# Data Science

2022. 8

Yongjin Jeong, KwangWoon University

# What is Data Science?

- **Definition (from Wikipedia)**
  - ✓ concept to unify statistics, data analysis, machine learning, domain knowledge and their related methods in order to understand and analyze actual phenomena with data
  - ✓ the application of computational and statistical techniques to address or gain insight into some problem in the real world (Pat Virtue, CMU)
  - ✓ It uses techniques and theories drawn from many fields within the context of mathematics, statistics, computer science, domain knowledge, and information science
- **Components of Data Science**
  - ✓ Software Programming -> Data mining, Database
  - ✓ Statistics/mathematical modeling -> Machine Learning, Scientific Computing
  - ✓ Domain Knowledge -> Data driven business analytics
- **Data Science** (Pat Virtue, CMU)
  - ✓ = **statistics** + **data processing** + **machine learning** + **scientific inquiry** + **visualization** + **business analytics** + **big data** + …

# List of Best Jobs

## 50 Best jobs in America for 2022

(www.glassdorr.com/List/Best-jobs-in-America)

| | Job Title | Median Base Salary |
|---|---|---|
| #1 | Enterprise Architect | $144,997 |
| #2 | Full Stack Engineer | $101,794 |
| #3 | Data Scientist | $120,000 |
| #4 | Devops Engineer | $120,095 |
| #5 | Strategy Manager | $140,000 |
| #6 | Machine Learning Engineer | $130,489 |
| #7 | Data Engineer | $113,960 |
| #8 | Software Engineer | $116,638 |
| #9 | Java Developer | $107,099 |
| #10 | Product Manager | $125,317 |

## 10 professions will be most sought-after 2025

(www.businesstoday.in)

Which job is best for future 2025?

**These 10 professions will be most sought-after by 2025**

- Data Analysts and Scientists. The role of data analysts and scientists is picking up steam currently. ...
- AI and Machine Learning Specialists. ...
- Process Automation Specialists. ...
- Information Security Analysts. ...
- Software and Applications Developers.

Mar 17, 2021

https://www.businesstoday.in › PANORAMA

These 10 professions will be most sought-aft

## Best jobs in America in 2022

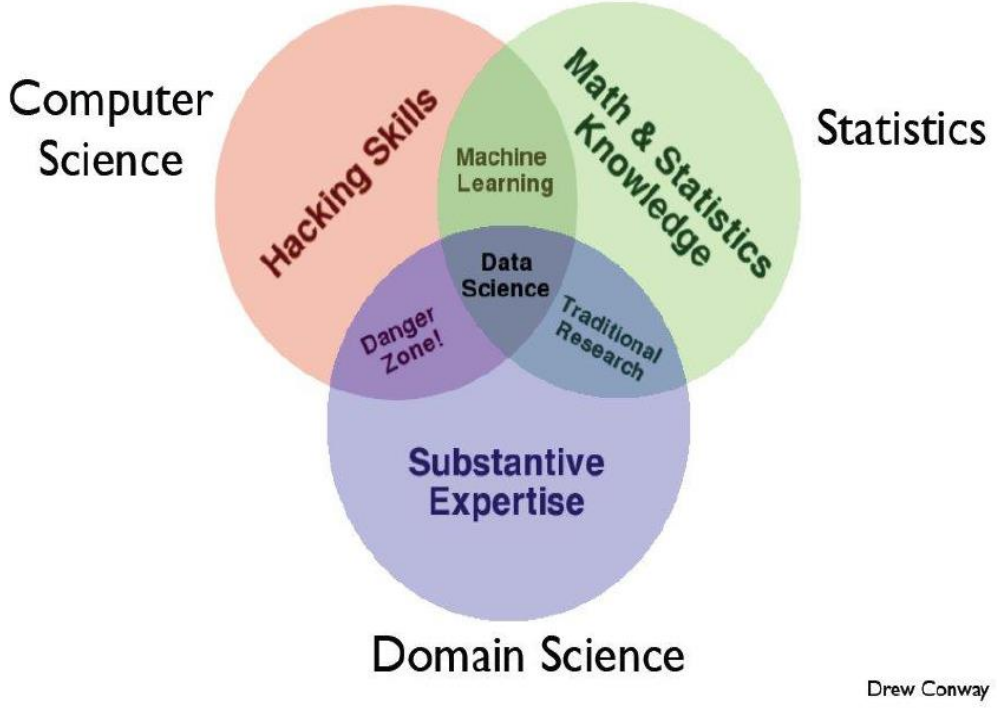
(usnews.com)

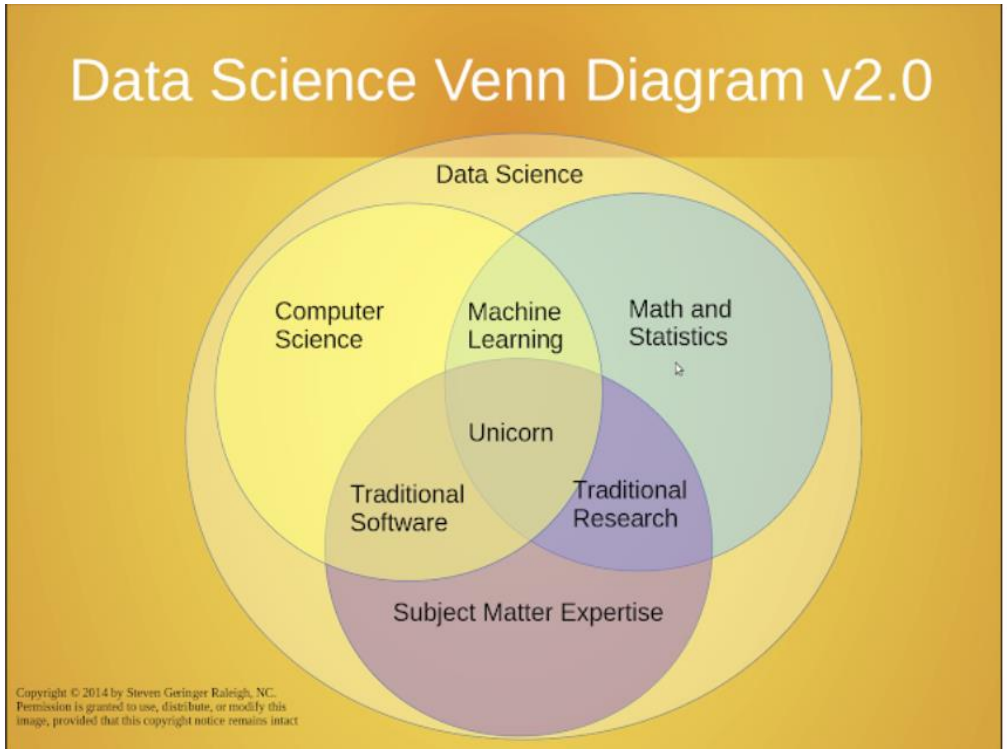

The 10 Best Jobs in America in 2022:

1. Information security analyst.
2. Nurse practitioner.
3. Physician assistant.
4. Medical and health services manager.
5. Software developer.
6. Data scientist.
7. Financial manager.
8. Statistician.
9. Lawyer.
10. Speech-language pathologist.

