1 Data Exploration

Informative analyses to formulate and refine hypotheses for

- feature subset selection
- selection of tools and techniques
- further data collection

1.1 Distribution of a single variable

In statistics, a quantity that we measure or classify.

1.1.1 Populations and samples

We study properties of one or more samples to characterize a wider, much larger, population.

1.1.2 Data sets

Collection of values of raw variables.

1.1.3 Variables and observations

An observation is a list of values of all variables for a single member of a population.

1.1.4 Types of data

Numeric variables are measurable, quantitative, may be continuous (fractional) or discrete (integer).

Categorical variables are classified, qualitative, may be ordinal (ranked) or nominal (without ordering).

Scale can be interval (time) or ratio (length, weight).

Cross-sectional data is measured at certain point of time while time series data is measured at various points in time.

Longitudinal data combines cross-sectional with time series.

1.1.5 Descriptive measures for numerical variables

Probability density function: likelihood that a continuous random variable is equal to some value

Central tendency: mean (average), median (middle observation in sorted data), mode (most common value)

minimum and maximum

percentiles: value below which given percentage of data falls

Dispersion: range, variance (average squared deviation from mean), mean absolute deviation (MAD, average absolute deviation)

skew: negatively or positively

kurtosis: flatness of tails

1.1.6 Descriptive measures for categorical variables

Probability mass function: probability that a discrete random variable is exactly equal to some value

Frequency distribution

Central tendency: mode (most common value), median (middle observation in sorted ordinal values)

Graphical summaries: bar chart

1.1.7 Outliers and missing values

Unusual values that lie outside the norm.

Should be either filtered out or treated separately.

Generally a rare case and contributes to noise in the data.

1.2 Finding relationships among variables

- Descriptive methods (describe attributes of a data set)
- Inferential methods (draw conclusions about a population from samples)

An association means that knowing the value of one variable gives us some information about the possible values of the second variable.

Associations between pairs of variables are called bivariate associations.

1.2.1 Numerical variables

Pearson's product-moment correlation coefficient

Data: (1,2), (0,4), (3,2), (1,6), (5,1)

- Pearson's correlation coefficient takes a value of 0 if two variables are uncorrelated, and a value of +1 or -1 if we can predict the exact value of one variable given knowledge of the other using linear equation.
- A positive value indicates that higher values in one variable is associated with higher values of the second. A negative value indicates that higher values of one variable is associated with lower values of the second.
- Pearson's correlation coefficient will be misleading when relationship is curved or scattered and not linear.

Kendall's τ

Measures of rank correlation are based on a comparison of the resulting ranks.

Any pair of observations i and j , where i < j, are said to be concordant if the sort order of i and j agrees.

$$\frac{concordant\ pairs - disconcordant\ pairs}{C(n,2)}$$

Data: (1,2), (0,4), (3,2), (1,6), (5,1)

1.2.2 Categorical variables

Numerically exploring associations between pairs of categorical variables is not as simple as the numeric variable case. We need to find out which combinations are common and which are rare.

Contingency table

Display frequencies in the form of a table

```
##                         month
## type                    Jun Jul Aug Sep Oct Nov Dec
##   Extratropical         27  38  23 149 129  42   4
##   Hurricane              3  31 300 383 152  25   2
##   Tropical Depression   22  59 150 156  84  42   0
##   Tropical Storm        31 123 247 259 204  61   1
```

Which months are most affected?

Which storms are most common?

Both variables are nominal so Pearson's coefficient is not possible.

Spearman’s ρ rank correlation coefficient

It is valid for both numeric and categorical data.

$$\rho = 1 - \frac{6\sum(d_i^2)}{n(n^2-1)}$$

Data: (1,2), (0,4), (3,2), (1,6), (5,1)

Sample proportion

The sample proportion is used to estimate the value of a population proportion (the corresponding proportion of times something happens in the whole population).

1.2.3 Categorical variables and numerical variables

Such associations are studied visually using box plots and histogram that will be studied later.

Magic: Simpon’s Paradox

	Men		Women	
	Selected	**Applied**	**Selected**	**Applied**
Computer Sc.	70	100	40	50
Electronics	10	50	30	100

1.3 Sampling and distributions

1.3.1 Terminology

To obtain information about a large population by examining only a small fraction of that population.

This saves cost and time.

1.3.2 Estimation

See examples below.

1.3.3 Sampling distributions

Simple Random Sampling (SRS)

- A fixed number of samples is selected from the population without replacement. Without replacement means no individual member will appear more than once in a sample.

  How many simple random samples of size 2 can be generated from the population: 1, 2, 1, 3, 2 ?

- Expected value of sample mean, variance of sample mean and sample variance.

  A coin is tossed 10 times and heads appears thrice. What is the standard error in the estimate of the probability of 0.3?

A coin is tossed 50 times and heads appears 15 times. What is the standard error in the estimate of the probability of 0.3?

Stratified Sampling

Method of sampling from a population which can be partitioned into subpopulations.

Proportionate allocation uses a sampling fraction in each of the strata that are proportional to that of the total population.

Optimum allocation uses a sampling fraction of each stratum that is proportionate to both the proportion and the standard deviation of the stratum.

Cluster Sampling

Total population is divided into groups (clusters) and a simple random sample of the groups is selected. The elements in each cluster are then sampled.

If all elements in each sampled cluster are sampled, then this is referred to as a "one-stage" cluster sampling.

If a simple random subsample of elements is selected within each of these groups, this is referred to as a "two-stage" cluster sampling.

1.3.4 Confidence interval

Standardized score is measure of relative standing.

Normal approximation to the sampling distribution of $\bar{X}$

1.3.5 Confidence Interval estimation

A coin is tossed 10 times and heads appears 8 times. What are the chances of error this large if the coin is known to be unbiased? Perform both one-sided and two-sided analysis.

An unbiased coin is tossed 100 times. What is its 95% confidence interval?

How many times a biased coin should be tossed so that the fraction of heads falls between 0.35 to 0.45 with 90% confidence?

