

Course Book



LEVERAGING DATA SOURCES & DATA MINING

DLMDMEDM01

iu

INTERNATIONAL
UNIVERSITY OF
APPLIED SCIENCES

LEARNING OBJECTIVES



The **Leveraging Data Sources and Data Mining** course book provides key aspects of data mining and its methods. It will introduce information about the data mining process, offer a discussion about why data is important to businesses, and advise on what processes should be performed on data to obtain useful and business-related information. In addition, the topics of data modeling, data evaluation and data deployment, determining the data modeling quality, and presentation of data mining results are covered.

This course book provides information about ensuring the quality of data and preparing it in such a way that analyses and modeling can be performed. With this in mind, the course book addresses data collection, data selection and cleaning, as well as taking care of missing values and ensuring data consistency. Retrieval of data is also an important part of data mining for business-related purposes. To extract data from various sources, you will first learn about data types and how to use queries for data extraction, how to perform data streams mining, and mining big data.

Various methods will be introduced to help master data mining. Accounting statistical methods that deal with descriptive, correlation, and regression analysis are provided. To understand the differences between data warehousing and data mining, machine learning and event processing techniques are explained to extract meaningful or predictive information using data. In addition, the course book investigates mining data from web sources and data lakes. After you have an overview of the above-mentioned methods, techniques, and implementations, the course book will move on to discuss the data economy, legal requirements, and usage guidelines associated with data mining.

UNIT 1

DATA MINING PROCESS

STUDY GOALS

On completion of this unit, you will be able to ...

- identify the importance of data for businesses.
- understand what data are and the overall processes used to convert them into information and knowledge.
- apply descriptive data analysis.
- describe data modeling and its purpose.
- evaluate the quality of data modeling.
- present data analysis results.

1. DATA MINING PROCESS

Introduction

The amount of data available to us is rapidly growing, and data management is now essential to business success. Gut-instinct, traditional decision-making processes are not effective anymore: Data-driven and evidence-based decision-making are now needed to attain business objectives. To provide facts to data-driven decisions, the steps from data gathering analysis and reporting need to take place transparently. To answer the question of how data-driven decisions should be made by businesses, the data relevant to prioritized business objectives must be analyzed, desired knowledge must be extracted from underlying data, and conclusions must be drawn. Those conclusions can then be used to set the strategies and identify the measures of performance in business.



Prior to the mentioned steps, it is critical to have a good understanding of data and its journey from where it starts to how it can help make data-driven decisions. It is essential to understand data types in terms of their organization and format. In addition, large-scale or big data characters, such as volume, velocity, and variety, play an important role in how data are treated and modeled.

The modeling in this unit concerns how descriptive and predictive data modeling is done. Descriptive modeling focuses on statistically looking at data points to get a global understanding. This is when predictive modeling investigates what conclusions can be drawn based on historical data. Model evaluation is needed to ensure accuracy, precision, and sensitivity based on the metrics identified when evaluating a classification model. In addition to data classification, regression models with calculation of mean square error and correlation coefficients are explored in this unit.