# What is Data Science? – One definition

- **Venn Diagrams (Drew Conway 2010, Steven Geringer 2014)**



# Data in Data Science

- **Contrast to Databases**

| | Databases | Data Science |
|---|---|---|
| data values | "Precious" | "Cheap" |
| data volume | Modest | Massive |
| examples | Bank Records,<br>Personal Records,<br>Census,<br>Medical Records | On line clicks,<br>GPS logs,<br>Tweets,<br>Web surfing,<br>building censor readings |
| structured | Strongly (Schema) | Weak or None (Text) |
| priority | Consistency,<br>Error recovery,<br>auditability | Speed,<br>Availability,<br>Query richness |
| realizations | SQL | No SQL<br>Python, R, TensorFlow, Keras |
| | Querying the Past | Querying the Future |

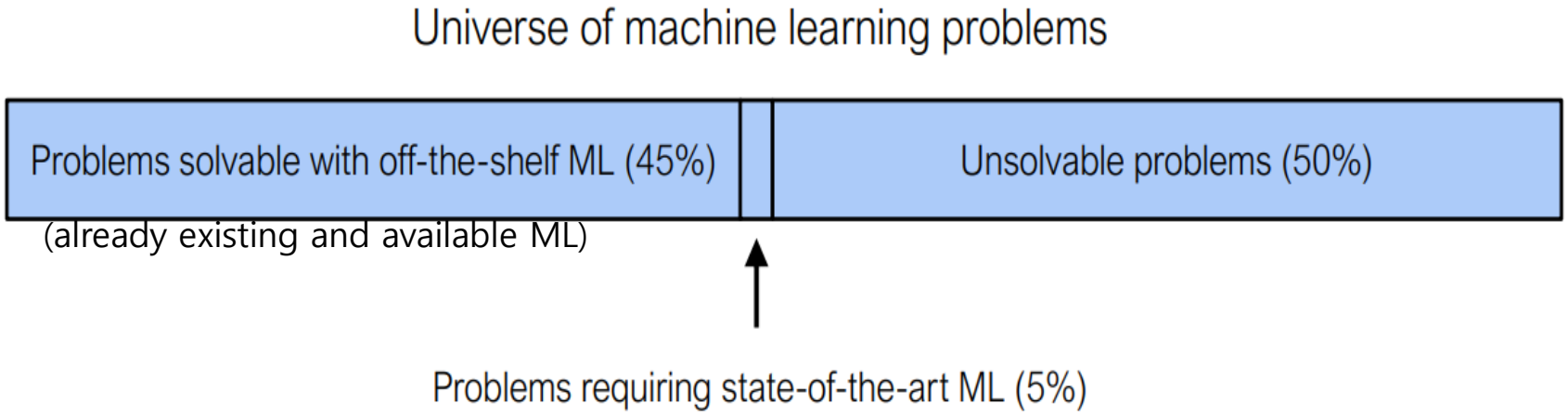
ref: CS194 berkeley.edu by John Canny

# Statistics vs. Data Science

| | Statistics | Data science |
|---|---|---|
| common | aim to extract knowledge from data<br>- exploratory data analysis & Visualization<br>- characterization & prediction | |
| language | Estimating (inferencing)<br>data point/observation<br>independent variable<br>dependent variable<br>dummy variable | learning (training) & prediction<br>example/instance/sample<br>feature<br>label or target<br>one-hot encoding |
| data | small or medium sized<br>mostly structured<br>more manual data collection (or surveys)<br>In general, no web scraping or data processing | huge (big data)<br>structured or unstructured<br>more data collection/acquisition (from web and SNS) |
| processing | query (past) | predict (future) |
| tools | Mathematics<br>prefer R<br>SAS (statistics package) | programming (prefer Python)<br>ML libraries (sklearn, tensorflow, etc.) |

# Some Comments from Experts

- **Data science is not machine learning** (Pat Virtue, CMU)
  - Machine learning involves computation and statistics, but has not (traditionally) been very concerned about answering scientific questions
  - Machine learning has a heavy focus on fancy algorithms…
  - … but sometimes the best way to solve a problem is just by visualizing the data, for instance.



# Some Comments from Experts

- **Data science is not big data** (Pat Virtue, CMU)
  - Sometimes, in order to truly understand and answer your question, you need massive amounts of data…
  - …But sometimes you don't
  - **Don't create more work for yourself than you need to !**

# Data Science Applications and Examples

- (ref) https://builtin.com/data-science/data-science-applications-examples
- **Healthcare**
  - Google: machine learning for metastasis (identifying breast cancer)
  - CLUE, Germany: predict periods and forecast conditions for pregnancy
  - Oncora Medical: cancer care recommendations
- **Road Travel**
  - UPS: optimizing package routing (save up to $200 million)
  - Streetlight data: traffic patterns for cars, bikes, and pedestrians (use for commuter transit design)
  - Uber Eats (Uber's delivery app): optimize full delivery process
- **Sports**
  - Liverpool F.C.: recruited undervalued soccer players
  - RSPCT: basketball-coaching sensor (shooting analysis system)
  - British Olympic Rowing team: model athlete evolution and find a promising newbie rower

# Data Science Applications and Examples

- **Government**
  - Equivant: data-driven crime prediction
  - ICE (Immigrations and Customs Enforcement): facial recognition in ID databases
  - IRS: tax-fraud detection
- **E-commerce**
  - SOVRN: automated AD placement (target campaigns to custmoers)
  - Instagram: convert users' likes and comments, their usage of other apps and their web history into predictions about the products they might buy
  - Airbnb: search that highlights areas of cool neighborhoods (high density of bookings)
- **Social life**
  - Tinder (most popular dating app): find a good match for singles
  - Facebook: "people you may know" sidebar (based on friend list, photos, schools, etc.)

# www.kaggle.com

- **What is Kaggle?**
  - Owned by Google, and over 3 million data scientist registered.
  - The world's largest data science and machine learning community with powerful tools and resource. (over 50,000 public datasets and 400,000 public notebooks)
  - You can find and publish data sets, and all data sets are **free**.
  - Can participate competitions to solve data science challenges.
  - Provide self-learning courses (from Python to Deep Learning)
  - Explore and run machine learning code with Kaggle Notebooks (with source codes).
  - Can discuss any data science issues with experts.
  - Try it at https://www.kaggle.com

# Data in Korea

- 데이터 이용을 활성화하기 위한 데이터 3법 통과 (2020.1)
  - 개인정보 보호법, 정보통신망법, 신용정보법
  - 핵심: 가명정보를 통계작성, 연구, 공익적 기록 보존 용도로 본인 동의없이 활용 가능
- 가명정보 (pseudonym or alias) (예: https://brunch.co.kr/@jaeyunchoi/18)

| | 개념 | 예시 | 활용가능 범위 |
|---|---|---|---|
| 개인정보 | 특정 개인에 관한 정보, 개인을 알아볼 수 있게 하는 정보 | 강하늘, 1990년 2월 21일생, 남성, 2019년 12월 신용카드 사용금액 150만 원 | 사전적, 구체적 동의를 받은 범위 내에서만 활용 가능 |
| 가명정보 | 추가정보의 사용 없이는 특정 개인을 알아볼 수 없게 조치한 정보 | 강XX, 1990년생, 남성, 2019년 12월 신용카드 사용금액 150만 원 | 개인정보 범위에 포함되나, 다음 목적에 한하여 동의없이 활용 가능<br>① 통계작성(상업적 목적 포함)<br>② 연구(상업적 연구 포함)<br>③ 공익적 기록보존 목적 등 |
| 익명정보 | 더 이상 개인을 알아볼 수 없게 (복원 불가능할 정도로) 조치한 정보 | 남성, 20대, 2019년 12월 신용카드 사용금액 100만 원 이상 | 개인정보가 아니므로 제한없이 자유롭게 활용 가능 |

# Different Views for Data

- **Simpson’s Paradox (심슨의 역설)**
  - 각 그룹 데이터에서 개별적으로 나타나는 특징과 전체의 경향이 달라지는 현상 (같은 데이터가 분석 방법에 따라 해석이 달라질 수 있음) -> 데이터의 분석에 주의
  - (예)

| 도시 | A사 | B사 |
|---|---|---|
| 서울 | 정상품 90, 불량품 10<br>(불량률 10%) | 정상품 920, 불량품 80<br>(불량률 8%) |
| 부산 | 정상품 980, 불량품 20<br>(불량률 2%) | 정상품 99, 불량품 1<br>(불량률 1%) |
| 전체 | A사 총 불량률 30/1,100<br>= 3% | B사 총 불량률 81/1,100<br>= 8% |

