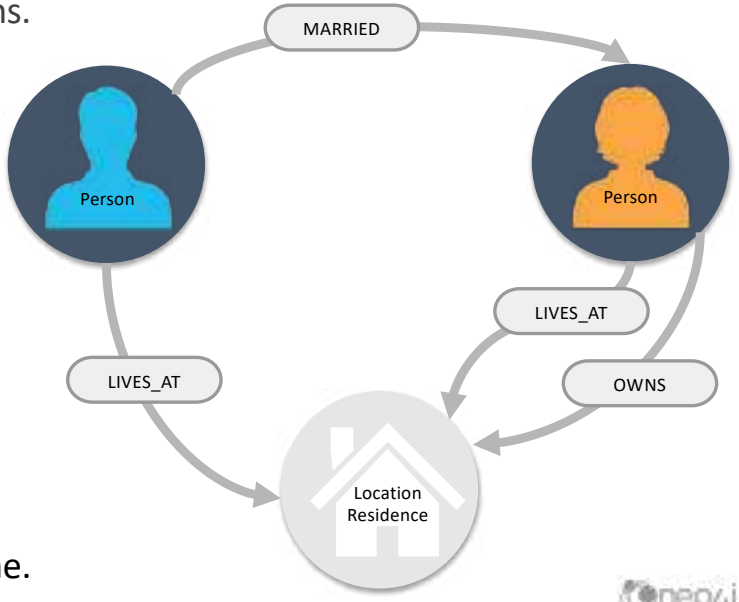
Paths are created and navigated based upon the relationships between nodes.

In Neo4j, entities are nodes with labels

- Represent the nouns in your domain questions.
- Represent the objects or entities in the graph.
- Can be ***labeled*** to represent the <u>type</u> of node or <u>role</u> of node:
  - Person
  - Location
  - Residence
  - Business
- Nodes are grouped by their **labels** for optimizing queries.

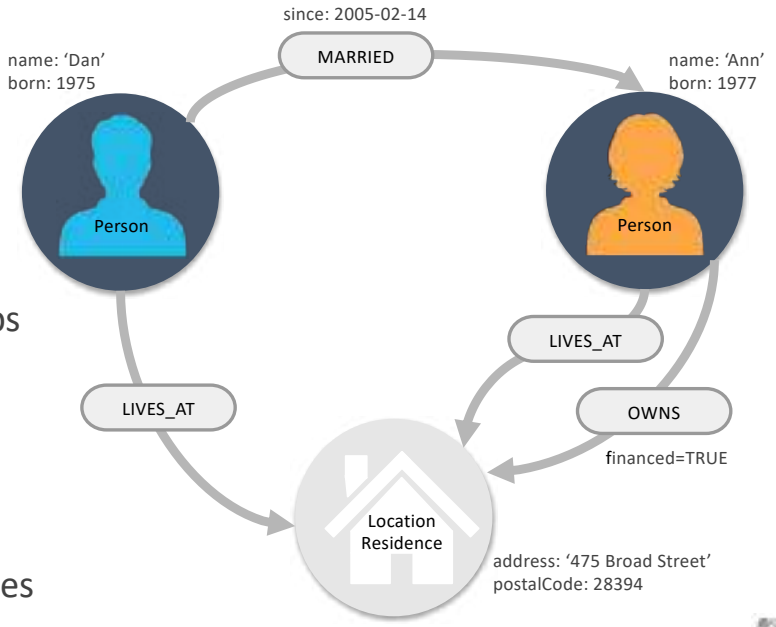
In Neo4j, connections are relationships

- Represent verbs in your domain questions.
- Represent the **connection** between exactly two nodes in the graph.
- Connect nodes of same type or of different types.
- Has a type:
  - MARRIED, LIVES_AT, OWNS
- Directed relationship in the database.
- Used to define paths navigated at runtime.



In Neo4j, properties provide the data

- Adjectives to describe nodes
  - Proper nouns
- Adverbs to describe relationships
  - Strength, weight, quality
- Property:
  - Key/value pair
  - Can be optional or required
  - Values can be unique for nodes



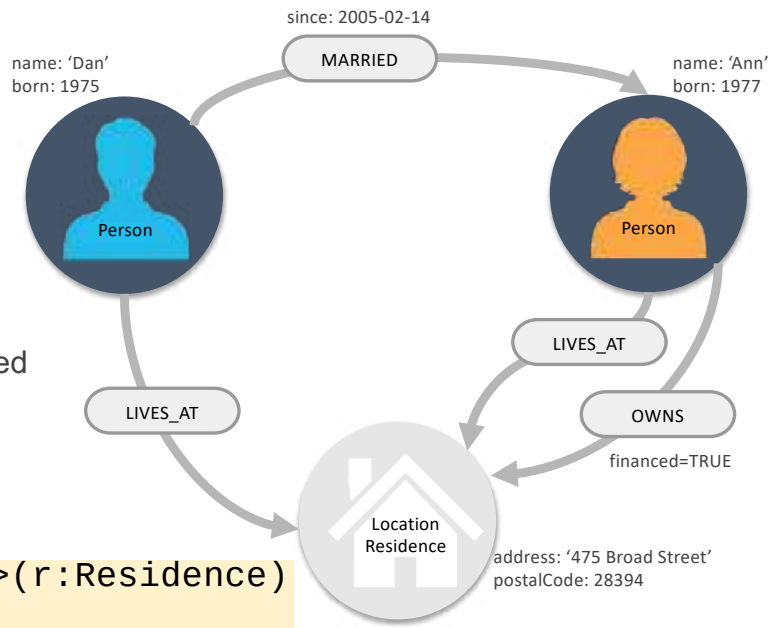
How to Query the Graph Database?

- Cypher: A graph query language
  - Pattern-matching query language
  - Declarative grammar with clauses (like SQL)
  - Aggregation, ordering, limits
  - Create, read, update, delete

In Neo4j, paths are traversed at runtime

Paths are used to:

- Discover what is absolutely necessary to answer questions.
- Eliminate parts of graph not needed to answer a question.

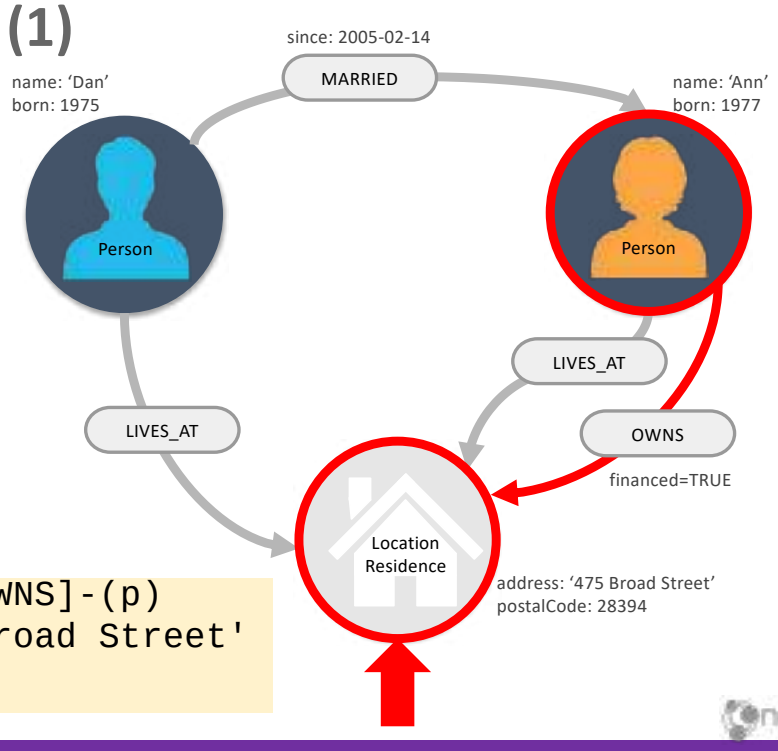


Cypher query:

```
MATCH (p:Person)-[:OWNS]->(r:Residence)
RETURN p,r
```

Path is starting point (1)

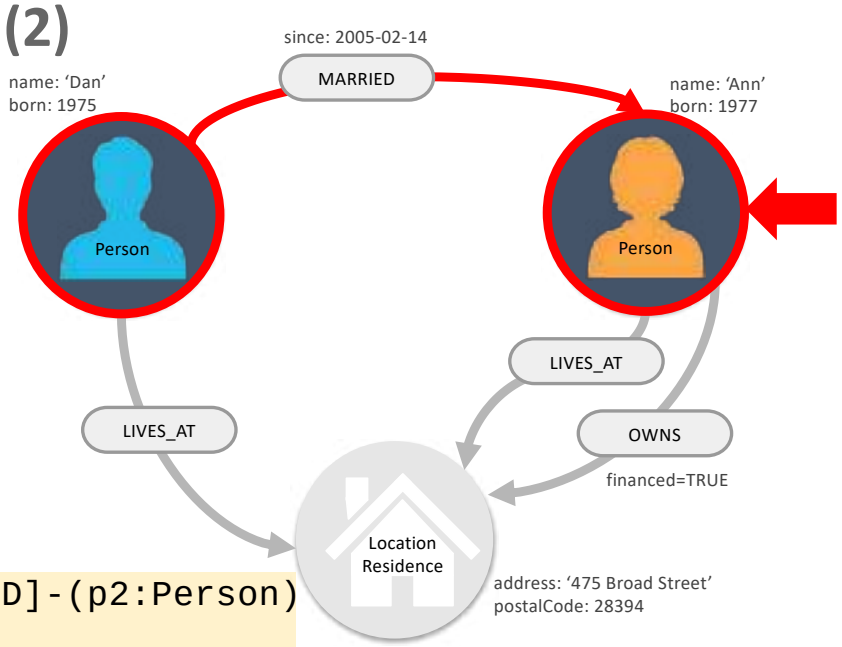
What nodes are visited to find all Person nodes that own the house at 475 Broad Street?




```
MATCH (r:Residence)<-[:OWNS]-(p)
WHERE r.address = '475 Broad Street'
RETURN p
```

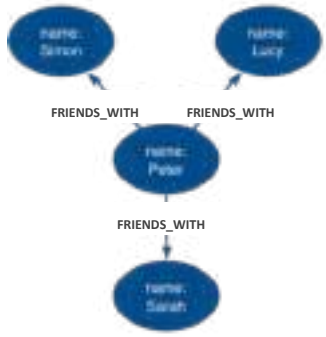

Path is starting point (2)

What nodes are visited to find who is married to Ann?




```
MATCH (p1:Person)-[:MARRIED]-(p2:Person)
WHERE p1.name = 'Ann'
RETURN p2
```

How many relationships?



Nodes can have more than one relationship

Nodes can be connected by more than one relationship

Self relationships are allowed

How important is direction?

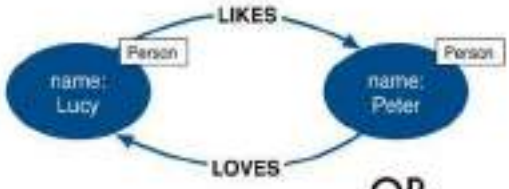
Direction is <u>required</u> for creating the model and in particular for implementing the model (Cypher code) in the underlying Neo4j graph.