1.1 Role of Data in Businesses

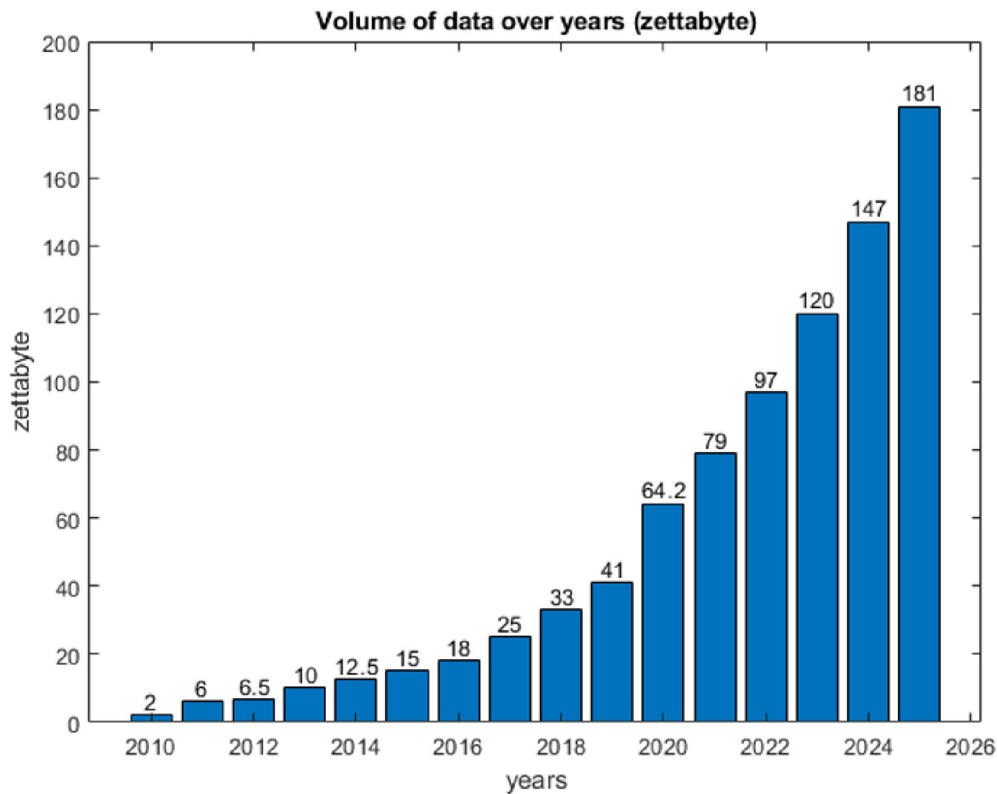
It's said that knowledge is power, which is an important concept in business because it leads to decisions and actions. To obtain knowledge, understanding, and awareness of a subject, information is needed. Information is essentially processed data or facts. The earliest example of data use can be traced as early as 19,000 BCE: The Ishango bone is thought to have acted as a tally stick, demonstrating data collection and storage (Crevecoeur et al., 2016).

Historical data are important and their abundance is constantly growing. This includes the written content we read every day, updates we post in social media platforms, products we buy, and videos we watch. According to Statista (2022), the amount of generated, ingested, copied, and consumed data will reach approximately 180 **zettabytes** by 2025.

Zettabyte

The number of bytes in a zettabyte is 2^{70} . One zettabyte can also be expressed as one sextillion bytes.

Figure 1: Data Volume Over Time



Source: Somayeh Aghanavasi (2024), based on Statista (2022).

Traditional Decision Process

In ancient times, business owners and medical doctors were making decisions based on a method called symptomatic diagnosis. It means doctors treated the patients based on the observation of symptoms. In the context of business, professional sales people, marketing managers, and business owners use their experiences, intuition, and knowledge to make decisions. This is generally referred to as a leader's **gut instinct**.



Gut instinct
This is an immediate or basic feeling that occurs without logical rationale.

Data-Driven Decision-Making

In contrast to traditional decision making, data-driven decision-making in business uses facts, metrics, and data to help with strategic decision-making. It further helps businesses to align decisions with goals, objectives, and initiatives. For instance, **ecommerce** sites use data to manage profits and sales. If you purchased an item or saved items as favorites in a web shop, perhaps you received a product recommendation to your email. These recommendations can be based on past purchases, ranks, reviewed items, or items viewed during a visit to the site. Some web shops use click-through rates as a metric to gain insight on customer engagement, as well as **open rates** and **opt out rates** to make decisions about pushing recommendations. Brynjolfsson et al. (2011) examined the performance of the firms that made decisions based on data-driven methods and found that firms adopting data-driven decision-making have 56% higher output and performance.