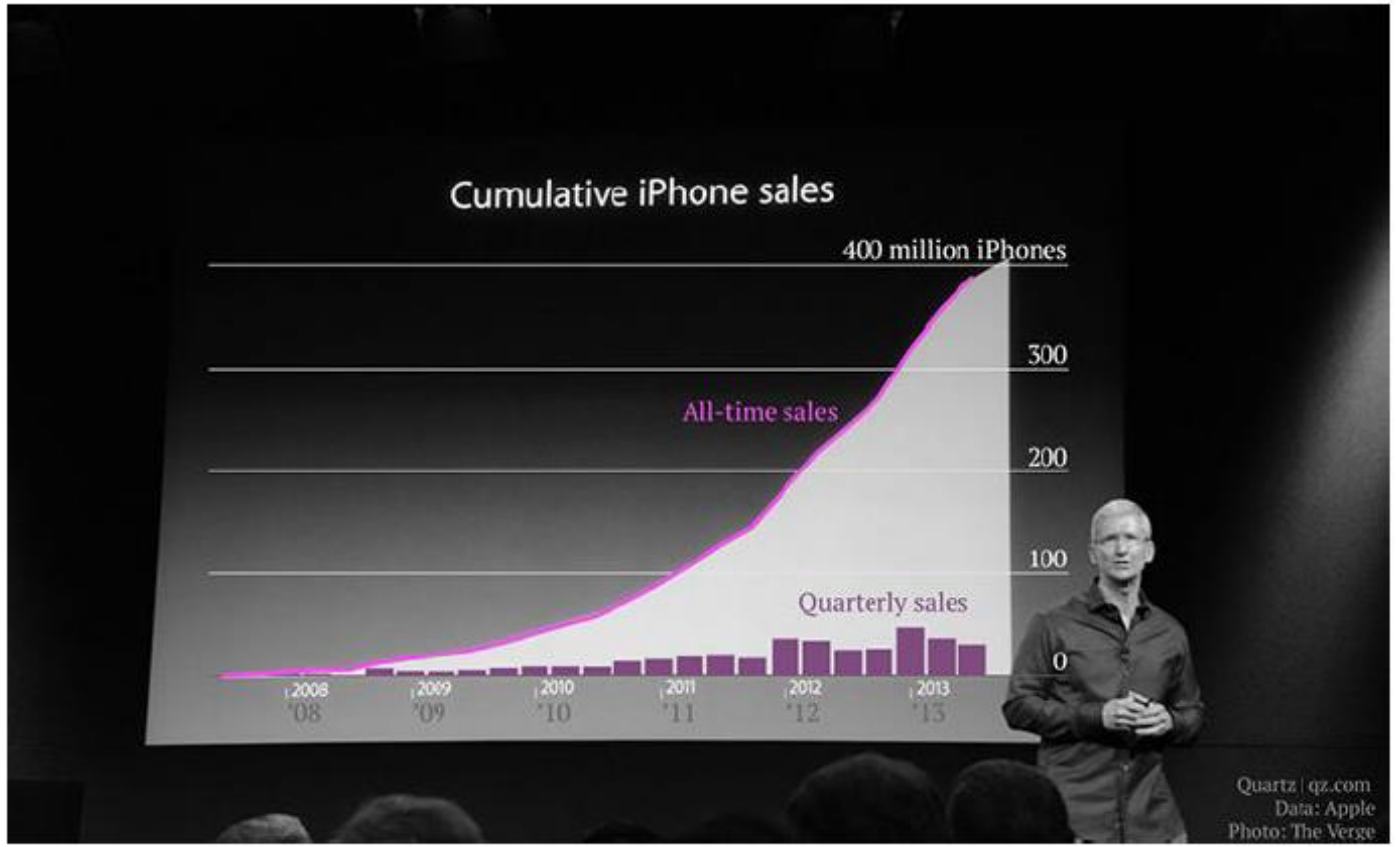
$$\left(\frac{a1}{A1} > \frac{b1}{B1}\right) \;\&\; \left(\frac{a2}{A2} > \frac{b2}{B2}\right) \overset{?}{\rightarrow\rightarrow} \frac{a1+a2}{A1+A2} > \frac{b1+b2}{B1+B2}$$

# Cumulative and Incremental Data

- Apple iPhone sales have been exploding, right?

# Cumulative and Incremental Data

- Cumulative distributions present a misleading view of growth rate



# Public datasets for data scientist

- **Dataset finders**
  - Google dataset search: contains over 25 million datasets (https://toolbox.google.com/datasetsearch)
  - Kaggle (www.kaggle.com)
  - UCI machine learning repository (https://archive.ics.uci.edu/ml)
  - VisualData (https://www.visualdata.io/discovery) : computer vision dataset
  - CMU Libraries (https://guides.library.cmu.edu/machine-learning/datasets)
  - NLP Database by Quantum Stat. (https://datasets.quantumstat.com): natural language processing, sentiment analysis dataset
  - US Government and official dataset (https://www.data.gov)
  - Korea government's public dataset (https://www.data.go.kr)
  - Eurostat: open data from the EU statistical office

# Public datasets for data scientist

- **Machine learning dataset**
  - Mall customers dataset : people visiting the mall (gender, id, age, income, spending score, etc.)
  - IRIS dataset: simple and beginner-friendly (flower petal and sepal width)
  - MNIST dataset: 60,000 training images and 10,000 testing images
  - Cifar-10: 60,000 32x32 color images with 10 classes
  - Boston housing dataset: collected by US Census Service
  - Fake news detection: 7,796 rows with 4 columns (news, title, news text, result)
  - Wine quality dataset: different chemical information about the wine
  - Titanic dataset: 891 training and 418 test passengers (name, age, sex, no of siblings abroad, etc.)
  - Credit card fraud detection dataset: recognize credit card transactions

# Public datasets for data scientist

- **Computer vision dataset**
  - xView: one of the most massive publicly available image dataset (images from complex scenes annotated using bounding boxes) (xviewdataset.org)
  - ImageNet: largest image dataset for computer vision, used in ILSVRC(ImageNet Large Scale Visual Recognition Challenge), more than 1.2 million images with 1,000 classes (hierarchically organized)
  - Kinetics-700: (https://deepmind.com/research/open-source/kinetics) a collection of large-scale, high-quality datasets of URL links of up to 650,000 video clips (human -object interactions)
  - Google open images dataset: more practical than ImageNet, ~9 million images annotated with labels over 6,000 categories
  - Cityscapes dataset: video sequences taken from 50 city streets
  - IMDB-Wiki dataset: dataset for face images with labeled gender and age
  - Color detection dataset
  - Stanford Dogs dataset

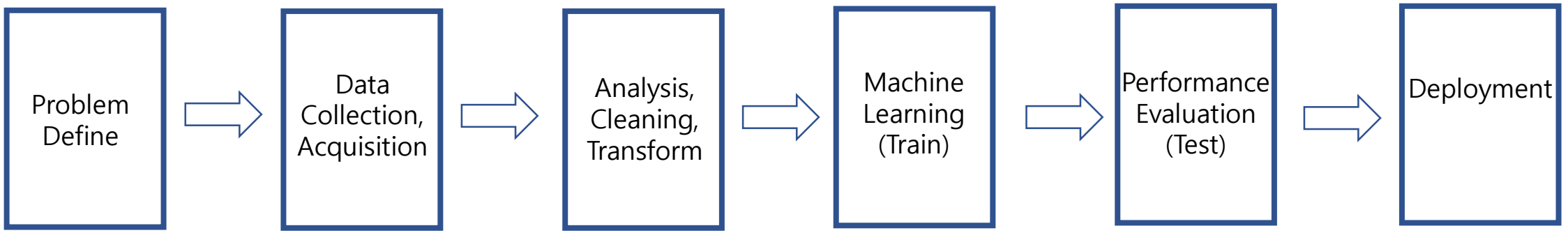
# Public datasets for data scientist

- **Self-driving  (autonomous driving) datasets**
  - Waymo open dataset (https://waymo.com/open/)
  - Berkeley DeepDrive BDD100k: for self-driving over 2,000 hours in NY and California
  - Bosch small traffic light dataset: traffic light for deep learning
  - WPI datasets: traffics lights, pedestrians, and lane detection
  - LISA: traffics signs, vehicle detection, and trajectory patterns
  - Comma.ai: car's speed, acceleration, steering angle, and GPS coordinates
  - Cityscape datasets: street scenes
- **Clinical datasets**
  - MaskedFace-Net: masked faces
  - COVID-19 datasets: from over 45,000 scholarly articles about COVID-19
  - MIMIC-III: from MIOT Lab for computational physiology from ~40,000 critical care patients