Whether direction is used for queries depends on the question:

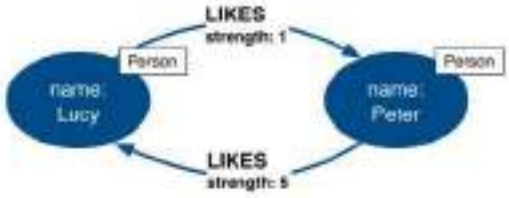
- What are the names of the episodes of the Dr. Who series? (direction <u>not</u> used for the query)
- What episode follows The Ark in Space? (direction used for the query)

Qualifying a relationship

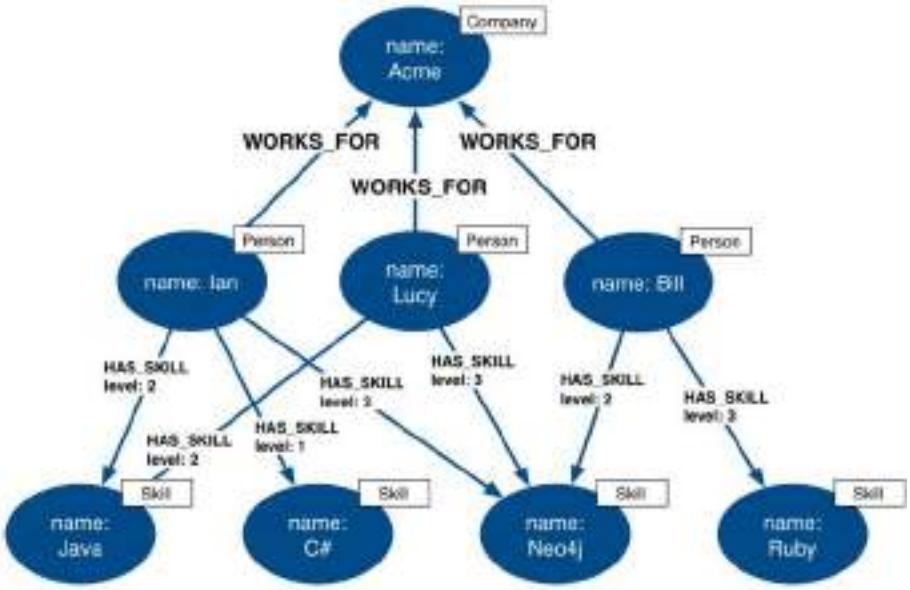
Use properties to describe the weight or quality of the relationship.



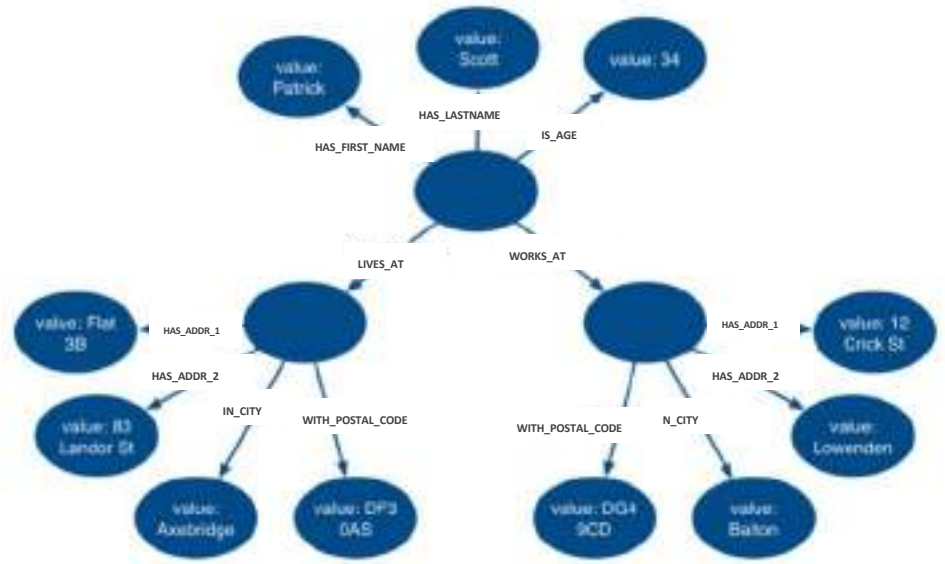
OR



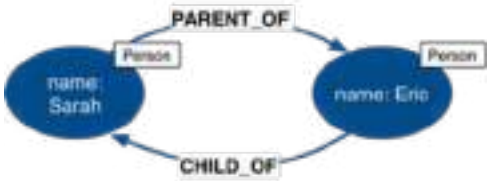
Example: Find expert colleagues



How much fanout will a node have?



Best practice: No symmetric relationships

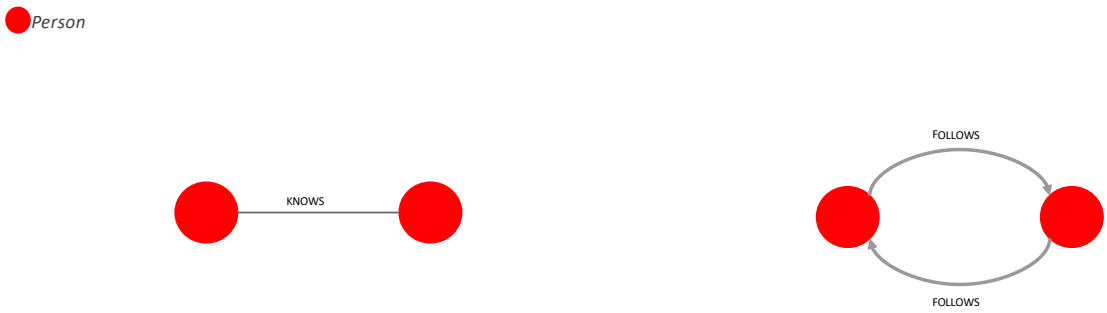


You should <u>never</u> do this.

OR

Note: Relationships require space in the graph so minimizing their numbers is always a good thing.

Be mindful of symmetric relationships



Facebook	Twitter
KNOWS relationship is symmetrical. Even though direction is always stored, relationships can be traversed as undirected.	FOLLOWS relationship is asymmetrical. Direction matters. Two users may follow each other, or one may follow the other but not visa versa.

Knowledge management domain question

Which people who work for the same company as me have similar skills to me?

Identify entities and relationships

Which people who work for the same company as me have similar skills to me?

Entities:

Person
Company
Skill

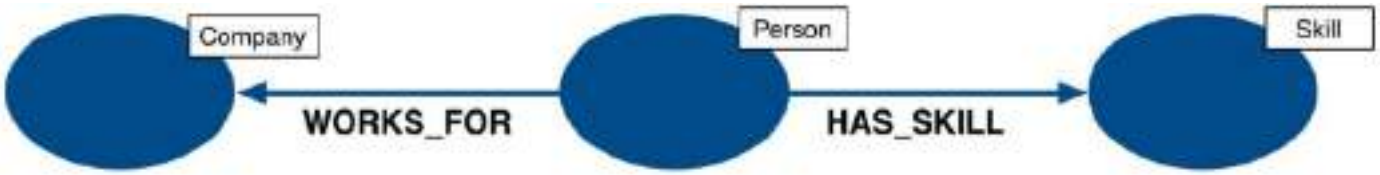
Relationships:

Person **WORKS_FOR** Company
Person **HAS_SKILL** Skill

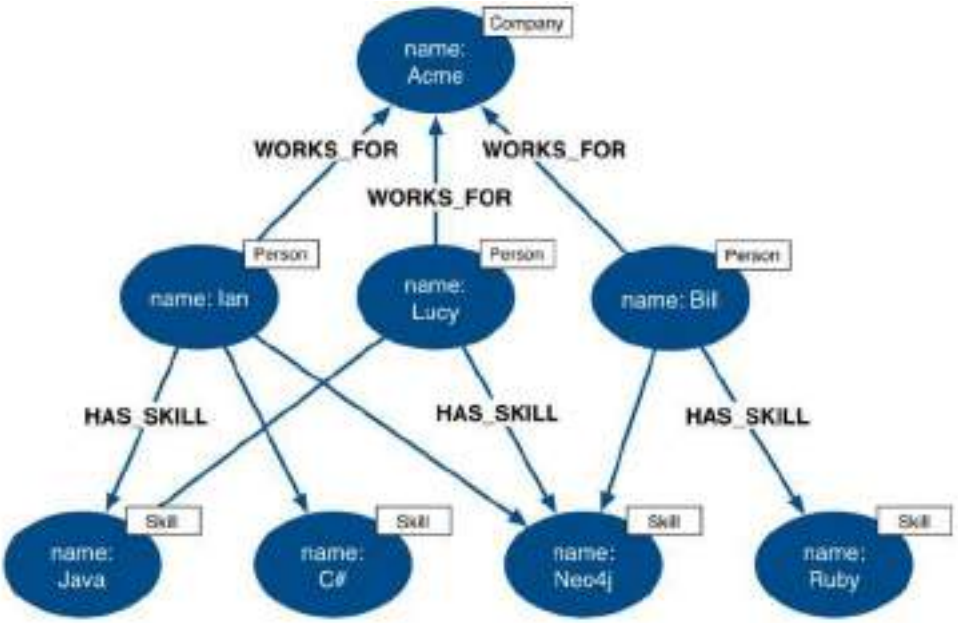
Determine paths used for model

```
(:Person)-[:WORKS_FOR]->(:Company),
(:Person)-[:HAS_SKILL]->(:Skill)
```

```
(:Company)<-[:WORKS_FOR]-(:Person)->[:HAS_SKILL]->(:Skill)
```



Data model with sample data



Cypher query for the question

Which people who work for the same company as me have similar skills to me?

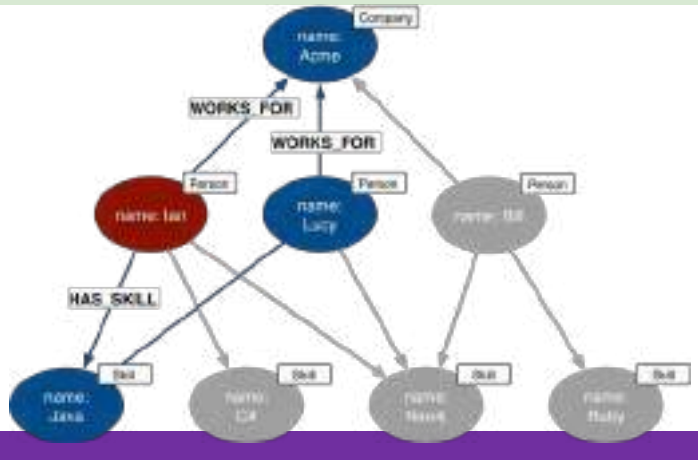
Anchor used to start the navigation

```
MATCH (company)<-[:WORKS_FOR]-(:Person{name:'Ian'})
                -[:HAS_SKILL]->(skill),
(company)<-[:WORKS_FOR]-(colleague)-[:HAS_SKILL]->(skill)

RETURN colleague.name AS name,
count(skill) AS score,
collect(skill.name) AS skills
ORDER BY score DESC
```

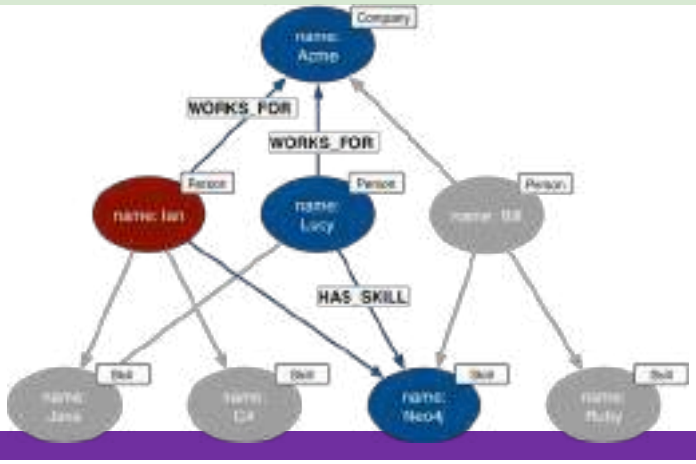
First match

```
MATCH (company)<-[:WORKS_FOR]-(:Person{name:'Ian'})-[:HAS_SKILL]->(skill),
      (company)<-[:WORKS_FOR]-(colleague)-[:HAS_SKILL]->(skill)
RETURN colleague.name AS name,
count(skill) AS score,
collect(skill.name) AS skills
ORDER BY score DESC
```