Inverse Normal Table

0.50	0.0000	**0.30**	0.5244	**0.10**	1.2816
0.49	0.0251	**0.29**	0.5534	**0.09**	1.3408
0.48	0.0502	**0.28**	0.5828	**0.08**	1.4051
0.47	0.0753	**0.27**	0.6128	**0.07**	1.4758
0.46	0.1004	**0.26**	0.6433	**0.06**	1.5548
0.45	0.1257	**0.25**	0.6745	**0.05**	1.6449
0.44	0.1510	**0.24**	0.7063	**0.04**	1.7507
0.43	0.1764	**0.23**	0.7388	**0.03**	1.8808
0.42	0.2019	**0.22**	0.7722	**0.025**	1.9600
0.41	0.2275	**0.21**	0.8064	**0.02**	2.0537
0.40	0.2533	**0.20**	0.8416	**0.01**	2.3263
0.39	0.2793	**0.19**	0.8779	**0.009**	2.3656
0.38	0.3055	**0.18**	0.9154	**0.008**	2.4089
0.37	0.3319	**0.17**	0.9542	**0.007**	2.4573
0.36	0.3585	**0.16**	0.9945	**0.006**	2.5121
0.35	0.3853	**0.15**	1.0364	**0.005**	2.5758
0.34	0.4125	**0.14**	1.0803	**0.004**	2.6521
0.33	0.4399	**0.13**	1.1264	**0.003**	2.7478
0.32	0.4677	**0.12**	1.1750	**0.002**	2.8782
0.31	0.4959	**0.11**	1.2265	**0.001**	3.0902

Inverse Normal Table

0.50	0.0000	**0.30**	0.5244	**0.10**	1.2816
0.49	0.0251	**0.29**	0.5534	**0.09**	1.3408
0.48	0.0502	**0.28**	0.5828	**0.08**	1.4051
0.47	0.0753	**0.27**	0.6128	**0.07**	1.4758
0.46	0.1004	**0.26**	0.6433	**0.06**	1.5548
0.45	0.1257	**0.25**	0.6745	**0.05**	1.6449
0.44	0.1510	**0.24**	0.7063	**0.04**	1.7507
0.43	0.1764	**0.23**	0.7388	**0.03**	1.8808
0.42	0.2019	**0.22**	0.7722	**0.025**	1.9600
0.41	0.2275	**0.21**	0.8064	**0.02**	2.0537
0.40	0.2533	**0.20**	0.8416	**0.01**	2.3263
0.39	0.2793	**0.19**	0.8779	**0.009**	2.3656
0.38	0.3055	**0.18**	0.9154	**0.008**	2.4089
0.37	0.3319	**0.17**	0.9542	**0.007**	2.4573
0.36	0.3585	**0.16**	0.9945	**0.006**	2.5121
0.35	0.3853	**0.15**	1.0364	**0.005**	2.5758
0.34	0.4125	**0.14**	1.0803	**0.004**	2.6521
0.33	0.4399	**0.13**	1.1264	**0.003**	2.7478
0.32	0.4677	**0.12**	1.1750	**0.002**	2.8782
0.31	0.4959	**0.11**	1.2265	**0.001**	3.0902

1.3.6 Hypothesis testing

Neyman-Pearson approach

Null hypothesis: observed deviation from assumed distribution is by chance

Alternative hypothesis: observed deviation is not by chance

Type-I error happens if valid H0 is rejected and its maximum allowed probability is called *significance level* of the test denoted by α.

Type-II error happens if invalid H0 is accepted and probability of rejecting an invalid H0 is called power of the test.

Probability of obtaining observed results under the assumption that H0 is correct is called *p-value*.

What is the p-value of coin being not biased if 8 heads are observed in 10 tosses? (Use Normal or binomial distribution)

What are the chances that the coin is not biased if 8 heads are observed in 10 tosses when it is known that if biased, the probability of heads is 0.7? (Use binomial distribution)

Summary of Hypothesis Testing

Null hypothesis is rejected if p-value is less than significance level. Why?

Confidence interval is determined by z-score corresponding to the significance level.

How is p-value related to probability of null hypothesis to be true?

1.3.7 Chi-squared test for independence

Chi-square distribution

Let a coin tossed n times on an average gives n*p heads where p is the probability of getting a head.

Compute the population variance.

What is the variation in the estimation of p itself?

Partial derivation of Pearson's Chi-square test

$$\hat{p} \sim N\left(\hat{p}; p, \frac{p(1-p)}{n}\right)$$

Then $p' = \frac{n\hat{p} - np}{\sqrt{np}} \sim N(p'; 0, 1-p)$

$$\Rightarrow q = p'^2 = \frac{(n\hat{p} - np)^2}{np} \sim Z(q; 0, 1-p)$$

Here p' is normally distributed with variance $1-p$ and let the distribution of q be Z. We note that $n\hat{p}$ is observed frequency (O) and np is the expected frequency (E). Therefore,

$$q = \frac{(O-E)^2}{E} \sim Z(q; 0, 1-p)$$

We can divide the domain of any arbitrary distribution into b different bins. Each bin behaves like tossing a coin because a randomly generated variable from that distribution will either fall in a bin with probability p or not fall in that bin. Adding Z distributions for all the b bins we get **Chi-square distribution** of $b-1$ degrees of freedom (proof available in literature).

$$\sum_{j=1}^{b} \frac{(O_j - E_j)^2}{E_j} \sim \chi^2_{b-1}$$

Note that Z is not χ^2_1, and bin i and bin j are not independent, otherwise the sum would have been χ^2_b with b degrees of freedom.

Critical values of the Chi-square distribution with *d* degrees of freedom

	Probability of exceeding the critical value						
d	0.05	0.01	0.001	*d*	0.05	0.01	0.001
1	3.841	6.635	10.828	11	19.675	24.725	31.264
2	5.991	9.210	13.816	12	21.026	26.217	32.910
3	7.815	11.345	16.266	13	22.362	27.688	34.528
4	9.488	13.277	18.467	14	23.685	29.141	36.123
5	11.070	15.086	20.515	15	24.996	30.578	37.697
6	12.592	16.812	22.458	16	26.296	32.000	39.252
7	14.067	18.475	24.322	17	27.587	33.409	40.790
8	15.507	20.090	26.125	18	28.869	34.805	42.312
9	16.919	21.666	27.877	19	30.144	36.191	43.820
10	18.307	23.209	29.588	20	31.410	37.566	45.315

***INTRODUCTION TO POPULATION GENETICS*, Table D.1**

Example 1

Does the assumption of Poisson distribution seem appropriate as a model for following number of defects observed:

```
no defects:    35
1 defect:      11
2 defects:     11
3 defects:      3
```

Example 2

	Blue	Green	Pink	
Boys	100	150	~~20~~ 50	300
Girls	20	30	~~180~~ 150	200
	120	180	200	N = 500

Use $\alpha = 0.05$

H_0; *For the population of elementary school students, gender and favorite color are not related.*

H_1; *For the population of elementary school students, gender and favorite color are related.*

1.4 Regression

1.4.1 Linear Regression

Structure of input data

Size ($feet^2$) x_1	Number of bedrooms x_2	Number of floors x_3	age of home (years) x_4	Price($1000) $h_\theta(x) = y$
2104	5	1	45	460
1416	3	2	40	232
1534	2	2	30	315
…	…	…	…	…