# Public datasets for data scientist

- **Sentiment analysis datasets**
  - Lexicoder Sentiment Dictionary: over 3,000 negative words and over 2,000 positive sentiment words
  - IMDB reviews: over 50,000 movie reviews from Kaggle
  - Stanford sentiment Treebank
  - Twitter US airline sentiment: twitter data on US airlines, classified as positive, negative, and neutral
- **Natural Language Processing (NLP) datasets**
  - The Big Bad NLP database: various natural language processing tasks
  - HotspotQA datasets: for explainable question answering systems
  - Amazon reviews
  - Rotten Tomatoes Reviews: 480,000 critic reviews (fresh or rotten)
  - SMS spam collection in English: 5,574 SMS spam messages
  - UCI spambase dataset: 4,601 emails with 57 meta-information about emails

# Data Science Work Flow




- Domain knowledge
- Business strategy

- String(structured)
- Text(unstructured)
- CSV/Excel
- JSON
- HTML/XML
- SNS
- Image, Voice

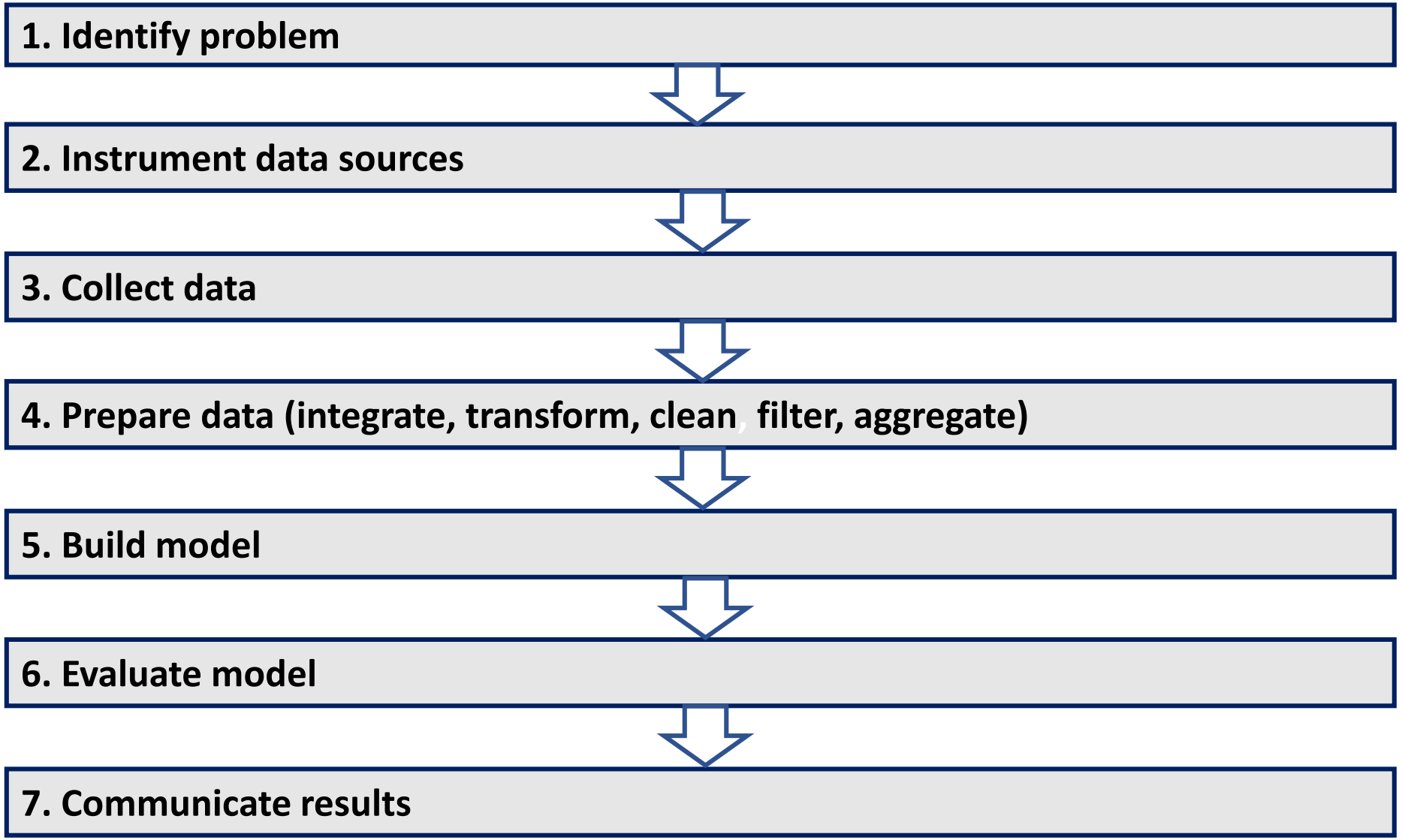
- Visualization
- Missing values
- Invalid values
- Outliers
- Categorical values
- Scaling
- Transform

- Supervised
- Unsupervised
- Error (or Loss)
- Bias and Variance
- Regularization
- CNN/RNN
- Generative model

- R-square
- Accuracy
- Precision/recall
- F-1 score
- ROC/AUC
- mAP
- IoU

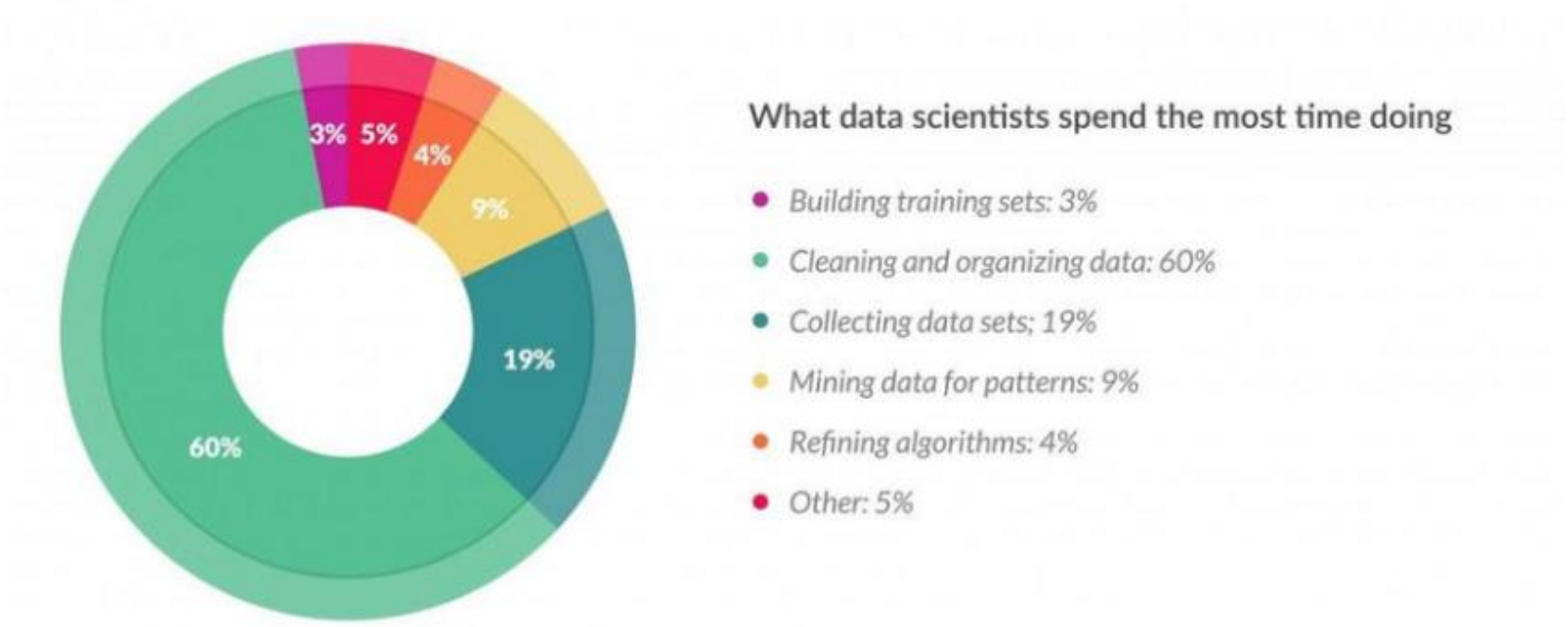
- Server
- Mobile

# Jeff Hammerbacher's Model



# Data Science Work Flow

- **According to a survey in Forbes,**
  - Data scientist spend 80+ % of their time on **data preparation**.