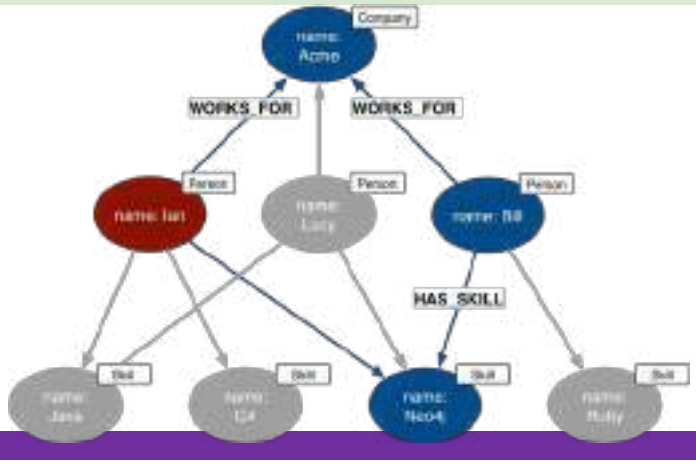
Second match

```
MATCH (company)<-[:WORKS_FOR]-(:Person{name:'Ian'})-[:HAS_SKILL]->(skill),
      (company)<-[:WORKS_FOR]-(colleague)-[:HAS_SKILL]->(skill)
RETURN colleague.name AS name,
count(skill) AS score,
collect(skill.name) AS skills
ORDER BY score DESC
```



Third match

```
MATCH (company)<-[:WORKS_FOR]-(:Person{name:'Ian'})-[:HAS_SKILL]->(skill),
      (company)<-[:WORKS_FOR]-(colleague)-[:HAS_SKILL]->(skill)
RETURN colleague.name AS name,
count(skill) AS score,
collect(skill.name) AS skills
ORDER BY score DESC
```

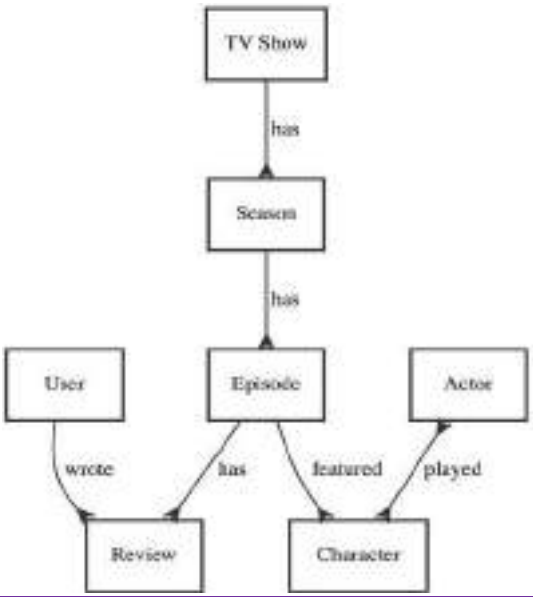


Result

```
MATCH (company)<-[:WORKS_FOR]-(:Person{name:'Ian'})-[:HAS_SKILL]->(skill),
      (company)<-[:WORKS_FOR]-(colleague)-[:HAS_SKILL]->(skill)
RETURN colleague.name AS name,
count(skill) AS score,
collect(skill.name) AS skills
ORDER BY score DESC
```

```
-----------------------------------------------------
| name       | score | skills                        |
-----------------------------------------------------
| "Lucy"     | 2     | ["Java","Neo4j"]              |
| "Bill"     | 1     | ["Neo4j", "Ruby"]             |
-----------------------------------------------------
2 rows
```

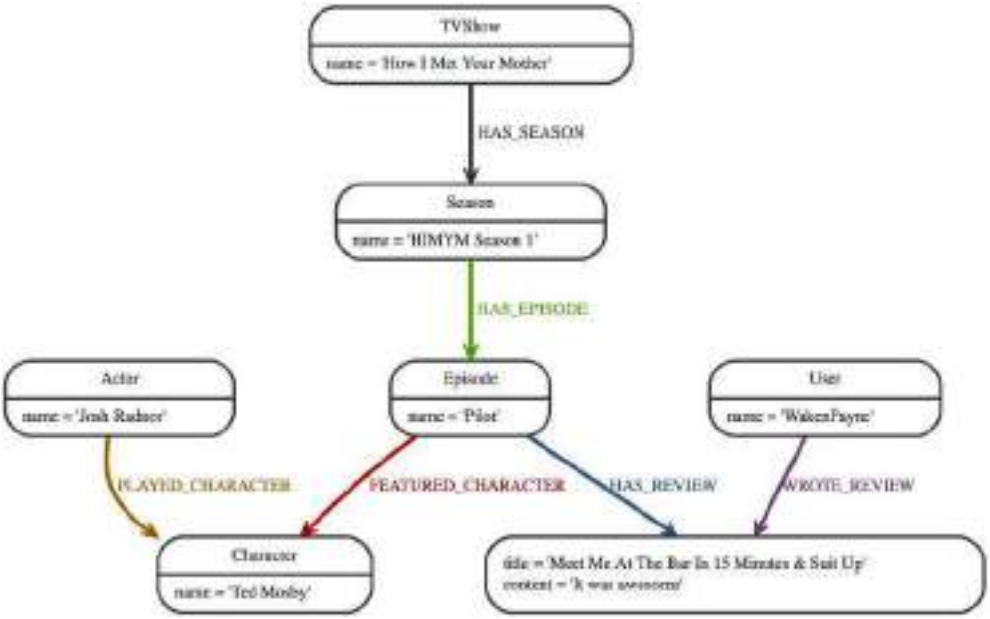
Entity-Relationship Model for TV Show Database



Create the Graph

```
CREATE (himym:TVShow { name: "How I Met Your Mother" })
CREATE (himym_s1:Season { name: "HIMYM Season 1" })
CREATE (himym_s1_e1:Episode { name: "Pilot" })
CREATE (ted:Character { name: "Ted Mosby" })
CREATE (joshRadnor:Actor { name: "Josh Radnor" })
CREATE UNIQUE (joshRadnor)-[:PLAYED_CHARACTER]->(ted)
CREATE UNIQUE (himym)-[:HAS_SEASON]->(himym_s1)
CREATE UNIQUE (himym_s1)-[:HAS_EPISODE]->(himym_s1_e1)
CREATE UNIQUE (himym_s1_e1)-[:FEATURED_CHARACTER]->(ted)
CREATE (himym_s1_e1_review1 { title: "Meet Me At The Bar In 15
Minutes & Suit Up", content: "It was awesome" })
CREATE (wakenPayne:User { name: "WakenPayne" })
CREATE (wakenPayne)-[:WROTE_REVIEW]->(himym_s1_e1_review1)<-
[:HAS_REVIEW]-(himym_s1_e1)
```

Inside the Graph Database



Performance of Dealing Connected Data

- Relationships in a graph naturally form paths. Querying—or traversing—the graph involves following paths.
- Because of the fundamentally path-oriented nature of the data model, the majority of path-based graph database operations are highly aligned with the way in which the data is laid out, making them extremely efficient.

Performance of Dealing with Connected Data

- Partner and Vukotic's experiment (Graph Databases from O'Reilly) seeks to find friends-of-friends in a social network, to a maximum depth of five.
- Given any two persons chosen at random, is there a path that connects them that is at most five relationships long?
- For a social network containing 1,000,000 people, each with approximately 50 friends.

Depth	RDBMS execution time (s)	Neo4j execution time (s)	Records returned
2	0.016	0.01	~2500
3	30.267	0.168	~110,000
4	1543.505	1.359	~600,000
5	Unfinished	2.132	~800,000

When to Use Graph Databases?

- There are two properties of graph databases you should consider when investigating graph database technologies:
  - ***The underlying storage****:* Some graph databases use *native graph storage* that is optimized and designed for storing and managing graphs.
    - Some serialize the graph data into a relational database, an object- oriented database, or some other general-purpose data store.
  - ***The processing engine****:* Some definitions require that a graph database use *index-free adjacency*, meaning that connected nodes physically "point" to each other in the database.

Acknowledgement - 1

- The course material used for this class is mostly taken and/or adopted* from the course materials of the Big Data class given by Nesime Tatbul and Donald Kossmann at ETH Zurich (http://www.systems.ethz.ch/).

(*) Original course material is reduced somehow to fit the needs of BBM467. Therefore, original slides were not used as they are.

Acknowledgement - 2

- Some material used for this lecture is taken and/or adopted from
  - http://docs.neo4j.org/chunked/milestone/index.html
  - Max De Marzi.
  - Michael Hunger from Neo Technology.

Hacettepe University
Computer Engineering Department

BBM467 - Data Intensive Applications

NOSOL

Dr. Fuat Akal
akal@hacettepe.edu.tr

DBMS Provides …

Database Management System (DBMS) provides….

… efficient, reliable, convenient, and safe multi-user storage of and access to massive amounts of persistent data.

What is Wrong with RDBMSs?

- Nothing
  - One size fits all? *Not really*
- Rigid schema design
- Harder to scale
- Joins across multiple nodes? *Hard*
- How does RDMS handle data growth? *Hard*
- Need for a DBA
- Many programmers are already familiar with it
- Transactions and ACID make development easy
- Lots of tools to use

NOSQL: The Name

- Recognition over past decade or so: Not every data management/analysis problem is best solved **exclusively** using a traditional relational DBMS

NOSQL: The Name

- SQL = Traditional relational DBMS
- NoSQL = No SQL = Not using traditional relational DBMS
- No SQL ≠ Don't use SQL language
- NoSQL = Not only SQL

NOSQL Systems

Alternative to traditional relational DBMS

- **Flexible schema**
- **Quicker/cheaper to set up**
- **Massive scalability**
- **Relaxed consistency → higher performance & availability**
- **No declarative query language → more programming**
- **Relaxed consistency → fewer guarantees**

Why NOSQL?

- Big Data
- Scalability
- Data Format
- Manageability

Types of Data

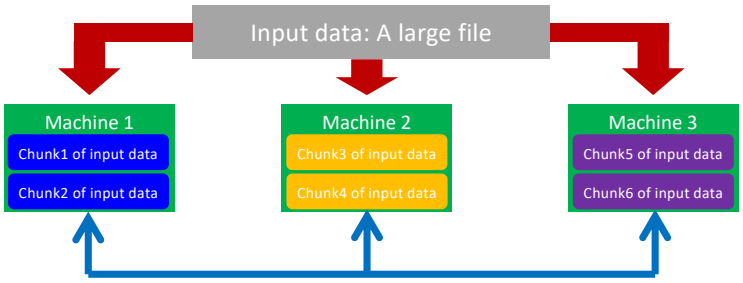
- Structured Data: Have a predefined model, which organizes data into a form that is relatively easy to store, process, retrieve and manage
  - relational data
- Unstructured Data: Opposite of structured data
  - Flat binary files containing text, video or audio
  - Note: data is not completely devoid of a structure (e.g., an audio file may still have an encoding structure and some metadata associated with it)
- Dynamic Data: Data that changes relatively frequentl
  - Office documents and transactional entries in a financial database
- Static Data: Opposite of dynamic data
  - Medical imaging data from MRI or CT scans

Scaling Traditional Databases

- Traditional RDBMSs can be either scaled
  - Vertically (or Up)
    - Can be achieved by hardware upgrades (e.g., faster CPU, more memory, or larger disk)
    - Limited by the amount of CPU, RAM and disk that can be configured on a single machine
  - Horizontally (or Out)
    - Can be achieved by adding more machines
    - Requires database *sharding* and probably *replication*
    - Limited by the Read-to-Write ratio and communication overhead

Why Sharding Data?