A data

Features (x)

Label ($h_\theta(x) = y$)

Structure of linear regression model

$$h_\theta(x) = \theta_0 + 2104\theta_1 + 5\theta_2 + \theta_3 + 45\theta_4 = 460$$
$$h_\theta(x) = \theta_0 + 1416\theta_1 + 3\theta_2 + 2\theta_3 + 40\theta_4 = 232$$
$$h_\theta(x) = \theta_0 + 1534\theta_1 + 2\theta_2 + 2\theta_3 + 30\theta_4 = 315$$

Definition: In linear regression model, a variable, called dependent variable, is assumed to be normally distributed around linear combination of other variables, called independent variables.

$$p(y|x_1, x_2, ...) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-\|y - \boldsymbol{x}\theta\|^2}{2\sigma^2}}$$

Here y is the dependent variable and $\boldsymbol{x} = x_1, x_2, \ldots$ are independent variables. *(Multiple Linear Regression)*

We need to find θ such that this probability is maximized.

This is equivalent to minimizing $\|y - \boldsymbol{x}\,\theta\|^2$.

For multiple data points, the quantity to be minimized is $\|\boldsymbol{y} - X\,\theta\|^2$

Taking derivative with respect to θ and equating to 0 gives the solution, $\theta = (X^T X)^{-1} X^T \boldsymbol{y}$

Example 1

X: [1,0], [0,-1], [1,-2], [2,0], [0,-2]

y: 0, -1, -4, -1, -3

Example 2 (Homework)

X: 1, 0, 1, 2, 0

Y: 1.0, 2.0, 0.9, 0.0, 2.2

1.4.2 Logistic Regression

[As discussed on board.]

1.4.3 Non Linear Regression

Use linear regression after adding additional attributes derived by applying non-linear functions on original attributes. For example, x^2 and x^3 can be incorporated to use linear regression to fit cubic equation.

2 Introduction to Data Science

2.1 Data Science

- Data science is a field that uses statistics, scientific computing, scientific methods, processes, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data.
- It unifies domain knowledge with statistics and data analytics.

2.1.1 Challenges of Traditional Systems

1. Data set size is assumed to be limited and fixed. Huge data sets are slow to handle in traditional systems.
2. Designed to handle structured data only (containing numbers and categories). They cannot directly work on unstructured data like text, video and audio.
3. They are efficient in carrying out relatively easy-to-perform queries and analysis, but not otherwise.
4. They use conventional methods (like basic statistics) for analysis and need to support new methodologies (like machine learning).

2.1.2 Evolution of Analytic Scalability

1. Until well into the 1900s, doing analytics was very, very difficult. To do a deep analysis, such as a predictive model, it required manually computing all of the statistics. Scalability of any sort was virtually nonexistent.
2. Scalability is the ability of a system to handle increasing amount of work required to perform its task.
3. Increase in data storage ability has grown in recent years to accommodate the need for big data. The amount of data has grown at least as fast as the computing power of the machines that process it.
4. As new big data sources become available the boundaries are being pushed further.
5. Generally the environments for data storage and data analytics are different and data is moved from storage to analytical environment. In the analytical environment advanced processing is carried out for data mining and predictive modeling. This has been disrupted by introduction of MPP (Massively Parallel Processing) systems.

Massively Parallel Processing System

1. It is a system containing lots of processors that workin parallel by using divide and conquer strategy.
2. An MPP database breaks the data into independent pieces managed by independent storage and CPU resources.
3. It specializes in handling queries on datasets of sizes up to a petabyte and more by parallelizing SQL queries across all resources in the cluster.
4. MPP systems have built-in redundancy to make recovery easy, for example by providing facility to manage CPU and disk space and by query optimization.
5. Examples of areas where query can be optimized: join, aggregation, derivation, transformation.
6. The reason MPP can be a huge benefit to advanced analytics is because most of the processing pain in advanced analytics comes during the data preparation stages. This is the process of combining various data sources to pull together all the information needed for an analysis.

2.1.3 Types of Computing (Distributed, Parallel, Grid)

Parallel computing

In parallel computing multiple processors perform multiple tasks assigned to them simultaneously. Memory in parallel systems may be shared or distributed.

Distributed computing

It consists of multiple autonomous computers which appear like a single system. So there is no shared memory and computes communicate (message passing) with each other using network.

Grid computing

The computation is divided into multiple tasks and each task is processed on different machine in parallel. Each machine may be able to handle only the fraction of the work and potentially handle only one job at a time therefore grid computing is suitable for large number of small to medium size tasks.

A grid configuration can help in optimizing both cost and performance. Computation relies on several low cost machines instead of single high-end server. The environment can be scaled relatively cheaply and quickly.

Differences between Parallel Computing and Distributed Computing

1. In parallel computing many operations are performed simultaneously. In distributed computing system components are located at different locations.

2. In parallel computing single computer is used while distributed computing uses multiple computers.

3. In parallel computing multiple processors perform multiple operations and in distributed computing multiple computers perform multiple operations.

4. Parallel computing may use shared or distributed memory but distributed computing has to use only distributed memory.

5. Processors communicate with each other through bus in parallel computing. Computers communicate with each other through message passing in distributed computing.

6. Parallel computing improves the system performance but distributed computing improves system scalability, fault tolerance and resource sharing capabilities.

2.2 Introduction to Data Analytics

It is a broad term that encompasses the processes, technologies, frameworks and algorithms to extract meaningful insights from data. This requires choice of technologies, algorithms, and frameworks to be used.

2.2.1 Importance of Data Analytics

- cost reduction: all operation like conception, design, production and marketing can be optimized and streamlined.
- better and faster decision making: it helps to figure out strategies to boost performance and solve problems. For example, improving efficiency of marketing campaign.
- designing new products and services: the capability of data analytics for exploration and estimation of information is extremely useful. It helps in accurate study of market requirements.

2.2.2 Types of Data Analytics

1. **Descriptive analytics**
   - It is used to summarize data for easy interpretation.
   - For example, basic statistics to find median or variance or range.
2. **Diagnostic analytics**
   - It is used to find reasons from data for why an event happened.
   - For example, linear algebra can be used to model patterns of previous faults.
3. **Prescriptive analytics**
   - It is used to predict from data the likely outcome on an event. Best course of action for the predicted outcome is also suggested.
   - For example product recommendation using graph theory.