Source: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/

# Data Collection and Preprocessing

2021. 8

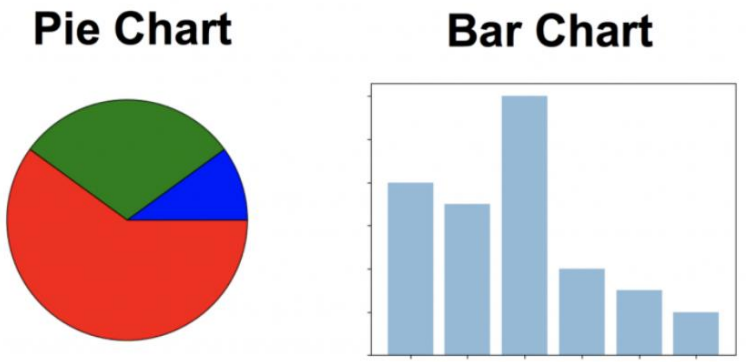
Yongjin Jeong, KwangWoon University

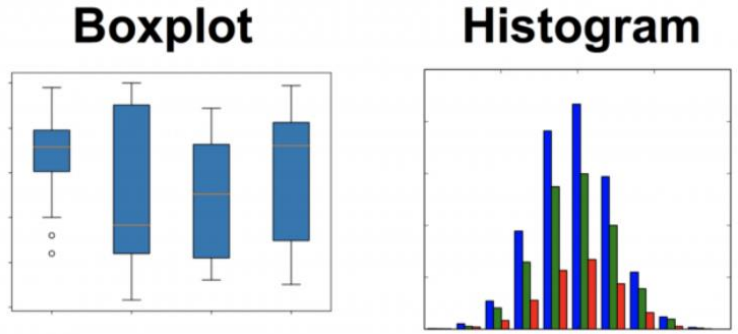
# Data types

- **Quantitative:** expressed as a number (**Numerical**)
  - Discrete Data: can not subdivide, and has limited number of possible values
    - Number of students in a class, number of workers in a company, number of correctly answered questions
  - Continuous Data: can be meaningfully divided
    - Interval Data: measurable and ordered, but have no meaningful zero
      - Can add/subtract, but can not multiply/divide/ratio
      - Because of no true zero, some descriptive and inferential statistics can not be applied
      - temperature
    - Ratios data: ordered and have a true zero
      - height, weight, length, speed
- **Qualitative:** can not be expressed as a number (**Categorical**)
  - Nominal Data: no particular order (referred as 'labels')
    - Gender (Women, Men), Hair color (Blonde, Brown, Red, etc.), Marital status (Married, Single, Widowed)
  - Ordinal Data: has particular order, but can not do arithmetic operations
    - Ranking (First, second, third, etc.), Economic status (low, medium, high), Clothes size (XL, L, M, S)

# Data types

- **Nominal data**
  - Mostly use One-hot encoding
  - Use Pie-chart or Bar chart to visualize
- **Ordinal data**
  - Use Label encoding (or Ordinal encoding) or One-hot encoding
  - Use Pie-chart, Bar chart, percentile, median, mode, IQR to summarize data
- **Continuous data**
  - Histogram: can check the central tendency, variability, kurtosis, etc. but no outliers
  - Box-plot shows outliers



# Data Collection and Acquisition

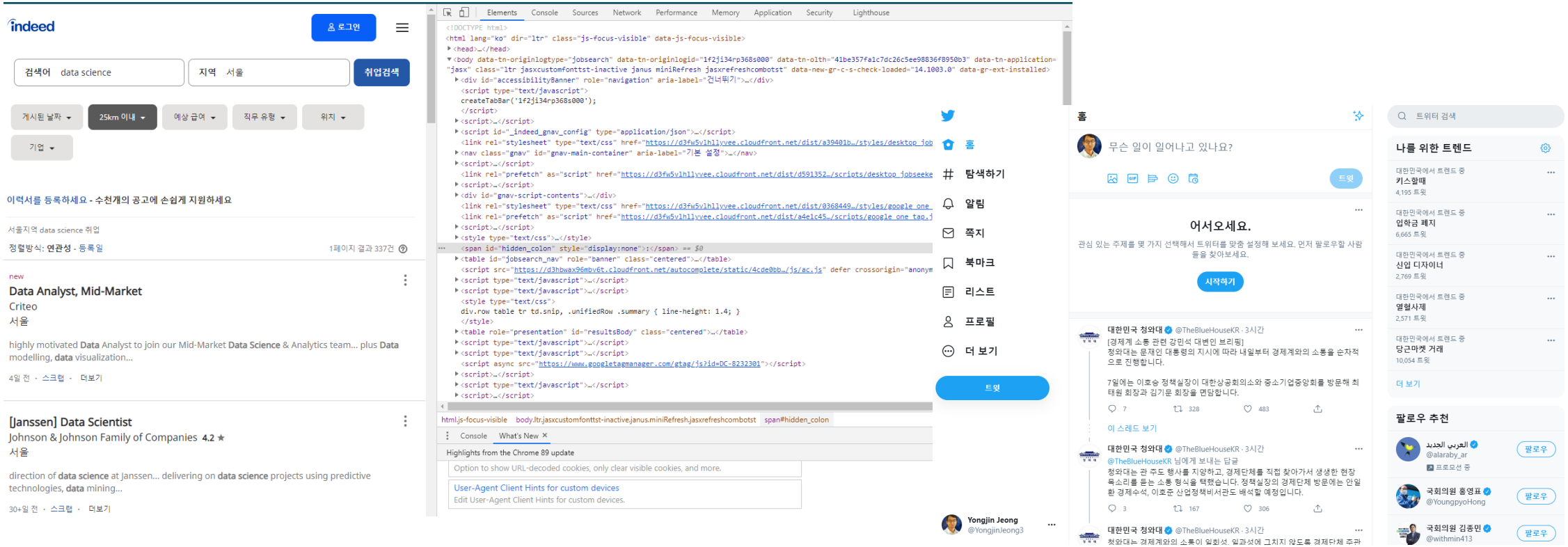
- **Collecting well-structured data**
  - ✓ String
  - ✓ CSV format: pd.read_csv(), csv.reader(), csv.writer()
  - ✓ JSON format: json.loads(), json.dumps(), pd.io.json_normalize()

```
"Month", "1958", "1959", "1960"
"JAN",  340,  360,  417
"FEB",  318,  342,  391
"MAR",  362,  406,  419
"APR",  348,  396,  461
"MAY",  363,  420,  472
"JUN",  435,  472,  535
"JUL",  491,  548,  622
"AUG",  505,  559,  606
"SEP",  404,  463,  508
"OCT",  359,  407,  461
"NOV",  310,  362,  390
"DEC",  337,  405,  432
```

```
>>> data = [{'state': 'Florida',
...          'shortname': 'FL',
...          'info': {'governor': 'Rick Scott'},
...          'counties': [{'name': 'Dade', 'population': 12345},
...                       {'name': 'Broward', 'population': 40000},
...                       {'name': 'Palm Beach', 'population': 60000}]},
...         {'state': 'Ohio',
...          'shortname': 'OH',
...          'info': {'governor': 'John Kasich'},
...          'counties': [{'name': 'Summit', 'population': 1234},
...                       {'name': 'Cuyahoga', 'population': 1337}]}]
>>> result = pd.json_normalize(data, 'counties', ['state', 'shortname',
...                                               ['info', 'governor']])
>>> result
         name  population    state shortname info.governor
0        Dade       12345   Florida    FL    Rick Scott
1     Broward       40000   Florida    FL    Rick Scott
2  Palm Beach       60000   Florida    FL    Rick Scott
3      Summit        1234   Ohio       OH    John Kasich
4    Cuyahoga        1337   Ohio       OH    John Kasich
```