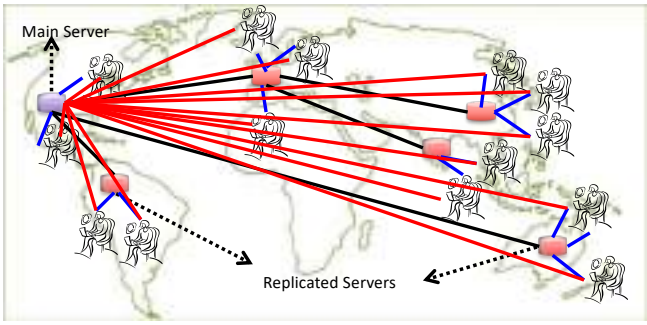
- Data is typically *sharded* (or *striped*) to allow for concurrent/parallel accesses



E.g., Chunks 1, 3 and 5 can be accessed in parallel

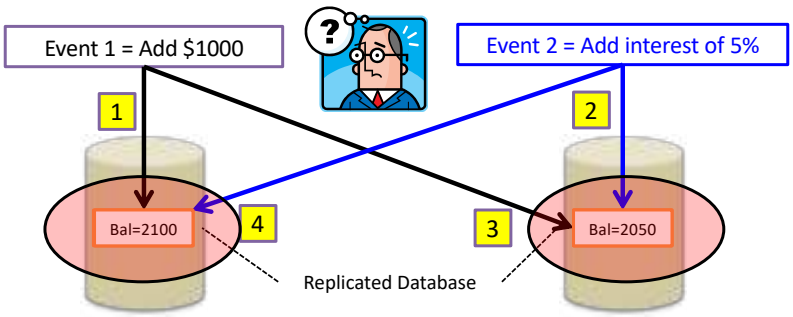
Why Replicating Data?

- Replicating data across servers helps in:
  - Avoiding performance bottlenecks
  - Avoiding single point of failures
  - And, hence, enhancing scalability and availability



Consistency Becomes a Challenge (Example)

- In an e-commerce application, the bank database has been replicated across two servers
- Maintaining consistency of replicated data is a challenge

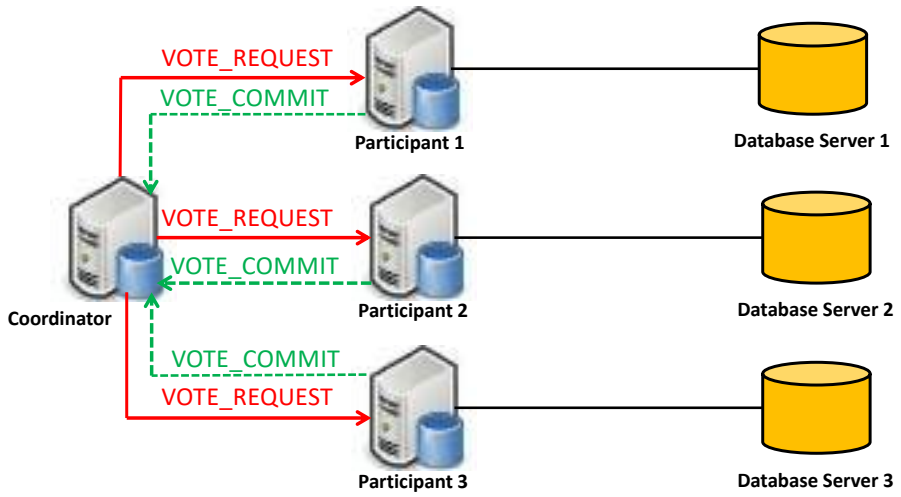


ACID Semantics

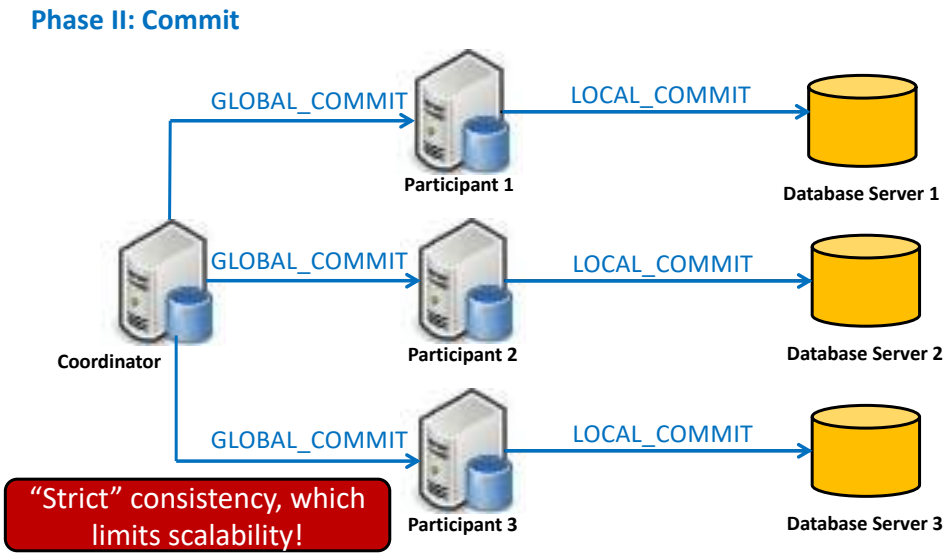
- **Atomicity**: All or nothing.
- **Consistency**: Consistent state of data and transactions.
- **Isolation**: Transactions are isolated from each other.
- **Durability**: When the transaction is committed, state will be durable.

The Two-Phase Commit Protocol (2PC)

Phase I: Voting



The Two-Phase Commit Protocol (2PC)



CAP Theorem

- Also known as Brewer's Theorem by Prof. Eric Brewer, published in 2000 at University of Berkeley.
- "Of three properties of a shared data system: **data consistency, system availability and tolerance to network partitions**, only two can be achieved **at any given moment**."
- Check this out: http://www.cs.berkeley.edu/~brewer/cs262b-2004/PODC-keynote.pdf

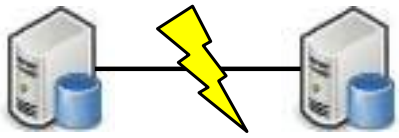
The CAP Theorem

- The limitations of distributed databases can be described in the so called the CAP theorem
  - Consistency: every node always sees the same data at any given instance (i.e., strict consistency)
  - Availability: the system continues to operate, even if nodes in a cluster crash, or some hardware or software parts are down due to upgrades
  - Partition Tolerance: the system continues to operate in the presence of network partitions

CAP theorem: any distributed database with shared data, can have *at most two* of the three desirable properties, C, A or P

The CAP Theorem (*Cont'd*)

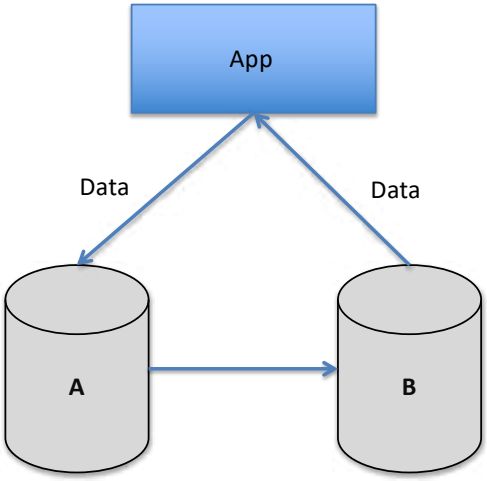
- Let us assume two nodes on opposite sides of a network partition:



- Availability + Partition Tolerance forfeit Consistency
- Consistency + Partition Tolerance entails that one side of the partition must act as if it is unavailable, thus forfeiting Availability
- Consistency + Availability is only possible if there is no network partition, thereby forfeiting Partition Tolerance

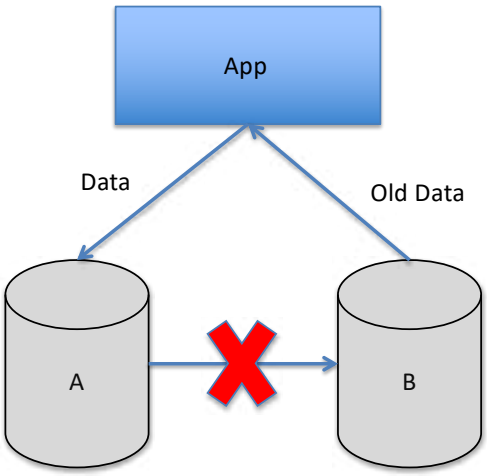
Simple Proof

Consistent and available
No partition.



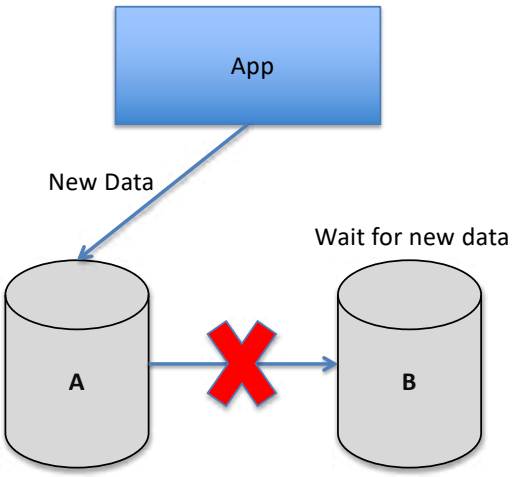
Simple Proof

Available and partitioned
Not consistent, we get back old data.



Simple Proof

Consistent and partitioned
Not available, waiting…

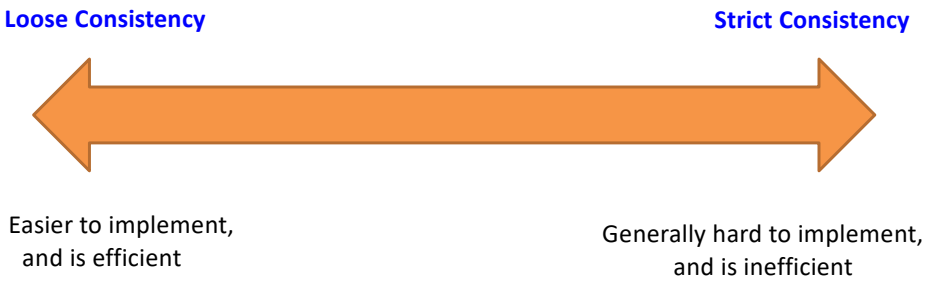


Large-Scale Databases

- When companies such as Google and Amazon were designing large-scale databases, 24/7 Availability was a key
  - A few minutes of downtime means lost revenue
- When *horizontally* scaling databases to 1000s of machines, the likelihood of a node or a network failure
  increases tremendously
- Therefore, in order to have strong guarantees on Availability and Partition Tolerance, they had to sacrifice "strict" Consistency (*implied by the CAP theorem*)

Trading-Off Consistency

- Maintaining consistency should balance between the strictness of consistency versus availability/scalability
  - Good-enough consistency *depends on your application*



BASE Semantics

- Any data store can achieve Atomicity, Isolation and Durability but do you always need consistency?
  - By giving up ACID properties, one can achieve higher performance and scalability.
- **BASE** stands for Basically Available, Soft state, Eventual consistency.
- Eventually-consistent services are often classified as providing **BASE semantics**, in contrast to traditional ACID guarantees.