2.2.3 Data Analytics Lifecycle

The lifecycle consists of several phases

1. **Discovery**
   - Learn about the problem domain, conduct interviews, study sources of data.
   - Study the past problems that were solved.
   - Formulate initial hypothesis, ideas that can be tested.
2. **Data preparation**
   - Execute ELT (extract, load, transform) to get data into the data warehouse.
   - Conditioning through cleaning and normalization.
   - Conversion from categorical to numerical or one-hot vector.
3. **Model planning**
   - Selection of methods, techniques and processing piplines.
   - Ensuring proper structure of datasets.
   - Exploring data to select important variables.
   - Selecting models appropriate for the nature of data available (structured or unstructured).
4. **Model building**

- Develop testing and training datasets.
- Implement the model.
- Ensure availability of hardware requirements.
- Execute the model.

5. **Communication of results**
   - Evaluate business value of the results.
   - List the key findings and summaries.
   - Suggest future improvements.
   - Prepare documentation.
6. **Operationalize**
   - Start using the model in the production environment.
   - Deploy the model in controlled way.
   - Provide concise responses for effective real-time decision making.

2.2.4 Data Analytics Applications

Few examples,

1. Web analytics
   - collection and analysis of data on the user visits on websites and cloud applications to get insigned about user engagement.
   - The key reporting metrics include user sessions, page visits, top entry and exit pages, bounce rate, most visited page, time spent on each page, number of unique visitors, number of repeat visitors
2. Retail and marketing
   - Ads are displayed within websites, videos and mobile applications who participate in the advertising network.
   - Advertises can use big data tools for tracking the performance of advertisements, optimizing the bids for pay-per-click advertising, tracking which keywords link the most to the advertising landing pages and optimizing budget allocation to various advertisement campaigns.
3. Banking and financial applications
   - Investments and stock prices can experience rapid changes in market.

- For maintining and tracking the performance of invenstment portfolios.
- Credit risk modeling and fraud detection are important applications.

4. Healthcare
    - Healthcare ecosystem consists of numerous entities including healthcare providers (primary care physicians, specialists, or hospitals), payers (government, private health insurance companies, employers), pharmaceutical, device and medical service companies, IT solutions and services firms, and patients.
    - Data analytics is useful for valuable aggregated information about overall patient populations to determine best treatments, medicines and real-time monitoring practices.

2.3 Introduction to Big Data

Big data is defined as collections of datasets whose volume, velocity or variety is so large that it is difficult to store, manage, process and analyze the data using traditional databases and data processing tools.

2.3.1 Challenges

1. Meeting the need for speed
   - Storing data efficiently
   - Finding relevant data quickly
2. Visualization helps organizations perform analyses
3. Degree of granularity increases
   - user base of a company may grow explosively
   - companies release apps that produce even more real-time data for each customer
4. Privacy issues
   - data breach
   - ransomware
5. Fault tolerance
   - distributed systems are fault-tolerant
   - highly available architectures using data replication
6. Scalability and performance

7. Batch-processing and real-time processing

2.3.2 Characteristics

1. **Volume** of data involved is so large that it is difficult to store, process and analyze data on a single machine (MB, GB, TB, PB)
2. **Velocity** of data is very high and the data needs to be analyzed in real-time (moving target)

   Categories: batch >> period >> real-time
3. There is **variety** of data involved, which can be structured, unstructured or semi-structured, and is collected from multiple data sources

   Categories: databases >> photo/web/audio >> social/mobile
4. **Veracity** refers to quality of data. Cleaning the data can improve its quality.
5. **Value** contribution towards intended application requires various types of analytics to be performed such as descriptive, diagnostic, predictive and prescriptive analytics.
6. Big Data tools and frameworks have **distributed and parallel processing architectures** and can leverage the storage and computational resources of a large cluster of machines.

2.3.3 Hadoop

Apache Hadoop

It is an open source framework for distributed batch processing (MapReduce) of big data.

Hadoop YARN

- It is the next generation architecture of Hadoop.
- The original processing engine of Hadoop (MapReduce) has been separated from the resource management component (which is now part of YARN).
- YARN is like operating system for Hadoop that supports different processing engines on a Hadoop cluster such as MapReduce for batch processing, Apache Tez for interactive queries, Apache Storm for stream processing.

Resource Manager (RM)

It manages the global assignment of compute resources (resources) to applications. RM consists of two main services:

- Scheduler: It is a pluggable service that manages and enforces the resource scheduling policy in the cluster.
- Applications Manager (AsM): AsM manages the running Application Masters in the cluster. AsM is responsible for starting application masters and for monitoring and restarting them on different nodes in case of failures.

Application Master (AM)

It manages the application’s life cycle. AM is responsible for negotiating resources from the RM and working with the NMs to execute and monitor the tasks.

Node Manager (NM)

A per-machine NM manages the user processes on that machine.

Containers

- Container is a bundle of resources allocated by RM (memory, CPU and network).
- It is a conceptual entity that grants an application the privilege to use a certain amount of resources on a given machine to run a task.
- Each node has an NM that spawns multiple containers based on the resource allocations made by the RM.

Hadoop Schedulers

- FIFO scheduler maintains a work queue in which the jobs are queued. The scheduler pulls jobs in first-in first-out manner (oldest job first) for scheduling. There is no concept of priority
- Fair Scheduler assigns resources to jobs such that each job gets an equal share of the available resources on average over time. Unlike the FIFO scheduler, which

forms a queue of jobs, the Fair Scheduler lets short jobs finish in reasonable time while not starving long jobs.

- In Capacity Scheduler, multiple named queues are defined, each with a configurable number of map and reduce slots. The Capacity Scheduler gives each queue its capacity when it contains jobs, and shares any unused capacity between the queues. Within each queue FIFO policy is used.

2.3.4 MapReduce

- MapReduce is a popular programming model for data intensive applications. It is a parallel processing framework.
- Computational processing can occur on data (even semi-structured and unstructured data) stored in a file system without loading it into any kind of database.
- It’s neither a database nor a direct competitor to databases.
- The run-time systems take care of tasks such partitioning the data, scheduling of jobs and communication between nodes in the cluster. This makes it easier for programmers to analyze massive scale data without worrying about tasks such as data partitioning and scheduling.
- This model has been implemented by the open source software Hadoop.

MapReduce phases

MapReduce model has two phases: Map and Reduce.

- In the Map phase, data is read from a distributed file system, partitioned among a set of computing nodes in the cluster, and sent to the nodes as a set of key-value pairs.
- When all the Map tasks are completed, the Reduce phase begins with the shuffle and sort step, in which the intermediate data is sorted by the key and the key-value pairs are grouped and shuffled to the reduce tasks.
- The reduce tasks then take the key-value pairs grouped by the key and run the reduce function for each group of key-value pairs.