# Data Collection and Acquisition

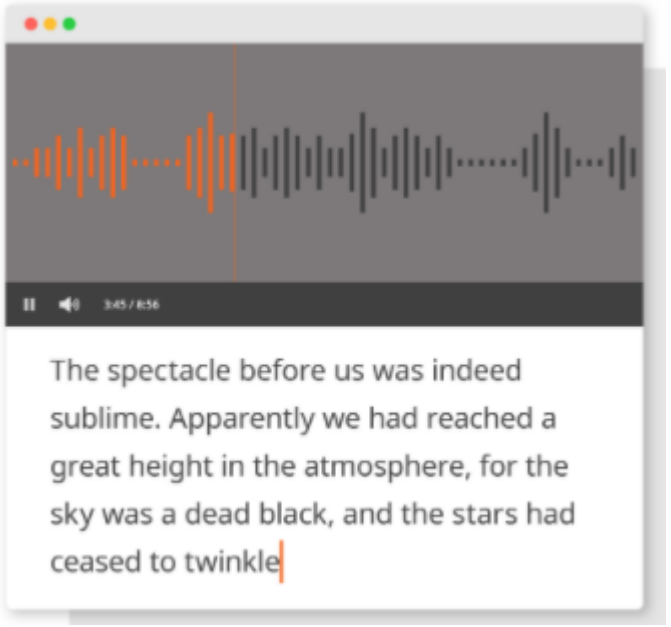
- **Acquisition from Web and SNS (scraping)**
  - ✓ Web scraping (html format): BeautifulSoup(), lxml.html()
  - ✓ SNS scraping (text processing):



# Data Collection and Acquisition

- **Collecting complex and unstructured data**
  - ✓ Text: CountVectorizer(), WordVec(), Embedding layer
  - ✓ audio (and speech), video
  - ✓ Natural language

| | type | text |
|---|---|---|
| 0 | ham | Go until jurong point, crazy.. Available only ... |
| 1 | ham | Ok lar... Joking wif u oni... |
| 2 | spam | Free entry in 2 a wkly comp to win FA Cup fina... |
| 3 | ham | U dun say so early hor... U c already then say... |
| 4 | ham | Nah I don't think he goes to usf, he lives aro... |
| 5 | spam | FreeMsg Hey there darling it's been 3 week's n... |
| 6 | ham | Even my brother is not like to speak with me. ... |
| 7 | ham | As per your request 'Melle Melle (Oru Minnamin... |
| 8 | spam | WINNER!! As a valued network customer you have... |
| 9 | spam | Had your mobile 11 months or more? U R entitle... |



# Data Preprocessing

- **Transform raw data into an understandable format.**
- **Data in real world is:**
  - Incomplete: missing data, lacking attribute values, lacking certain attributes of interest, or containing only aggregate (총액) data
  - Noisy: containing errors (or measurement errors) and outliers
  - Inconsistent: containing discrepancies in codes or names
  - Distorted: sampling distortion
- **"No quality data, no quality mining (and data science) result!"**
  - Garbage In, Garbage Out.
  - Quality decisions must be based on quality data
  - Data warehouse needs consistent integration of quality data

# Data Preprocessing

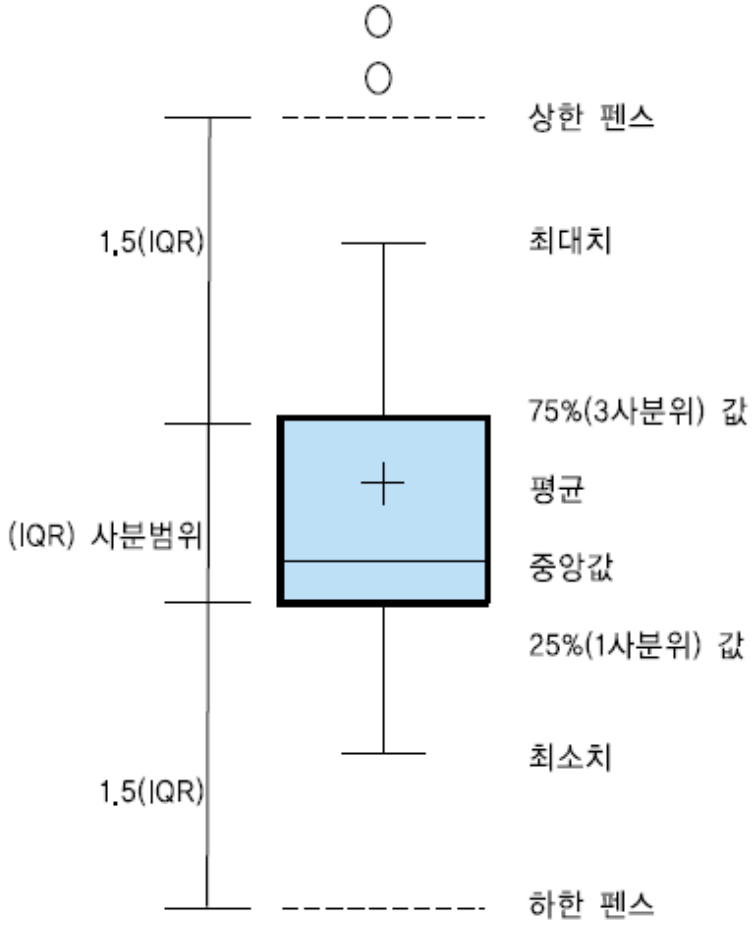
- **Feature Engineering**
  - Redefine / integrate / create / reduce features
  - (ex) built_year –> how_old, cumulative -> by_terms, total -> per_section, etc.
- **Data Cleaning**
  - ✓ Distinguishing errors (during data collection, unrecoverable) from artifacts (during data processing, recoverable)
  - ✓ Data compatibility
  - ✓ Dropping or Imputation of Missing values (결손값의 대체)
  - ✓ Estimating unobserved (zero) counts
  - ✓ Detection and processing of Outliers and Invalid values
- **Data Transformation**
  - ✓ Normalization and Scaling, Categorical Encoding
- **Data Reduction**
  - ✓ Obtains reduced representation in volume but produces the same or similar analytical results

# Data Preprocessing

- **Missing value handling**
  - Drop the rows or the entire column (if data size is large enough)
  - Numerical imputation (replace missing values with 0, median, mean value, max/min, etc.)
  - Categorical imputation (replacing missing values with the maximum occurred value in a column, or imputing a category like "Other")
  - dataframe.dropna(), dataframe.fillna(data.mean())
- **Invalid value handling**
  - Dropping or replacing
- **Handling Outliers**
  - Visualizing is the best way to detect outliers.
  - Statistical methodologies can also be used. (x * standard deviation, z-score, etc.)
  - Dropping or Capping (to upper and lower limit)