Eventual Consistency

- A database is termed as *Eventually Consistent* if:
  - All replicas will *gradually* become consistent in the absence of updates

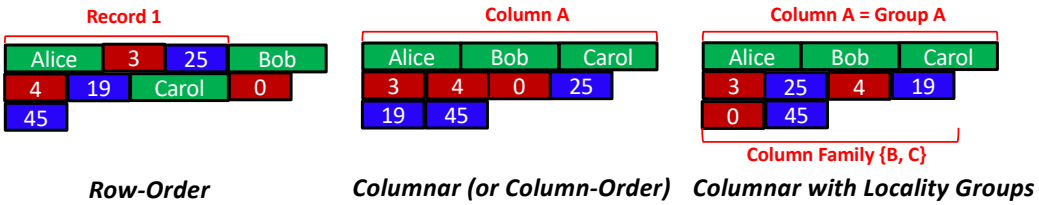
NOSQL Database Types

- Column Stores
  - Each storage block contains data from only one column
    - C-Store, Hbase, Amazon Simple DB, Cloudate
- Document Stores
  - Stores documents made up of tagged elements
    - MongoDB, CouchDB
- Key-Value Stores
  - Hash table of keys
    - DynamoDB, Azure Table Storage
- Graph Databases
  - Stores data as connected
    - Neo4J, VertexDB

Many more can be found here: http://nosql-database.org/

Columnar Databases

- Columnar databases are a hybrid of RDBMSs and Key-Value stores
  - Values are stored in groups of zero or more columns, but in Column-Order (as opposed to Row-Order)



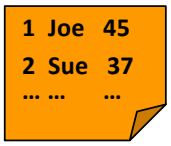
Row-Order ***Columnar (or Column-Order)*** ***Columnar with Locality Groups***

  - Values are queried by matching keys

- E.g., HBase and Vertica

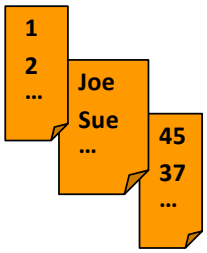
Rows vs. Columns

SQL Server
DB2
Oracle



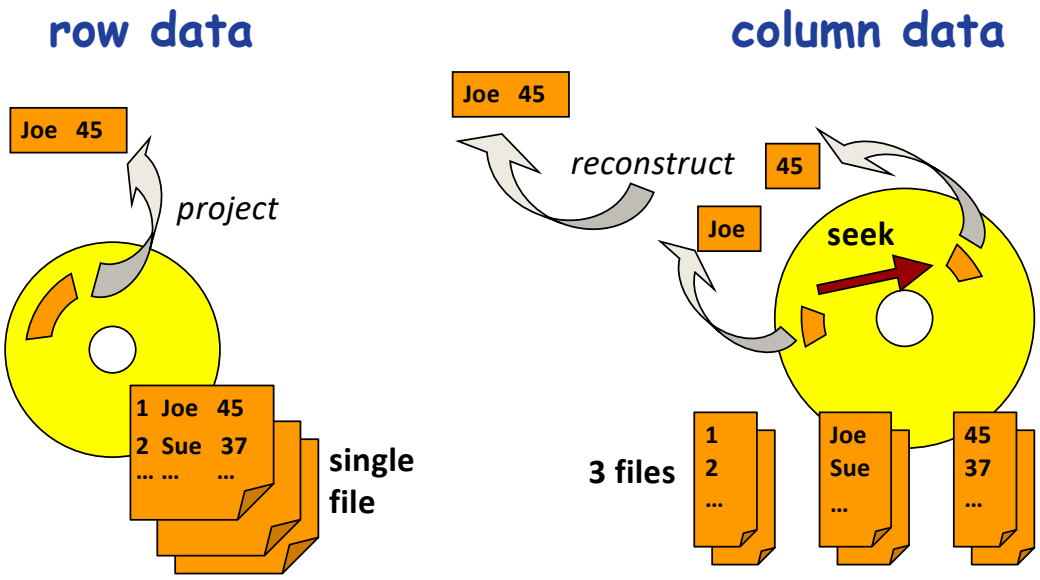
row stores

Hbase
Sybase IQ
MonetDB
CStore



column stores

Rows vs. Columns



Rows vs. Columns

Pros

- Data compression
- Improved Bandwidth Utilization
- Improved Cache Locality

Cons

- Increased Disk Seek Time ?
- Increased Cost of Inserts
- Increased Tuple Reconstruction Costs

Document Stores

- Documents are stored in some standard format or encoding (e.g., XML, JSON, PDF or Office Documents)
  - These are typically referred to as Binary Large Objects (BLOBs)
- Documents can be indexed
  - This allows document stores to outperform traditional file systems
- E.g., MongoDB and CouchDB

JSON

- JSON: **J**ava**S**cript **O**bject **N**otation.
- JSON is syntax for storing and exchanging text information. Much like XML.
- JSON is smaller than XML, and faster and easier to parse.
- JSON is language independent
- JSON is "self-describing" and easy to understand

JSON - Evaluates to JavaScript Objects

- The JSON text format is syntactically identical to the code for creating JavaScript objects.
- Because of this similarity, instead of using a parser, a JavaScript program can use the built-in **eval()** function and execute JSON data to produce native JavaScript objects.

JSON Example

```
{
   "name":"John Johnson",
   "street":"Oslo West 555",
   "age":33,
   "phone":"555 1234567”
};
```

http://www.w3schools.com/json/json_intro.asp

```
<!DOCTYPE html>
<html>
<body>
<p>
  Name: <span id="jname"></span><br/>
  Age: <span id="jage"></span><br/>
  Address: <span id="jstreet"></span><br/>
  Phone: <span id="jphone"></span><br/>
</p>

<script>
  var JSONObject = {
    "name":"John Johnson",
    "street":"Oslo West 555",
    "age":33,
    "phone":"555 1234567"};
  document.getElementById("jname").innerHTML=JSONObject.name
  document.getElementById("jage").innerHTML=JSONObject.age
  document.getElementById("jstreet").innerHTML=JSONObject.street
  document.getElementById("jphone").innerHTML=JSONObject.phone
</script>
</body>
</html>
```


JSON Example

```
var employees = [
   { "firstName":"John" , "lastName":"Doe" },
   { "firstName":"Anna" , "lastName":"Smith" },
   { "firstName":"Peter" , "lastName": "Jones" }
];
```

The first entry in the JavaScript object array can be accessed like this:

`employees[0].firstName + " " + employees[0].lastName;`

The returned content will be:

John Doe

http://www.w3schools.com/json/tryit.asp?filename=tryjson_objectarray

JSON

- Plain text
- Self-describing (human readable)
- Hierarchical (values within values)
- No end tag
- Shorter
- Quicker to read and write
- Uses arrays
- No reserved words

Why JSON?

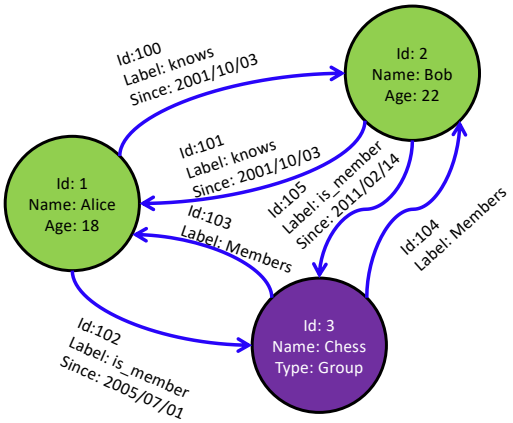
- For AJAX applications, JSON is faster and easier than XML
- Using XML
  - Fetch an XML document
  - Use the XML DOM to loop through the document
  - Extract values and store in variables
- Using JSON
  - Fetch a JSON string
  - eval() the JSON string
    - The eval() function evaluates or executes an argument.

Key-Value Stores

- Keys are mapped to (possibly) more complex value (e.g., lists)
- Fast access to small data values
- Keys can be stored in a hash table and can be distributed easily
- Such stores typically support regular CRUD (create, read, update, and delete) operations
  - That is, no joins and aggregate functions
- E.g., Amazon DynamoDB and Cassandra, Redis

Graph Databases

- Data are represented as vertices and edges
- Graph databases are powerful for graph-like queries
  - Find the shortest path between two elements
- E.g., Neo4j and VertexDB



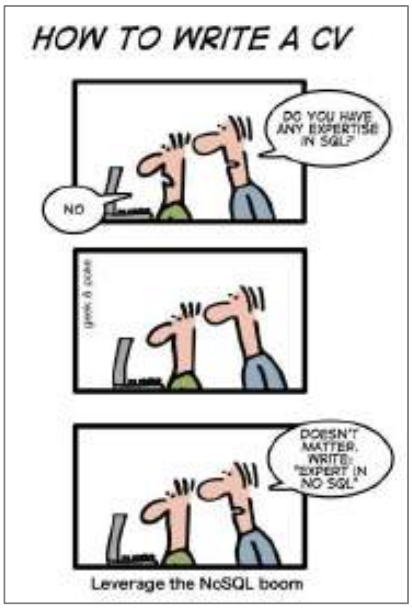
Which One to Use?

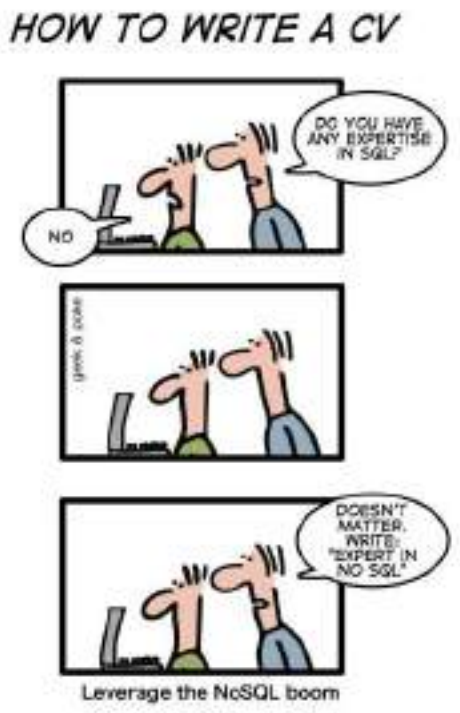
- *Key-value stores*: Processing a constant stream of small reads and writes.
- *Document databases*: Natural data modeling. Programmer friendly. Rapid development. Web friendly.
- *RDMBS*: OLTP. SQL. Transactions. Relations.
- *Columnar*: Handles size well. Massive write loads. High availability. Multiple-data centers. MapReduce
- More: http://highscalability.com/blog/2011/6/20/35-use-cases-for-choosing-your-next-nosql-database.html

Good Starting Point for NOSQL

- Your Ultimate Guide to the Non-Relational Universe!

  http://nosql-database.org/





Acknowledgements

- Some material used for this lecture is taken and/or adopted from
  - Firat Atagun
  - Rakesh Venkat
  - Keith W. Hare
  - Jennifer Widom from Stanford University
  - Stavros Harizopoulos from MIT
  - Mohammad Hammoud from CMU Qatar

Big Data Analysis with SQL

BBM467 Data Intensive Applications
Dr. Fuat Akal
akal@hacettepe.edu.tr

Big Database, Big Data Stores, and SQL

Big Database Types

- Analytic systems (data warehouses)
- Operational Systems
  - Non-transactional, unstructured or semi-structured
  - Non-transactional, structured
  - ACID-compliant RDBMs
- Search

Analytic Systems

- Good for deep analysis
- Let you ask lots of questions

Impala SQL

Hive QL

Higher cost per terabyte

NoSQL: Operational, Unstructured and Semi-structured

- Good for carefully focused operational applications
  - Key value stores like Apache HBase and Apache Cassandra
  - Document stores like MongoDB and Couchbase

- These systems vary in the form of data you store
  - Binary arrays in Hbase
  - JSON like structures in MongoDB and Couchbase

NoSQL: Operational, Unstructured and Semi-structured (cont'd)

- These all belong to the category of NoSQL databases and their support for SQL is usually weak or non-existent.

- They provide simple DML and query commands and they physically organize records by a specific lookup key.

- Records can be stored in massive numbers and then a record can be found rapidly by its lookup key, but not easily by other values in the record.

- Performance vs. Flexibility

Non-transactional, Structured Systems

- A midway between NoSQL and full RDBMS systems, for structured tables.

- Apache Kudu is good for structured data, mix of analytic and simple operational uses
  - Enforce primary keys constraint
  - Allow single DML on individual rows
  - No foreign key constraints
  - Does not allow multi-row ACID-complaint transactions

Performance of Big Data Stores

	Large analytic queries	Single-row DML statements
Impala	Excellent performance	Slower INSERT, no UPDATE or DELETE
HBase	Poor performance	Excellent performance
Kudu	Good performance	Good performance

- Impala and HBase are great at one thing
- Kudu provides a compromise in performance between the large-scale analytic databases and the large-scale NoSQL operational databases.