An optional Combine task can be used to perform data aggregation on the intermediate data of the same key for the output of the mapper before transferring the output to the Reduce task.

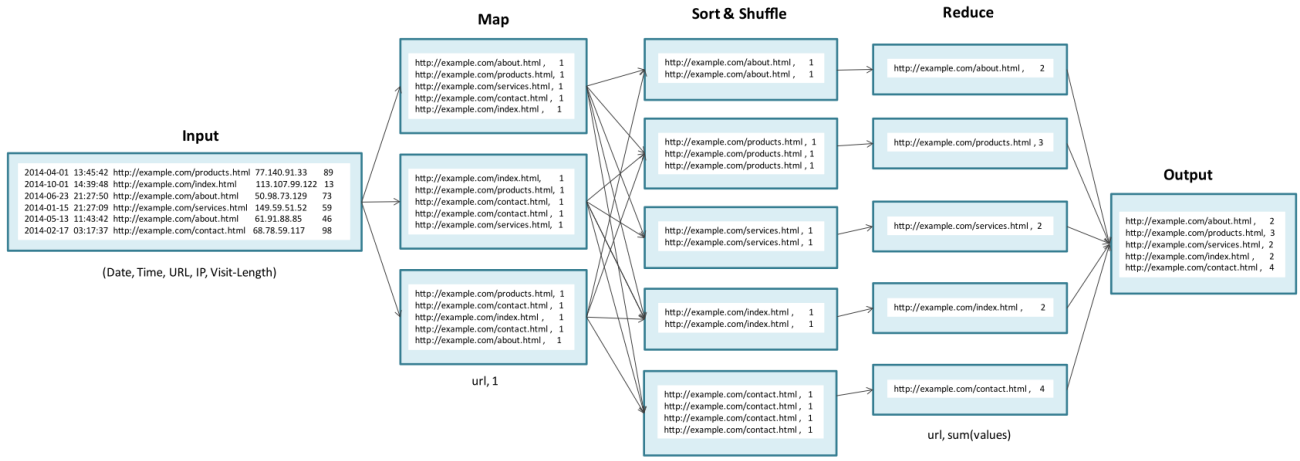
MapReduce Examples

Example 1: Counting

The map step parses the provided text string into individual words and emits a set of key/value pairs of the form

<word, 1 >.

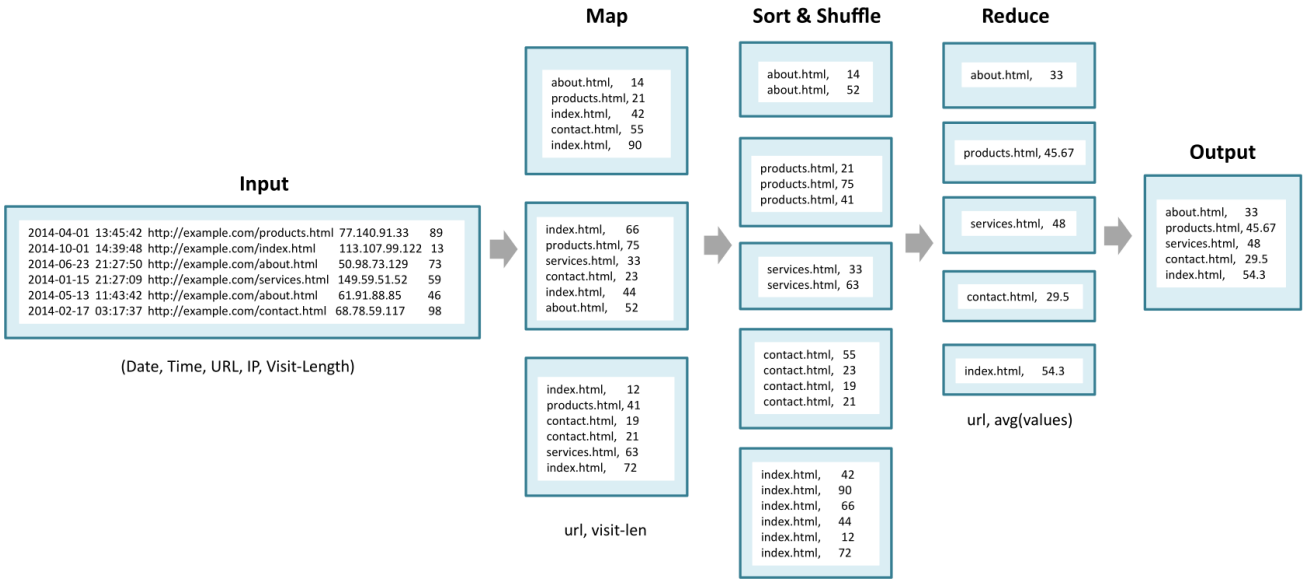
For each unique key-in this example, word-the reduce step sums the 1 values and outputs the <word, count> key/value pairs.



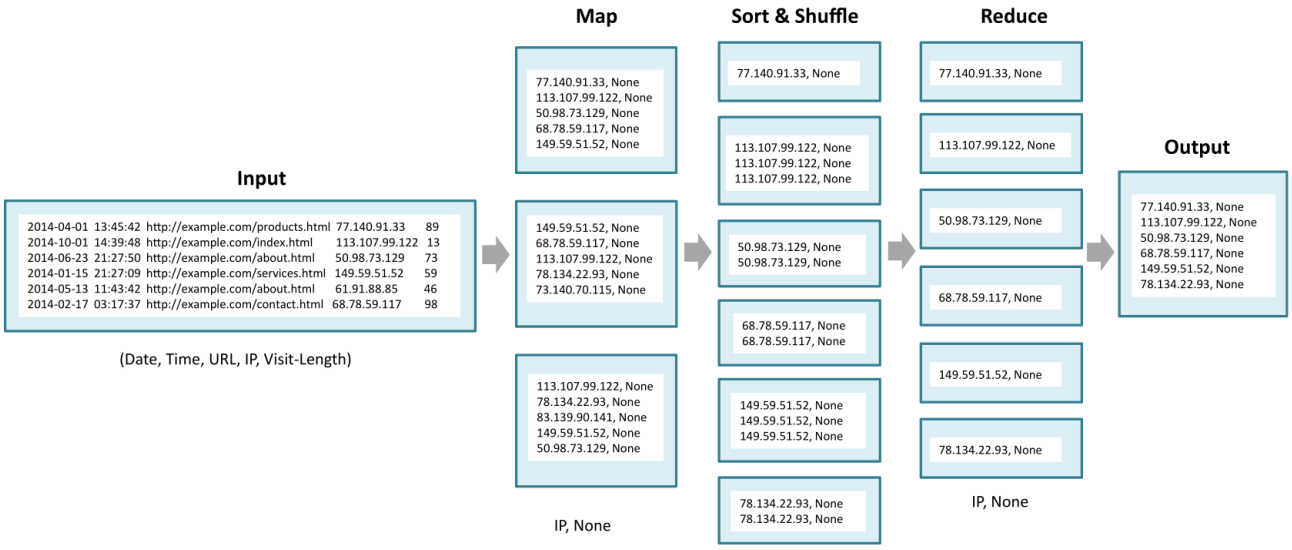
[More examples on next page]