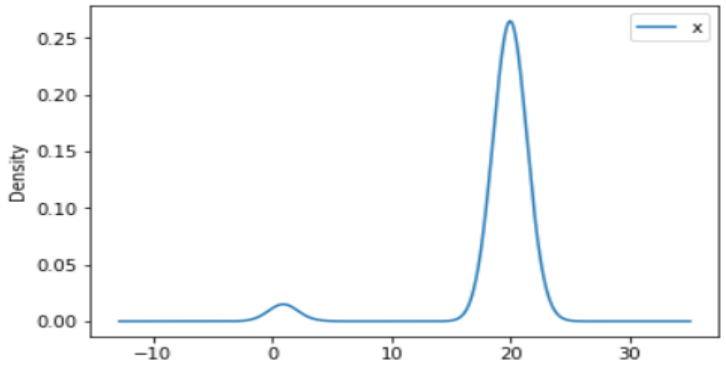
# Data Preprocessing - Scaling

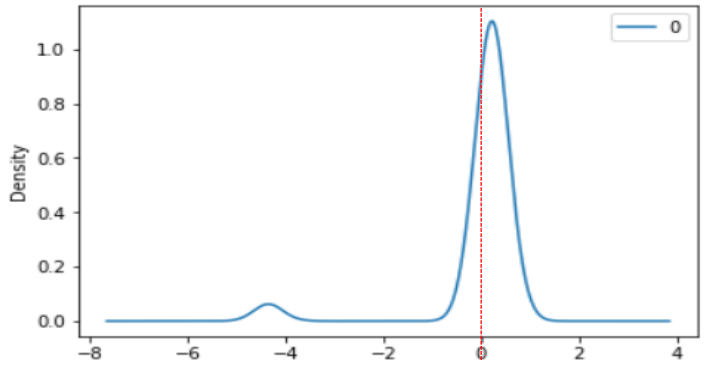
- **Why scaling?**
  - What if two columns have different ranges for the values? Hard to compare.
- **Normalization (min-max scaling):**
  - $x = (x - x_{min}) / (x_{max} - x_{min})$
- **Standardization (standard scaling):**
  - $z = (x - \mu) / \sigma$
- **Robust scaling:**
  - $z = (x - median) / IQR$
  - use median and IQR (instead of μ and σ), robust to outliers
- **Which one to use?**
  - depends on your data

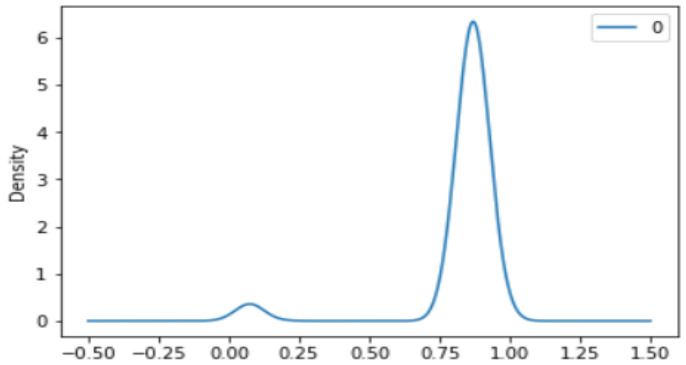


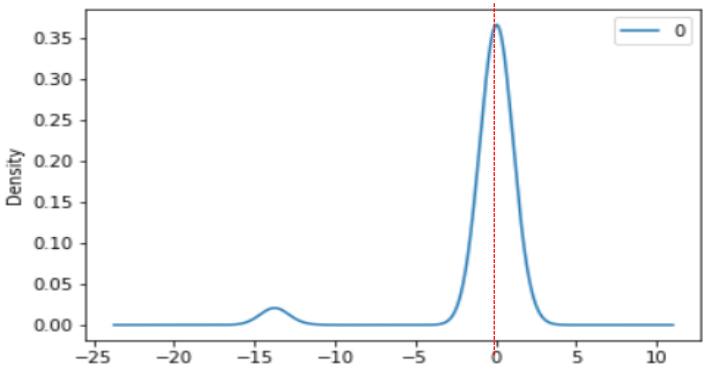
# Data Preprocessing - Scaling

- **Scaling example**
  - ✓ Standard, Min-max, Robust scaling
  - ✓ `pd.DataFrame({'x':np.concatenate([np.random.normal(20, 1, 1000), np.random.normal(1, 1, 50)])`



# Data Preprocessing – Categorical Encoding

- **Categorical variables**
  - ✓ **Nominal Variable** (**Categorical**): Variable comprises a finite set of discrete values with no relationship between values.
  - ✓ **Ordinal Variable:** Variable comprises a finite set of discrete values with a ranked ordering between values.
- **Nominal Encoding: —** Where Order of data does not matter
  - ✓ One Hot Encoding
  - ✓ Binary encoding
  - ✓ One Hot Encoding With Many Categories
  - ✓ Mean Encoding
- **Ordinal Encoding: —** Where Order of data matters
  - ✓ Label Encoding
  - ✓ Target Guided Ordinal Encoding

# Data Preprocessing – Categorical Encoding

- **Label encoding**:
  - ✓ convert the labels into numeric form (0 ~ n_classes-1)
  - ✓ Do not care whether the variables is ordinal or not
- **Ordinal encoding:**
  - ✓ Integer encoding (like Label Encoding) for features, but can specify an order
  - ✓ When there is a known relationship between the classes
  - ✓ (ex) cancer stages, earthquake magnitudes, size of shoes or clothes
- **One-hot encoding**:
  - ✓ spread the feature values to multiple flag columns and assigns 0 or 1 to them.
  - ✓ When there is no known relationship between the classes
  - ✓ (ex) colors (R, G, B), countries, species
- For more, see
  - ✓ https://towardsdatascience.com/all-about-categorical-variable-encoding-305f3361fd02

# Data Preprocessing – Categorical Encoding

| State (Nominal Scale) |
|---|
| Maharashtra |
| Tamil Nadu |
| Delhi |
| Karnataka |
| Gujarat |
| Uttar Pradesh |

| State (Label Encoding) |
|---|
| 3 |
| 4 |
| 0 |
| 2 |
| 1 |
| 5 |

label encoding

| Original Encoding | Ordinal Encoding |
|---|---|
| Poor | 1 |
| Good | 2 |
| Very Good | 3 |
| Excellent | 4 |