Big Data ACID-Compliant RDBMSs

- Based on Apache Hbase
- Good choice if you need to support massive databases, that have online transaction processing or OLTP.
  - Like a large financial application, or a large travel scheduling system.
- They fully implement SQL on big data

Search Engines

- Good for rapid lookups, especially with long text content
- If you have some data set like the text of a few hundred thousand news articles, a relational database would be crippled, trying to find some particular quotation or passage.

Big Database Types

- Analytic systems (data warehouses) Impala, Hive
- Operational Systems
  - Non-transactional, unstructured or semi-structured HBase, MongoDB
  - Non-transactional, structured Kudu
  - ACID-compliant RDBMs Splice Machine, Trafodian, Phoenix
- Search Solr, Elasticsearch

Features of SQL for Big Data Analysis

Challenges

- Although SQL is so well understood and widely supported
- There are challenges at large scale
- Distributed transactions
  - Hard to implement, many nodes and replicas, potential network partition
- Data variety
  - SQL is almost by definition weak with unstructured and semi-structured data.
  - The practice of schema on write is an undesirable way to handle new content for big data

What We Keep

- A primary strength of SQL is the straightforward way unless you express a wide variety of analytic queries on data using SELECT statements.
  - Some big data systems offer only very limited support, for example they might not allow multi-table selects.
  - Other big data systems have much broader support for SELECT statements.
    - E.g., Hive and Impala
- Some dialect of data-definition or DDL that create, alter and drop statements.
- SQL grant and revoke statements

What We Give Up

- Transactions, which means no…
  - Unique columns
  - Primary key constraints
  - Foreign key constraints
  - Synchronized indexes
  - Triggers
- UPDATE and DELETE statements
  - Due to lack the ability (of HDFS) to update file content in place

What We Add

- Table partitions and bucketing
- Support for many file formats
- Complex data types

Table Partitions

- Rows are stored in physically separate file directories, or partitions, according to some value found in the rows.

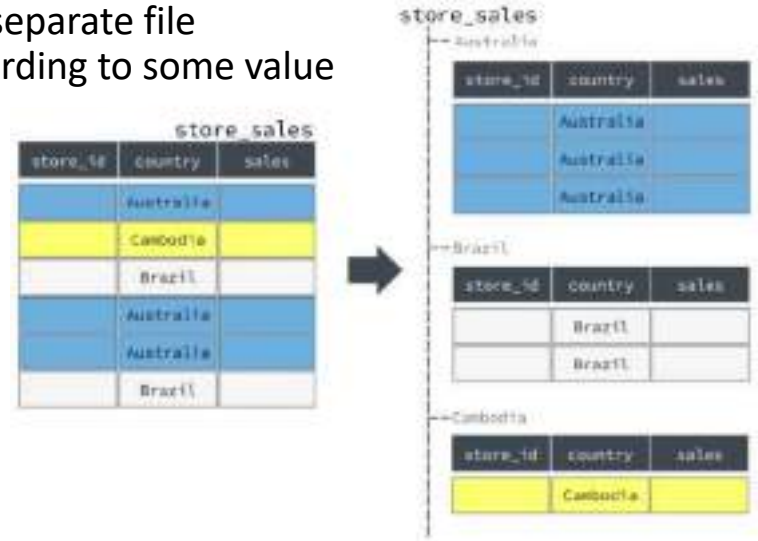
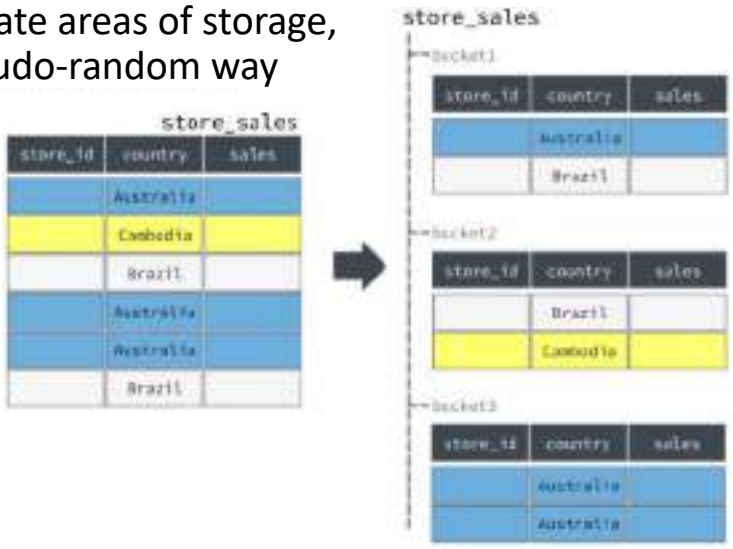


Table Bucketing

- Subdivides the rows into separate areas of storage, but usually in a random or pseudo-random way
- Used for analysis based on an arbitrary sample of your table, and not all the data.



Support for Many File Formats

csv (comma delimited)

```
12,The Lemon Tree,comedy-drama,2017
```

xml

```
<movie>
   <id>12</id>
   <title>The Lemon Tree</title>
   <genre>comedy-drama</genre>
   <year>2017</year>
</movie>
```

json

```
{"id":12,"title":"The Lemon Tree",
 "genre":"comedy-drama","year":2017}
```

No lock into one vendor or one kind of file

Complex Types: ARRAY

movie	actor	showtimes
The Lemon Tree	Mary Nichols Joe Dean	3:40pm 6:00pm 8:30pm

movie

id	title
12	The Lemon Tree

actor_in_movie

movie_id	actor
12	Mary Nichols
12	Joe Dean

movie_showtime

movie_id	showtime
12	3:40pm
12	6:00pm
12	8:30pm

Normalized Design

Complex Types: MAP

customer

cust_id	name	phones
a	Alisa	
b	Berat	

key	value
home	555-1111
work	555-2222
mobile	555-3333

key	value
mobile	555-4444

Complex data types allow your SELECTs to easily fetch all the contents from one table at runtime, rather than having to compute results from multiple tables.

And this can result in improved query time.

Different Big Data Stores

Where to Store Big Data

- On premises
- In cloud
- Both (hybrid)

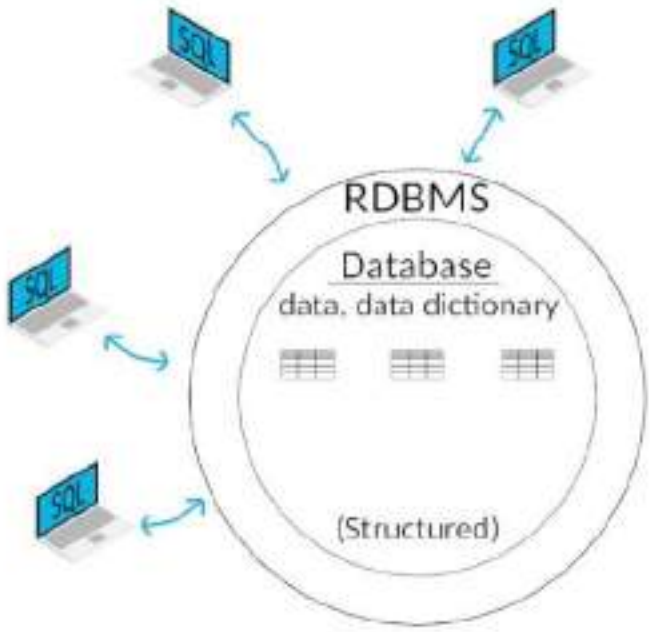
Rent or Buy

Operating Expenditure
(OPEX)

Capital Expenditure
(CAPEX)

Coupling of Data and Metadata

- CREATE TABLE in a transactional RDBMS
  - Check for foreign key consistency
  - Set up files
  - Set up indexes
  - Record table details



Storing Data in an RDBMS

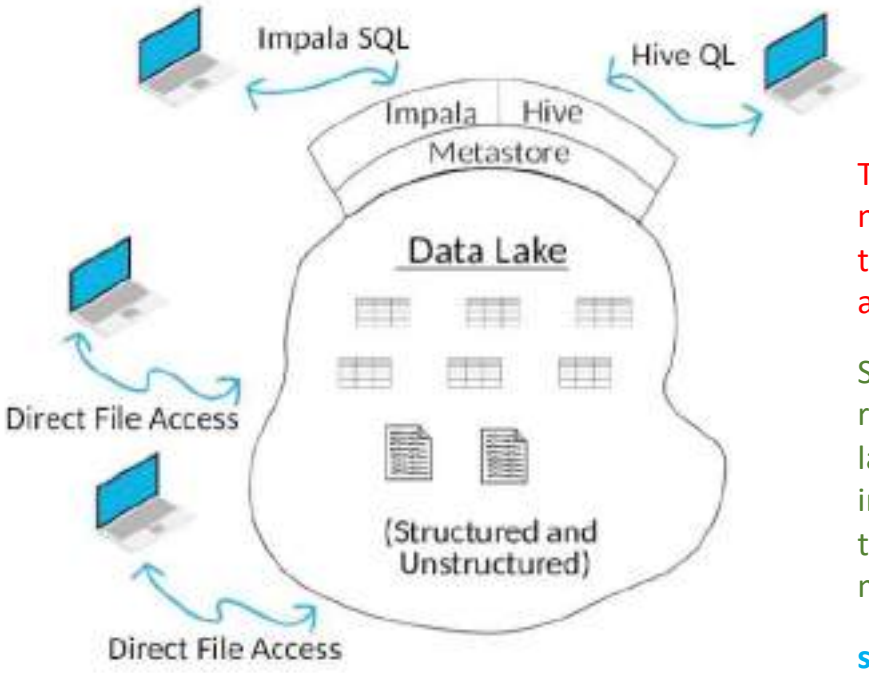
table_columns

table_name	column	data_type	…
toy	id	INT	…
toy	name	STRING	…
toy	price	DECIMAL(5,2)	…
maker	id	INT	…
…	…	…	…

toy

id	name	price
21	Lite-Brite	14.47
22	Mr. Potato Head	11.50
23	Etch A Sketch	29.35

Storing Data in a Big Data Store



The metastore is not kept directly in the data lake, but alongside it.

Some files may reside in the data lake without any information about them in the metastore at all.

schema on read

Operation of Apache Hive and Apache Impala

Apache Hive

- Hive translates SQL select statements to MapReduce programs

```
SELECT count(cust_id)
    FROM customers
    WHERE zipcode='94305';
```

Apache Hive

- Hive translates SQL select statements to MapReduce programs
- Since late 2015, Hive has the option to produce Apache Spark programs as well

```
SELECT count(cust_id)
    FROM customers
    WHERE zipcode='94305';
```

Use Case for Hive

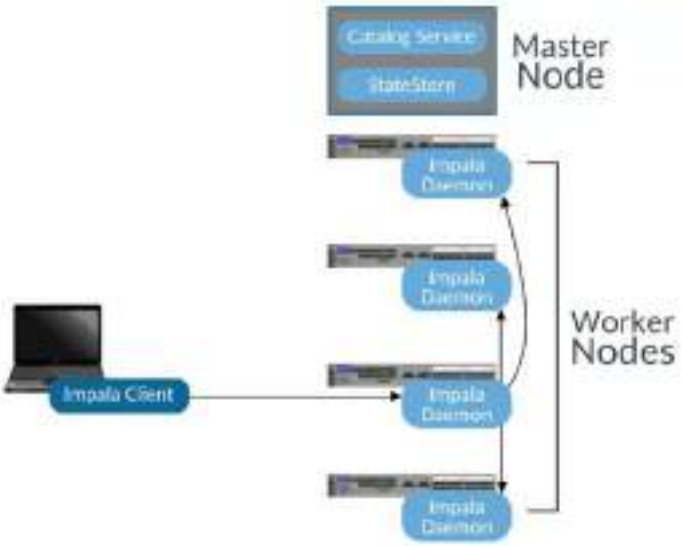
- When you want to process large amounts of data in a cluster using SQL,
  - and especially when the data you produce will be large as well,
  - and you want to store the result back in your cluster.
- E.g., create a copy of a billion row table with names converted to uppercase.
- Note that, Hive distributed programs are fault tolerant.

id	name	birthdate	…
8392	Lamija Tomčić	03-20-1985	…
1032	jaxon lynton	Aug 18, 1949	…
89	Floor ACHTERBURG	4/16/2002	…

id	name	birthdate	…
8392	LAMIJA TOMČIĆ	03/20/1985	…
1032	JAXON LYNTON	08/18/1949	…
89	FLOOR ACHTERBURG	04/16/2002	…

Apache Impala

- Built from the ground up as a distributed SQL engine for big data.
- Runs as a collection of Impala daemons running in a cluster.
- Client connects to any Impala daemon in the cluster, and sends its select statement
- This daemon coordinates with the other Impala daemons, and they collectively read and process the data needed to perform the query



Hive vs. Impala

- Impala SQL and Hive QL are about 95% the same.
- An Impala query can run 10 or even 50 times faster than the same query in Hive.
- Hive provides reliability and fault tolerance.
- Impala always opts for speed as a priority, even a priority over fault tolerance.