Identify the tasks in these examples:

Example 2



Example 3



Additional Map-Reduce Algorithms

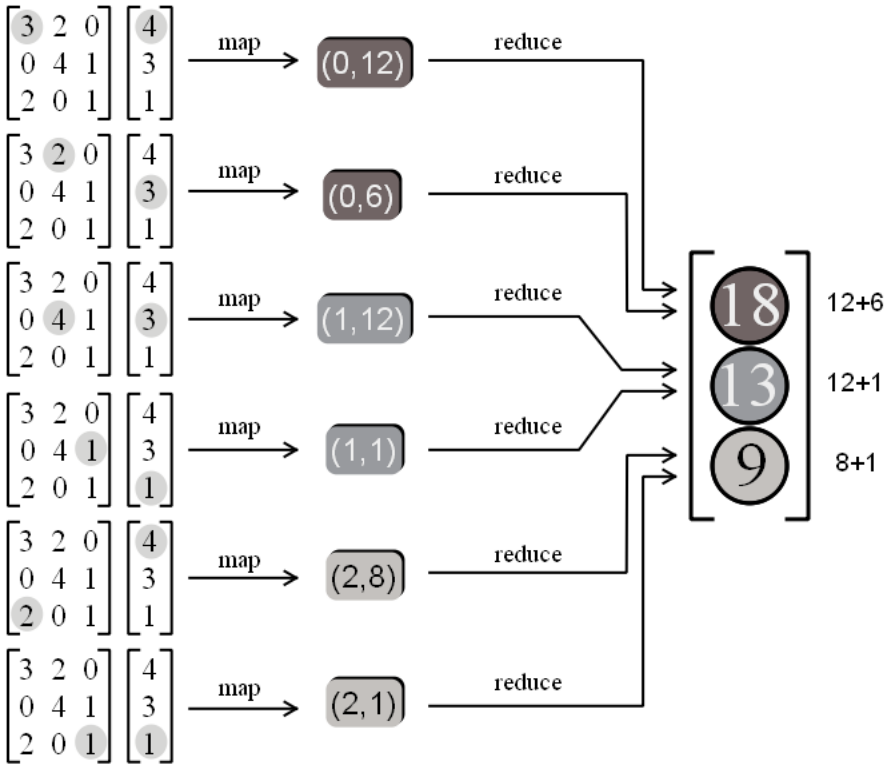
MapReduce is suitable for problems that involve numerous elements.

1. **Matrix-Vector Multiplication**

   Map: For each element in each input emit ((i,k), M[i,j]) and ((i,k), V[j,k])

   Reduce1: First multiply M[i,j] with V[j,k]

   Reduce2: Sum values for key (i,k)



2. Relational Operations

Selection	**Map**	Emit row (r,r) if r satisfied the test condition.
	Reduce	Identity function. Pass input to the output.
Projection	**Map**	Emit (r_s, r_s) where r_s contains only the attributes that are in s for the row r.
	Reduce	Reduce [r_s, r_s, r_s, ...] into single r_s.
Union/ Intersection	**Map**	Emit each row as (r, r).
	Reduce	Union: Reduce each [r] or [r,r] values to r. Intersection: Reduce [r] to ∅ and [r,r] to r.
Natural Join	**Map**	For (r, k) in T1 emit (k, (T1, r)). For (k, c) in T2 emit(k, (T2, c)).
	Reduce	Reduce [(T1,r), (T2,c)] to (r, k, c).
Grouping/ Aggregation	**Map** **Reduce**	See examples 2 and 3 above in “MapReduce Examples” section.

2.3.5 Hadoop Ecosystem

HDFS

- Hadoop Distributed File System (HDFS) is a distributed file system designed for high throughput.
- When a file is loaded into HDFS, it is replicated and fragmented into “blocks” of data, which are stored across the cluster nodes; the cluster nodes are also called the DataNodes.
- The NameNode is responsible for storage and management of metadata, so that when MapReduce or another execution framework calls for the data, the NameNode informs it where the data that is needed resides.
- This enables reliable and rapid access.
- NameNode manages file system in memory (inodes) and authorization to file system.
- So DataNodes, maintain block integrity and send block reports to Namenode.

  [Architecture diagram]

Hbase

- HBase “is an open-source, distributed, versioned, column-oriented store” that sits on top of HDFS.

- HBase is based on columns rather than rows.
- This essentially increases the speed of execution of operations if they are need to be performed on similar values across massive datasets; for example, read/write operations that involve all rows but only a small subset of all columns.

Hive

- Hive provides a warehouse structure for other Hadoop input sources and SQL-like access for data in HDFS.
- Hive's query language, HiveQL, compiles to MapReduce and also allows user-defined functions (UDFs).
- Hive's data model is based primarily on three related data structures: *tables* correspond to HDFS directories that are divided into *partitions*, which in turn can be divided into *buckets*.

Pig

- Pig is a run-time environment that allows users to execute MapReduce on a Hadoop cluster.
- Pig Latin is a high-level scripting language on Pig platform. Like HiveQL in Hive, Pig Latin is a higher-level language that compiles to MapReduce.
- Pig's data model is similar to the relational data model, but here tuples can be nested. For example, a table of

tuples can have a table in the third field of each tuple. In Pig, tables are called bags.

- Pig also has a “map” data type, which is useful in representing semi-structured data such as JSON or XML.”

Sqoop

- Sqoop (“SQL-to-Hadoop”) is a tool which transfers data in both ways between relational systems and HDFS.
- Sqoop can be used to import data from external structured databases into HDFS or any other related systems such as Hive and HBase.
- On the other hand, Sqoop can also be used to extract data from Hadoop and export it to external structured databases such as relational databases and enterprise data warehouses.

Limitations of Hadoop

1. HDFS cannot be mounted directly by an existing operating system. Getting data into and out of the HDFS file system can be inconvenient.
2. Hadoop security model is limited to HDFS file system permissons and is disabled by default due to complexity.
3. Hadoop framework is not fit for small data.
4. Stability issues in the API.