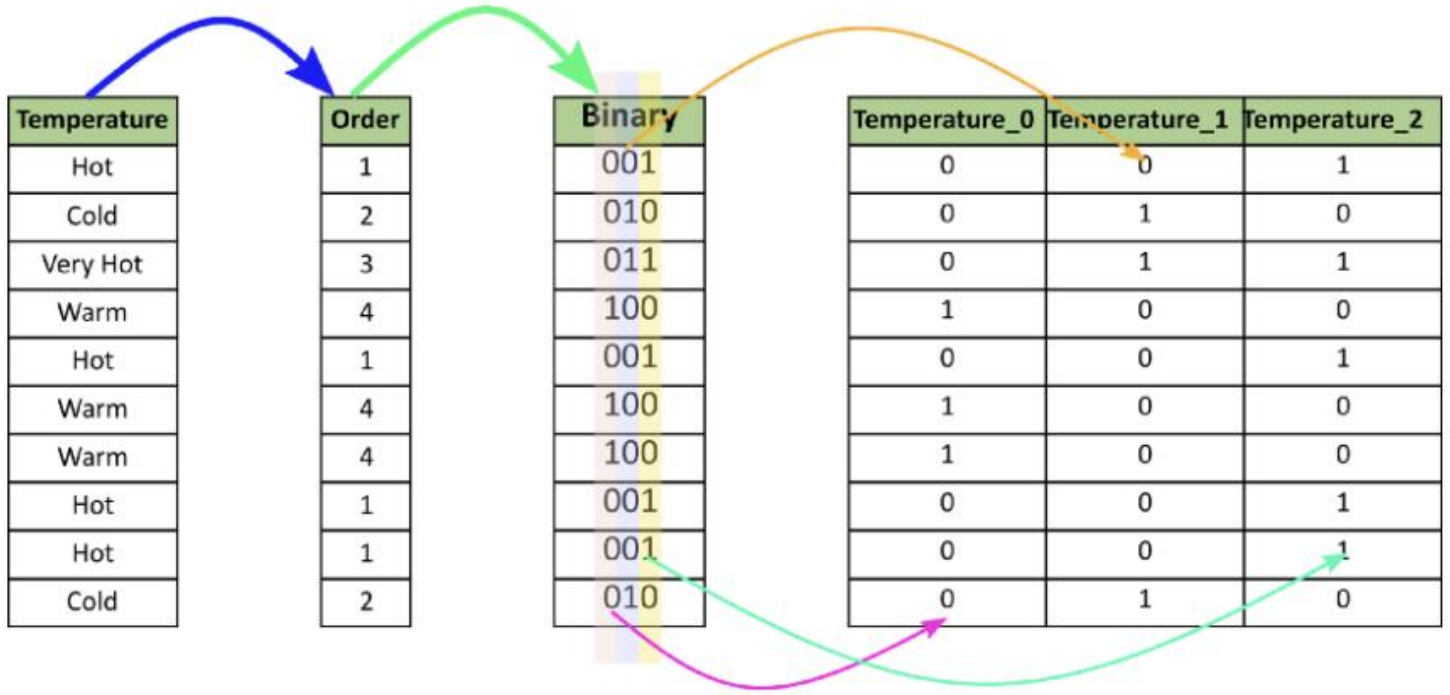
ordinal encoding

| State | State_Maharashtra | State_Tamil Nadu | State_Delhi | State_Karnataka | State_Gujarat | State_Uttar Pradesh |
|---|---|---|---|---|---|---|
| Maharashtra | 1 | 0 | 0 | 0 | 0 | 0 |
| Tamil Nadu | 0 | 1 | 0 | 0 | 0 | 0 |
| Delhi | 0 | 0 | 1 | 0 | 0 | 0 |
| Karnataka | 0 | 0 | 0 | 1 | 0 | 0 |
| Gujarat | 0 | 0 | 0 | 0 | 1 | 0 |
| Uttar Pradesh | 0 | 0 | 0 | 0 | 0 | 1 |

One-hot encoding

# Data Preprocessing – Categorical Encoding

- **Binary Encoding**
  - converts a category into binary digits (each binary digit creates one feature column)
  - Compared to One Hot Encoding, fewer feature columns
  - (ex) for 100 categories, One Hot Encoding will have 100 features while Binary encoding will need just seven features.
  - BinaryEncoder() function in category_encoders package



| Temperature |
|---|
| Hot |
| Cold |
| Very Hot |
| Warm |
| Hot |
| Warm |
| Warm |
| Hot |
| Hot |
| Cold |

| Order |
|---|
| 1 |
| 2 |
| 3 |
| 4 |
| 1 |
| 4 |
| 4 |
| 1 |
| 1 |
| 2 |

| Binary |
|---|
| 001 |
| 010 |
| 011 |
| 100 |
| 001 |
| 100 |
| 100 |
| 001 |
| 001 |
| 010 |

| Temperature_0 | Temperature_1 | Temperature_2 |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |

# Data Preprocessing – Categorical Encoding

- **Frequency Encoding**
  - utilize the frequency of the categories as labels
  - Useful when the frequency is related somewhat with the target variable

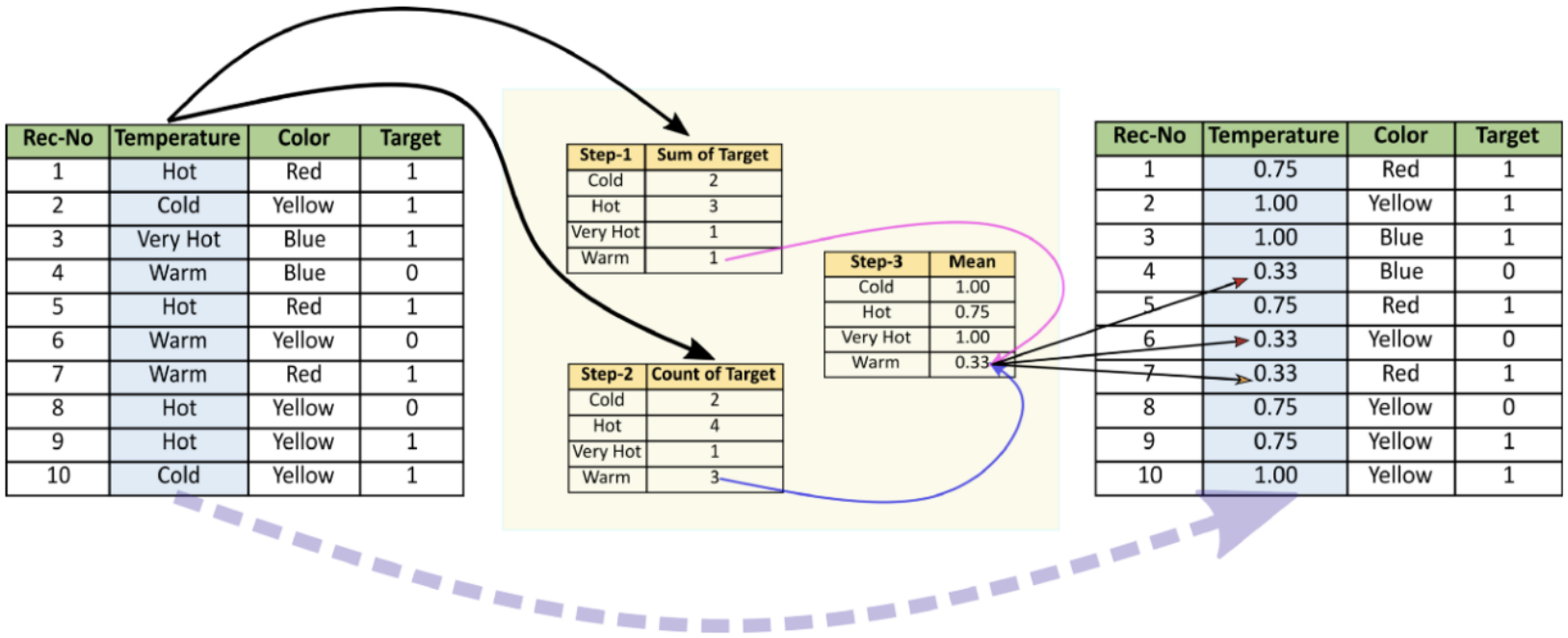
- Encoding of categorical levels of feature to values between 0 and 1 based on their relative frequency

| | |
|---|---|
| A | 0.44 (4 out of 9) |
| B | 0.33 (3 out of 9) |
| C | 0.22 (2 out of 9) |

| Feature | Encoded Feature |
|---|---|
| A | 0.44 |
| A | 0.44 |
| A | 0.44 |
| A | 0.44 |
| B | 0.33 |
| B | 0.33 |
| B | 0.33 |
| C | 0.22 |
| C | 0.22 |

# Data Preprocessing – Categorical Encoding

- **Mean Encoding (or Target Encoding)**
  - very popular encoding approach followed by Kagglers
  - similar to label encoding, except here labels are correlated directly with the target (each category is encoded as the mean value of the target variable on a training data)



| Rec-No | Temperature | Color | Target |
|---|---|---|---|
| 1 | Hot | Red | 1 |
| 2 | Cold | Yellow | 1 |
| 3 | Very Hot | Blue | 1 |
| 4 | Warm | Blue | 0 |
| 5 | Hot | Red | 1 |
| 6 | Warm | Yellow | 0 |
| 7 | Warm | Red | 1 |
| 8 | Hot | Yellow | 0 |
| 9 | Hot | Yellow | 1 |
| 10 | Cold | Yellow | 1 |

| Step-1 | Sum of Target |
|---|---|
| Cold | 2 |
| Hot | 3 |
| Very Hot | 1 |
| Warm | 1 |

| Step-2 | Count of Target |
|---|---|
| Cold | 2 |
| Hot | 4 |
| Very Hot | 1 |
| Warm | 3 |

| Step-3 | Mean |
|---|---|
| Cold | 1.00 |
| Hot | 0.75 |
| Very Hot | 1.00 |
| Warm | 0.33 |

| Rec-No | Temperature | Color | Target |
|---|---|---|---|
| 1 | 0.75 | Red | 1 |
| 2 | 1.00 | Yellow | 1 |
| 3 | 1.00 | Blue | 1 |
| 4 | 0.33 | Blue | 0 |
| 5 | 0.75 | Red | 1 |
| 6 | 0.33 | Yellow | 0 |
| 7 | 0.33 | Red | 1 |
| 8 | 0.75 | Yellow | 0 |
| 9 | 0.75 | Yellow | 1 |
| 10 | 1.00 | Yellow | 1 |