''We use Hive to prepare our data, and Impala to serve our data to users.''

Analyzing Big Data with SQL

Exploring Data in Hue

Cloudera Sandboxes
For VirtualBox
and VMware

https://s3.amazonaws.com/trainingvms/Cloudera-Training-Coursera-VM-cdh5.13.3c-virtualbox.zip

https://s3.amazonaws.com/trainingvms/Cloudera-Training-Coursera-VM-cdh5.13.3c-vmware.zip

Data Science Ethics

BBM467 Data Intensive Applications
Dr. Fuat Akal
akal@hacettepe.edu.tr

What are Ethics?

What are Ethics?

- Ethics are shared principles that help us distinguish right from wrong.
- Ethics are the cornerstone of civilization.
- Ethics are the basis for the rules we all voluntarily choose to follow because that makes the world a better place for all of us.

Ethics => No Theft

- Ethical principles stop me from stealing your wallet
  - *Even if you are smaller and weaker*
  - *Even if there is no chance I will be caught.*

Ethics are not Religion

- Most religions promote ethical behavior.
- “Thou Shalt Not Steal”
- But ethics need not be religious.
- Ethics flow from shared values, which could be on account of religion, or not.

Ethics are not Laws

- Suppose you tell me a secret and I promise to tell no one. If I then break my promise, I may not have broken any law, but I have certainly been unethical.

Ethics => Laws

- Ethics guide the creation of laws, so the two are often in consonance.
- Not everyone will be ethical, even when there are shared values
  - *That is why people go to jail for theft*
- Laws may be used to enforce ethical behavior.

Economic Benefits

- For the most part, society as a whole does best when each individual works to maximize their own individual benefit.
- But there are situations where the individual benefit comes at a cost to society.
- Shared societal value systems work to address these situations.

Tragedy of the Commons

Tragedy of the Commons

Summary: Economic Value

Cost to me of following rule

Is less than

Benefit to me from others following rule

Should make me want the rule, even if there is a cost to me

Benefits from Following Rules

- If we all agree to drive on the right hand side of the road,
  - we can all drive much faster than if we didn't have that rule. – But only if we all follow the rule!!
- If we all agree not to litter, we can enjoy clean roads and sidewalks.

Litter

- Would you discard your candy wrapper here?
- Or, would you walk it to the trash bin??

Summary: What are Ethics?

- Ethicsaresharedrulesweallagreetofollow because of the resulting benefits.
  - *Infused with a sense of right and wrong.*

Data Science Needs Ethics

- When the Wes Was Wild
  - There were few rules
  - You took what you could

Civilizations are Based on Rules

- We can no longer do everything we can.
- We have to follow rules.

The Possibilities Today

- We have unprecedented access to data
- We have unprecedented options for analysis
- There is virtually no limit to what Data Science could do
- Should we agree not to do some things that are possible?

Data Science has Impact

- There is tremendous excitement about Data Science precisely because of the many ways in which it provides us with a “better” way to do something.
- But there are possible undesired consequences, for privacy, fairness, etc.
- How do we decide what is OK to do?
- Ethics

Unsolicited Commercial Mails

- Today, in the US, no upstanding business will own up to spamming intentionally.

Data Science Needs Ethics

- While Data Science can help in so many ways, it can hurt too.
- By developing a shared sense of ethical values we can reap benefits while minimizing harms.

Human Subjects Research and Informed Consent – Part I

Tuskegee Study

- 1932-1972 in Macon County Alabama
- 600 impoverished black sharecroppers
- 399 with syphilis, 201 without
- Study effects of untreated syphilis
- Given free health-care and hot meals

Tuskegee Study (contd)

- Penicillin became standard
- treatment for syphilis in 1947
- Subjects were never informed
- Public outcry in 1972 ended study
- Led to IRBs

Informed Consent

- Human Subject must be
  - *Informed about the experiment*
  - *Must consent to the experiment*
    - *Voluntarily,withoutcoercion*
  - *Must have the right to withdraw consent at any time*

Informed Consent

- Human Subject must be
  - *Informed about the experiment* ***NO***
  - *Must consent to the experiment* ***YES***
    - *Voluntarily,withoutcoercion*
  - *Must have the right to withdraw consent at any time* ***NOT QUITE***

- Benefit vs Harm was assessed by experimenter

Instituitional Review Board

- Comprises diverse members, including non-scientists
- Approves human subjects study
- Balances potential harm to subjects against benefits to science
- Manages necessary violations of informed consent
  - *E.g. in psychology experiments*

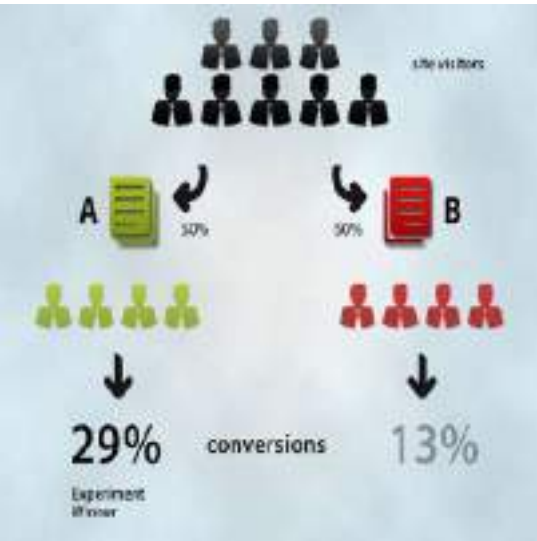
Human Subjects Research and Informed Consent – Part II

Prospective Data Collection

- IRB review required
- Driven by excesses of medical research
- Also for social science research involving human subjects
- Enforced by US federal law only for government funded experiments.
  - *Somestatesgofarther.*
- Applies to user studies in CS!!
- Not required if it is "business" and not "research".

A/B Testing

- Routinely used
- Particularly by companies with high customer volume
- Particularly for web-page design



Human Subjects Research?

- Web Companies do A/B testing all the time. Human users are impacted.
- Companies try to “push our buttons” to buy more.
  - *This is considered good business.*
- Companies can have imperfect algorithms that give us poor results.

Facebook/Cornell Experiment

- Choose articles shown in newsfeed to manipulate user mood: demonstrates “emotional contagion” (in 2012)
- Results published in PNAS (in 2014)
  - Proceedings of the National Academy of Sciences of the United States of America
- Users did not know their newsfeed was manipulated
- But Facebook regularly tinkers with the algorithm to choose items for your newsfeed
- User agreement permits Facebook to use data for research (explicitly so after 2012).

Human Subjects Research?

- Companies should not lie to us intentionally, even if effects are similar.
- OKCupid: Reported False Compatibility.

We Experiment On Human Beings!

July 28th, 2014 by Christian Rudder

Limitations of Informed Consent

Is it Informed and Voluntary?

- "Informed" is based on something hidden in multiple pages of fine print.
  - *Unclear value in law*
  - *E.g. Liability Waivers for Sports Activities*
- "Voluntary" in spite of the requirement of consent to obtain a desired action at the time of the action
  - *E.g. to use a software service*
  - *E.g. to buy product.*

Informed Consent -- Facebook

- Facebook explicitly tells user in it agreement that it may collect user data for research purposes.
- Facebook may have met the letter of its agreement, even when it conducted the mood contagion experiment.
- But got hammered by its users nonetheless.

Repurposing Can be Problematic

What is the data actually gonna be used for?

- I give data about myself to a merchant to obtain a specific service.
- I do not want the merchant :
  - *To use these data for other purposes*
  - *To share these data with others*
- Consent can often be limited to disallow repurposing.
- “Context” matters

Repurposing is not all Bad

- I understand that my credit card company collects data about my purchases and payments, and I am willing to accept that they also share some of this with a credit reporting agency, even if I don't particularly like it.
- (Ideally) the card company tells me about this intended sharing, and I (may be reluctantly) agree.

Repurposing is Often Necessary

- I share medical data with my hospital to get better care, but I may gladly support its repurposed use for medical research.
- However, the specific research questions may not be known at the time I receive my care.
  - *Retrospective Data Analysis, i.e. Data is already collected*
- Can we define a consent agreement broad enough to cover a range of possible medical research without being so broad as to cover "everything under the sun"?

Humans and Data

- Most data of interest are:
  - *By humans*
  - *About humans, OR*
  - *Affect humans*
- So we have to consider this impact as we practice Data Science

Attributions

- “Tuskegee Study Participants” is courtesy of the National Archives
- “Tuskegee Blood Draw” is courtesy of the National Archives
- “AB Testing” by RaHuL Rodriguez is licensed CC BY-SA
- Headline from a Website article about OKCupid is copied as fair use

Data Ownership

Biography

- OK, it is about you.
  - *But is it yours?*
- If I write your biography, I own copyright. If you dislike what I say, not much you can do, except sue for libel where I am inaccurate.

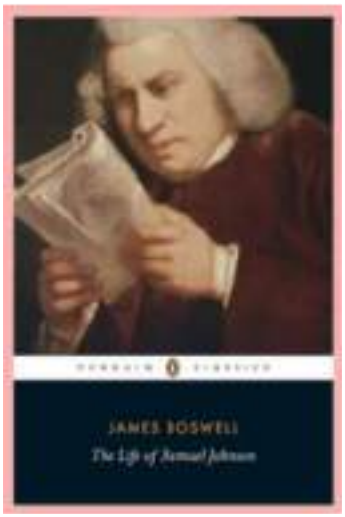
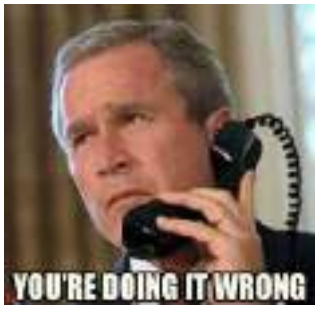


Photo Ownership

- If I photograph you, I own the photo.
- Limits on me can be:
  - *On taking the photo in certain private areas*
    - *In your home*
    - *In a bathroom in mystore*
  - *On using the photo in certain ways*
    - *As implied endorsement*
    - *As implied libel*

Unflattering Photos



- An altered photograph is an easy way to create a juxtaposition that is misleading or demeaning.

- But unaltered photographs can do that too, if clicked at the "right" time.
  - *Each of us has had moments when we did not look our best.*

Unflattering Photos (contd.)

- With cameras everywhere today, unflattering moments are all too easy to capture.
- This can be implied libel.

Data Ownership

- Similar limits on what data I can record about you and what I can do with it after
- Free to record and free to use otherwise.