2.3.6 NOSQL

1. A NoSQL (not only SQL) database provides a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases.
2. NoSQL is schema-free.
3. Relational databases are not designed to scale and change easily to cope up with the needs of the modern industry to be frequently stored and accessed. NoSQL uses distributed architecture and works on multiple processors to give high performance.
4. NoSQL may not provide atomicity, consistency, -isolation, durability (ACID) properties but guarantees eventual consistency, basically available, soft state (BASE/refreshing), by having a distributed and fault-tolerant architecture.

Brewer's theorem

Consistency, Availability, Partition tolerance (CAP) theorem or Brewer's theorem states that it is not possible for a distributed system to provide all three of the following guarantees-simultaneously:

1. Consistency

All storage and their replicated nodes have the same data at the same time.

2. Availability

   Every request is guaranteed to receive a success or failure response.

3. Partition tolerance

   System continues to operate in spite of network failures.

Properties

1. Volume

   The need to horizontal scaling made organizations to move from serial to distributed parallel processing where big data is fragmented and processed using clusters of commodity machines.

2. Velocity

   Queries have to be read and written in real time. Random bursts in web traffic slows down the response for every user in relational databases.

3. Variability

   Uncommon data is frequent.

   Special attributes lead to sparse matrix.

   ALTER TABLE can not be executed when transactions are happening continuously.

4. Agility

   Relational databases have to do multiple joins and handle nested repeated subgroups for complex queries along with object-relational mapping which makes them slow. NoSQL does not require schema, or foreign key and thus costly join can be avoided and can be easily fed to MapReduce for processing queries.

2.4.3 Types and Applications of NoSQL Data Stores

1. **Key−Value Store**
   - Data is stored in schema-less format using key-value pairs. Key can be URL, file path, REST call, SQL query and value can be String, JSON, BLOB etc.
   - Applications: dictionary, image store, lookup tables, cache query.
   - Examples: Redis, Amazon Dynamo, Azure Table Storage (ATS), Memcache.

   Amazon DynamoDB
   - Developers create a database table which is stored at multiple servers with replication across multiple zones.
   - For consistent and fast performance, the data are stored in the key−value store and then moved to RDBMS.
2. **Column Family Store/Wide Column Store**
   - Store data tables as section of columns rather than rows which is efficient for sparse matrix systems.
   - Column stores are used in OLAP systems for their ability to rapidly aggregate column data with high performance and highly scalable architecture.
   - Examples: BigTable, Apache Cassandra, HBase

Google BigTable

- It represents sparse data table as section of columns.
- Data is stored in a distributed, persistent, multi-dimensional sorted map indexed by a row key, column key and a timestamp.

3. **Document Store**

- Document store is more complex than key-value store.
- It stores, retrieves, and manages document-oriented semi-structured information like hierarchical tree-like data structures.
- Examples: MongoDB, CouchBase, and CouchDB.

MongoDB

- It changed the data model from relational to document based, to achieve speed, manageability, agility, schema free horizontally scalable JOIN free scalability.
- Relational databases like MySql or Oracle work well with indexes, dynamic queries and updates. MongoDB works similarly but can index an embedded field.
- Data is stored in JSON documents. JSON model seamlessly maps to native programming languages and allows dynamic schema which helps the data model to evolve. RDBMS have fixed schema that limits the evolution of data model over time.

4. **Graph Store**

- Designed for data whose relations are represented as a graph with interconnected elements.
- Use graph database to store their data.
- Applicable in in social networks and rule-based engines like Facebook, LinkedIn, Twitter, YouTube, and Flickr.
- Examples: Neo4j, AllegroGraph, TeradataAster.

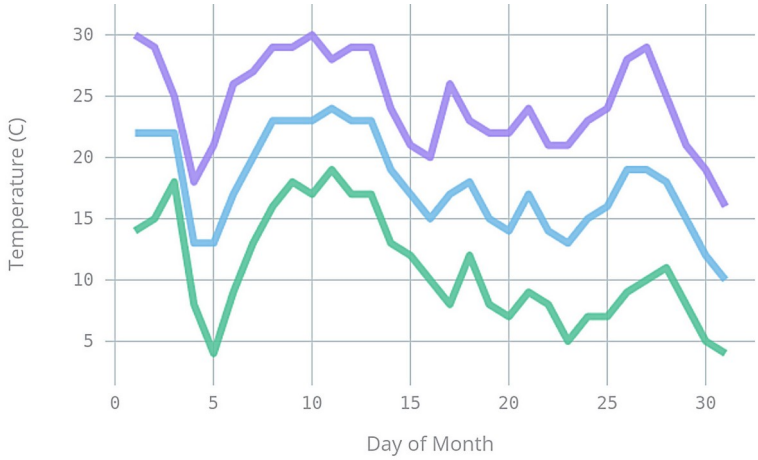
Neo4j

- Data can be shared among multiple connected domains.
- It stores, processes, and allows querying connections efficiently for ‘join’ like navigation operations to quickly traverse millions of connections per second per core. Traditional databases compute such relationships expensively at query time.
- Accessing already persistent connections is extremely fast. The property graph contains nodes (entities) that are connected and can hold any number of key- value pairs (attributes).
- Nodes contain labels that represent roles, relationships, metadata, index and constraints.

2.5 Visualizations

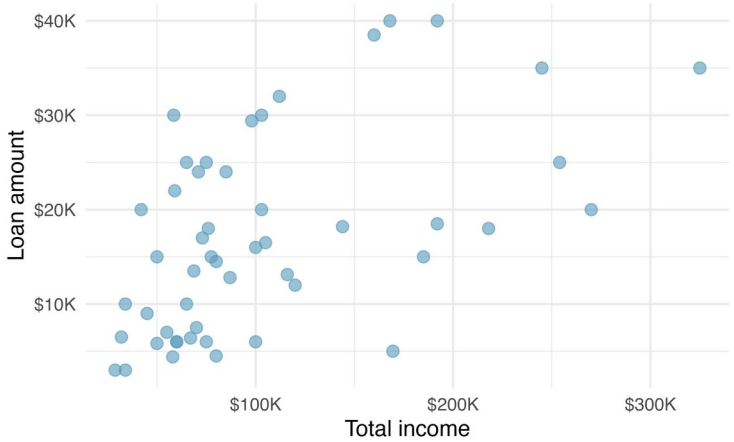
2.5.1 Line Chart

It is one of the simplest charts that can be used to display information as a series of data points connected by a line.