- And we have done this forever ...
  - *Recommendation Letters*
  - *Gossip*

Intellectual Property Basics

- Copyright
  - *Artistic expression*
  - *A rearrangement is a derivative work*
- Patent
  - *Idea for making or doing*
- Trade Secret
  - *I have it, but don't tell anyone*

Use of Data

- If I use an image, I display it, and can credit the owner.
- If I use your data, I am almost always taking some piece of what you know, merging it with what I know, and expressing something new.
  - *At best I can say I used some data from you.*
  - *Not easy to say exactly what and exactly how much.*

Digitized Data Can Preserve Culture

- Books, photos, movies, ...
- Materials that are “out of print” can be lost for ever.
- Ancient scripts, cultural artifacts

Digitized Data Can Propogate Culture

- One copy of a book at a library means that a patron has to trudge to the specific library location, or the book has to be transferred to the right location.
- One digital copy can be “loaned out” to one library user at a time anywhere in the world.
- Library still buys one copy, so publisher gets paid.
- Library enforces rule of only one reader check out at a time.

Data Collection and Curation

- It can take a great deal of effort to collect data about some thing or someone.
- It can often take even more effort to clean, validate, and standardize the collected data to place it into a form that is of immediate value.
- Whoever does this work deserves credit, and has ownership of the data “asset”.

Wikipedia

- Do not need an artistic creator to claim ownership.
- Crowd-sourced data will also do.
- Individual contributors do not own this.

Crowd-Sourced Reputation Sites

yelp

Epinions

Rotten Tomatoes

RATE MY PROFESSORS

tripadvisor

All completely have their business model reliant upon people in the community expressing their opinion about businesses that they're rating on these sites.

Limits on Recording

- Recording is wrong when there is reasonable expectation of privacy
  - *E.g. no cameras in clothing store fitting room*
- Similarly:
  - *Phone company must not record the (content of) phone calls.*

We Could Agree Otherwise

- You could pose for a photograph under an agreement that I will give you ownership of the photo.
- You could agree to participate in a research experiment under an agreement that has *informed consent*.
- Funding agency could require that data collected from your funded research be made public.

Video Cameras in Stores

- Can provide security
- Can even be used to improve placement of products
- But should not be published

Cell Phone Location Tracking

- Necessary to provide service.
- Required for many valuable applications.
- But can result in a huge loss of privacy.

Limits on Use

- Often there is a strong reason to record data, but also potential for misuse.
- Prefer to limit use rather than recording
- Allows desired legitimate use while disallowing other (undesirable) use.

Government Surveillance

- Never know what you will need.
- So suck in a lot of data.
- Plan never to look at it.
- Until there is a specific need and authorization.
- So it is important to separate recording from use.

Data Destruction

- Companies legitimately collect data as part of doing business.
- Companies need to retain goodwill of customers, and so will try not to do egregiously bad things with the data.
- Once the company ceases to do business (e.g. because of bankruptcy), this data is an asset that is likely sold to a third party who intends to misuse.
- Collected data must be destroyed, not sold.

Bankruptcy Law has Partial Protection

Bankruptcy law actually now has partial protection, basically saying that, whatever is the company's privacy policy, must survive even after the company goes bankrupt.

To Sum Up: Data Ownership

- Data ownership is complex,
  - *Even for things you want to share, e.g. Giving credit is also hard*
- For the most part you do not own data about you
  - *Need to create principles to control what isn't yours*
  - *Right to privacy*

Attributions

Carny Portrait (Times Two) George Norcus cc-by-nc-2.0
Book cover is licensed CC BY-NC
Carny by two by is licensed CC BY-NC
“Destiny” by Jordan Bintcliffe is licensed CC BY-NC-ND
“Sagging” by Tony Alter is licensed CC BY
“Walmart Fitting Room” by is licensed CC BY-NC-ND
Camera and Cellphone images from Microsoft Clip Art
Company Logos belong to respective companies

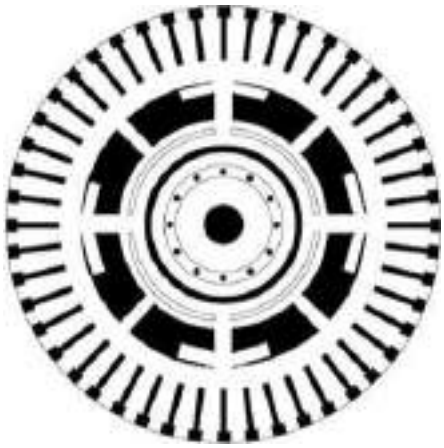
What is Privacy?

Argus Panoptes

- Giant with “a hundred” eyes in Greek myth.
- Was ever watchful.

Panopticon

- A prison where a single guard could observe all the prisoners.
- But prisoners would not know when they were being watched, so would “behave” all the time.
- Jeremy Bentham, 18th Cent.



Modern-Day Panopticon

Also, Even If Less Obvious

- OK, Facebook does not really watch everything you do.
- But how about the internet as a whole?
- Do you care if you are doing nothing wrong?

Technology is Eroding Privacy

- "You have zero privacy any way ... Get over it."
  - *Scott McNealy, CEO of SUN, 1999*
- Some dream of a world where everyone is completely honest because there is zero privacy.

Privacy is a Basic Human Need

- Even for people who have "nothing to hide"

Privacy also has Societal Benefits

- Anonymous voting is central to democracy

Voluntary Disclosure

- Anything you voluntarily disclose to others has much less protection than anything you keep completely to yourself.
- E.g. telephone company has to know what number you dialed to be able to place the call. So you have disclosed this metadata, even if not the content of your conversation.
- Quickly becomes problematic – due to ubiquity.
  - What is voluntary, what is not?

It Takes a Village

- In small towns, there was little privacy
  - *Everyone in town knew "everything" about everyone else*
- Big cities provide anonymity.
- Does information technology bring us back to the halcyon days in a small town?

Changing Attitude

- As society adapts to new technologies, attitudes will change.
  - *My kids freely post facts about their activities online that I never would dream of doing*
  - *Their privacy boundaries are different from mine*
- But different boundaries does not mean no boundaries.

No Option to Exit

- In the past, one could get a fresh start by:
  - *Moving to a new place*
  - *Waiting till the past fades*
    - *Reputations can be rebuilt overtime.*
- Big Data is universal and never forgets anything!!
- In a small town, information is (mostly) symmetric
  - *You know as much about me as I know about you.*
- Data Science can result in major asymmetries.

Wayback Machine

- Archives pages on the web
  - *Almost everything that is accessible (not password protected).*
  - *(intends to) retain for ever*
- If you have an unflattering page written about you, it will survive for ever in an archive, even if it was taken down!!

Right to be Forgotten

- Laws are often written to clear a person's record after some years.
  - *Expunge records about imprisonment, driving accidents, ...*
- If such information is on the web, how to remove?
- Law in EU and Argentina since 2006.
- In practice, primarily impacts search engines.

Degrees of Privacy

Privacy is not Non-Disclosure

- I tell my doctor about my health condition
  - *That does not let the doctor make these facts public*
- I share a deeply personal story with a friend
  - *That does not mean I make the story public*
- But I am OK with limited additional disclosure:
  - *The doctor may discuss my case with other caregivers.*
  - *My friend may tell my story to someone else to get me help.*

The World of Data Today

- Every merchant, and many others, has considerable data about me.
  - *At least my interactions with them*
  - *But often much more*
- I am happy for them to use what they know to serve me better.
- But I really do not want them to share with others, or use the data in other ways.
  - This is where privacy comes in

Privacy is not Non-Disclosure
It is Exercise of Control

Prevention of undesired disclosure is one crucial element of maintaining control

All or Nothing Agreements

- Many user agreements are “all-or-nothing”.
  - *User must completely give up control on shared data to get any service benefits at all.*
- Users complain about loss of privacy
- Better to provide graduated choices, so users could make the tradeoffs they like.
  - *E.g. Incognito browsing*

Friends can Harm Privacy

- You go to a party and your friend posts photos from the party. Now "everyone" knows you went, including people you didn't want to tell.

Relatives can Harm Privacy

- You make a personal decision to make your DNA public "for science".
- Your blood relatives share much of your DNA.
- Their DNA is now semi-public.
- But they were not involved in your decision.

Collection versus Use

- Privacy is usually harmed only upon use of data.
- Collection is a necessary first step before use.
- Existence of collection can quickly lead to use.
- But collection without use may sometimes be right.
  - E.g. Surveillance: *By the time you know what you need it is too late to go back and get it.*

To Sum Up: Loss of Privacy

- Due to loss of control over personal data.
- I am OK with you having certain data about me that I have chosen to share with you or that is public, but I really do not want you to share my data in ways that I do not approve.

Modern Privacy Risks

Main Drivers of Privacy Violation

- Surveillance
  - *Government agencies for national security*
  - *But also private enterprises, private investigators, ...*
- Advertising
  - *If a company can show you a focused, personalized, relevant ad, everyone wins.*
- Introduction
  - *Prospective employee, prospective borrower, prospective date, ...*

Main Sources of Information

- Data collected by merchants and service providers
- Activity tracking – on and off the web
- Sensors in devices around us
  - *Personal devices*
  - *Infrastructure*
  - *Third party, e.g., a friend's camera*

Open Government vs Privacy

- There are many benefits to government being open.
  - *Many government entities publish detailed records*
  - *Freedom of Information Act (FOIA: in the US) provides access to many unpublished records (with exceptions for privacy).*
- But most government records concern citizens.
- So privacy concerns have to be balanced against openness.

Driver License

- Driver License and Voter Registration databases are often semi-public.
- Can provide age, sex, address attributes.
  - *And also much identifying information.*
- Regularly used by commercial interests.

Data Brokers

- Companies that aggregate and link information from multiple sources to create more complete, and hence more valuable, information products.
- Many don't realize what can be learned about them by linking multiple sources.

Waste Data Collection

- Your ID is scanned at the bar to confirm proof of age.
- The scanning computer at the bar captures your name, address, date-of-birth, etc.

Metadata

- Data about the data.
- Often distinguished from data content.
- E.g. for a phone call, metadata includes
  - *Caller*
  - *Callee*
  - *Time and Date of Call*
  - *Duration*

Metadata May Carry Much Information

- Location information may be metadata for a cell phone call.
- Knowing location does not reveal content of call.
- But location tracking can reveal a great deal about the person.

Do not Under-Estimate Analysis

- A smart water meter at your house can recognize “signatures” of water use every time you flush the toilet, take a shower, or wash clothes.
- There have been attacks on encrypted data based on observation of power consumed by chips performing encryption.

Trust => Design

- Traditional social norms dealt with privacy by trust:
  - *You tell me private things because you trust me not to use what you told me in ways you would not approve.*
- Modern data systems must deal with privacy by design.
  - *Too many players in a complex system*
  - *Data sharing is contractual and not based on trust*

Conclusison

- Privacy is the ability to control sharing of information about self.
- Basic human need.
  - *Even for people who have "nothing to hide"*
- Privacy is easily eroded by thoughtless actions.
  - *And also by intention, in some cases.*
- Stakeholders have begun a conversation around privacy.
- But there isn't yet consensus on where the lines should be drawn.

Attributions

- “panoptes_mollusca_bw” by Kendra Bentle Rainey is licensed CC BY-NC-ND
- “Panopticon” by schadenfreude lola is licensed CC BY-ND

- “Surveillance Cameras on Wide Screen” by Exacq is licensed CC BY-NC
- “Facebook: The Panopticon of Modern Age” by joelle L is licensed CC BY
  “Funny Toilet Logo” by guccio is licensed CC BY-NC
- “Voting” by Sarah PAC is licensed CC BY-ND
- “Re_exposure_of_27” by Phil Campbell is licensed CC BY
- “Finger ID Scanner” by Ian Dexter Marquez is licensed CC BY-SA
- Portrait Images used are public domain

Acknowledgement

- All slides were taken from the **Data Science Ethics** course by **H.V. Jagadish**, University of Michigan

- The original full course is available at Coursera
  - https://www.coursera.org/learn/data-science-ethics

UNIVERSITY OF MICHIGAN