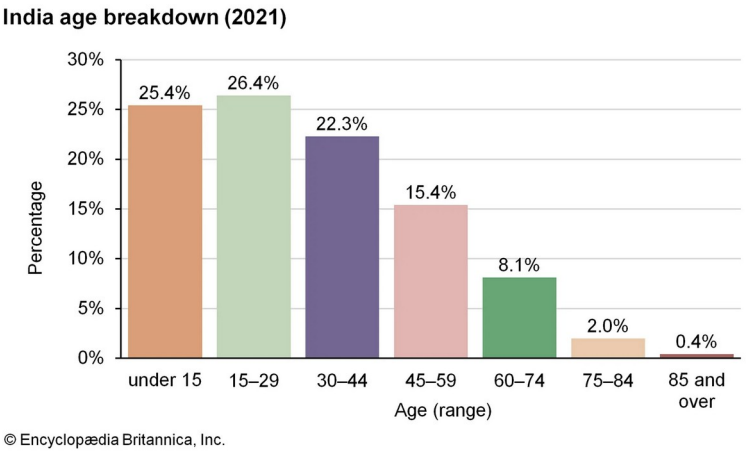
2.5.2 Scatter plot for 2D visualization

It can be used to visualize two variables along the X and Y axes. Scatter plots and are useful for identifying the relationships between two sets of data, for example fitting a regression line for bivariate data.



2.5.3 Bar Chart

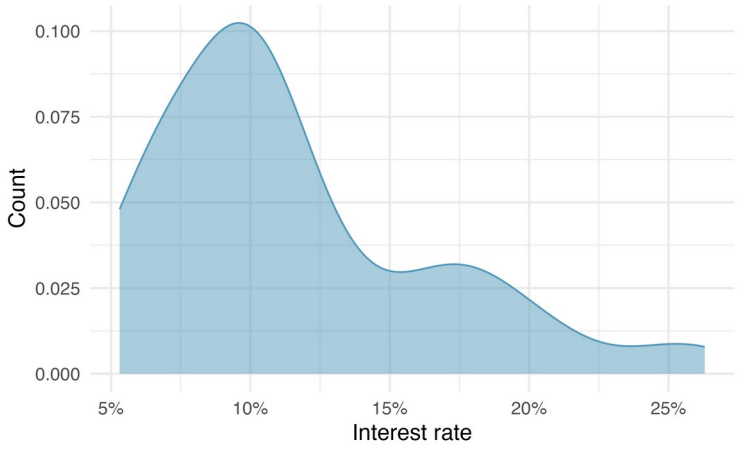
Bar chart can be used to display grouped data as bars with lengths proportional to the values represented. It is popularly used to display histogram of binned counts plotted as bars.



2.5.4 Density plot

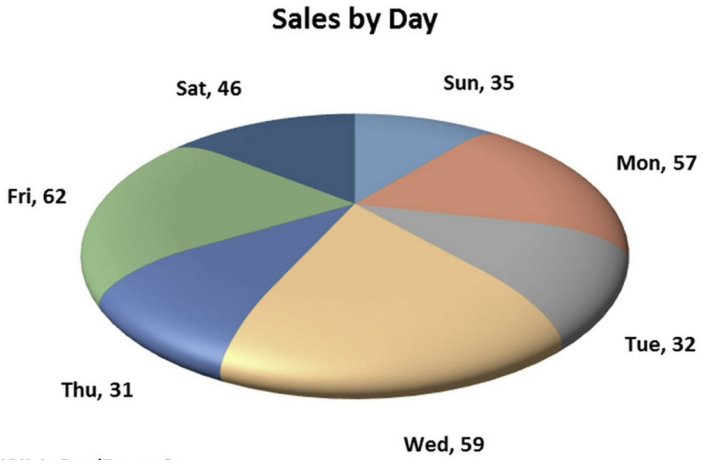
It is a smoothed out version histogram.

Based on number of peaks it can be unimodal, bimodal, and multimodal.



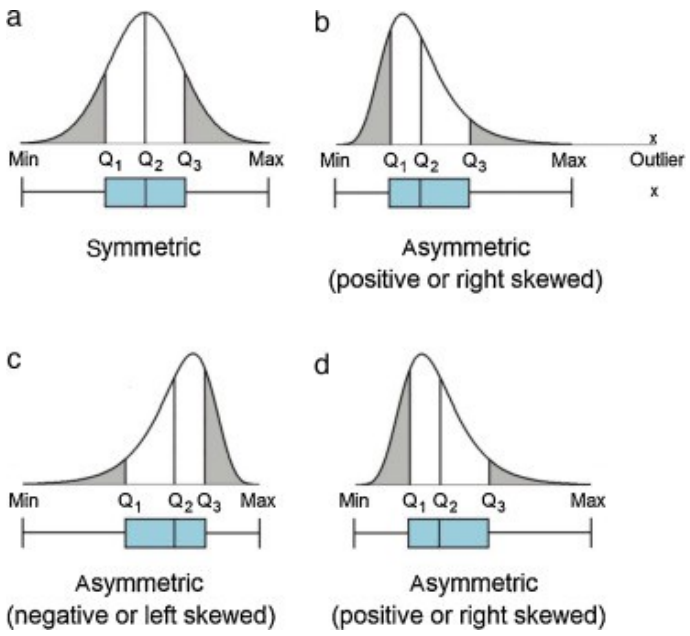
2.5.5 Pie Chart

Pie chart is used to display numerical proportions on a circle where the arc length is proportional to the quantity represented.



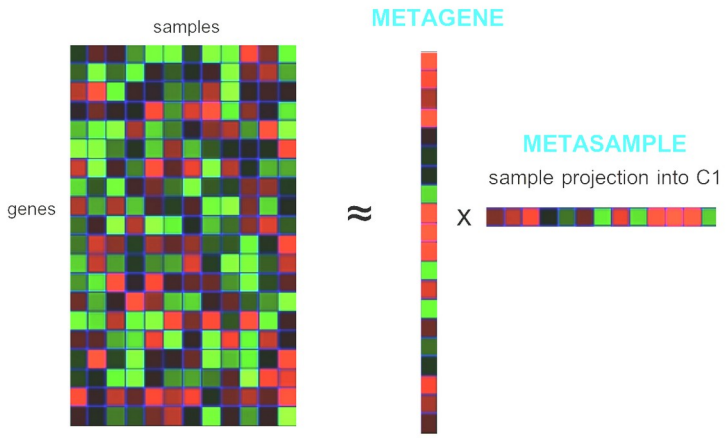
2.5.6 Box plot

It is used to display the minimum, medium and maximum for a data set. In a box plot, the whiskers denote the extremes of the dataset and the middle line is the median. The box goes from the first quartile to the third quartile of the data set.



2.5.7 Heatmap

It is used to plot a color-encoded matrix.



2.5.8 Pair grid

It is used for plotting pairwise relationships in a dataset.