Open rate
This is the percentage of customers who opened an email.

Opt out rates

Also known as unsubscribe rates, this is the number of unsubscribes divided by the number of delivered emails.

How to Make Data-Driven Decisions

The steps below are recognized as essential for making data-driven decisions (Provost & Fawcett, 2013):

1. Identify objectives
2. Prioritize objectives
3. Find and present relevant data
4. Extract knowledge and conclusions from the data
5. Set strategies based on findings
6. Measure performance and repeat the process

With the given overview on the history and future of data, it becomes evident that data plays a critical role in life and business. They help companies form multidimensional insights from production, sales, marketing, and even planning quarterly objectives. Data support predicting trends in customer behavior and determining enterprise values.

1.2 Understanding Data

Data are the smallest units of facts to be used as a basis for calculation and reasoning. Data contain raw facts in the form of numbers and characters that are then processed to capture useful information. In the pursuit knowledge, data are discrete collections of values that have information in the form of quantity, statistics, facts, and quality. Data collection is usually done by observation, measurement techniques, or querying data from databases (Canteloup et al., 2020).



Consider the raw data of a purchase in a retail store. Prior to any analysis or structuring, data are captured and stored in repositories. Types of data could include the time stamp, a list of items purchased in sequential order, the type of the purchase (e.g., online, in-store), the type of payment (credit, cash), or type of card used if a purchase was made online. Another example is when raw data is collected by a camera. It collects time; images; frames; video signals; and red, green, and blue (RGB) or cyan, magenta, yellow, and key (CMYK) color scales.

Before focusing on the definition of big data, the terms “dataset” and “data base” need to be differentiated. “Dataset” refers to sets of data records stored for analysis or modeling. A “data base” is a framework organized to store structured data accessibly for different purposes. Data bases are managed by a database management system (DBMS) and allow users to work with large amounts of data more efficiently.

Big Data

As the name implies, the main characteristic of big data is size. However, there are other characteristics that can be assigned to big data: It is high-volume, high-variety, high-velocity, and provides veracity (Gandomi & Haider, 2015).

Volume is the quantity or magnitude of the data. Big data can be multiple terabytes and petabytes. A terabyte is 1000 gigabytes (GB) or one trillion bytes. This can be visualized as an hour of four kilo (4k) video or 16 million photographs. This is when a petabyte is equal to 1024 terabytes. However, the volume of big data is relative and not precise. What's considered big data today may not be considered as such in the future because the storage capacity of data is increasing.

Variety refers to how structurally heterogeneous a dataset is. In other words, it refers to the diversity of the content of the dataset and to identifying how much of the data are structured, semi-structured, and unstructured. Structured data are tabular and account for only a small percentage of all data. Unstructured data constitutes most of the datasets. This is a type of data that need a process for analysis. Examples of unstructured data are audio, video, images, and text. Semi-structured datasets are partly structured. This is the case when a dataset contains user-defined tags in the documents. Extensible Markup Language (XML) is an example of a semi-structured data format.

Velocity refers to the rate of capturing, storing, and analyzing data. Recent advances in technology, such as smartphones and sensors, offer high rates of data creation, and this leads toward the need for real-time analytics (Gandomi & Haider, 2015).

Some literature includes veracity as the fourth characteristic of big data. Veracity refers to data accuracy, consistency, quality, and trustworthiness. For example, in some online account registrations, users can provide false contact information. Increasing veracity in data collection is required to reduce the amount of data cleaning needed for analysis (Geng, 2017). It follows these seven steps:

1. Acquisition
2. Extraction
3. Cleaning
4. Annotation
5. Aggregation
6. Modeling and analysis
7. Interpretation


Data Types

Data is distinguished by many dimensions, the most important of which is organization. Data can be organized, or classified, into three types: unstructured, semi-structured, and structured. However, this classification is indistinct and continuous, meaning there are no hard boundaries separating the three classes. The type of data mainly refers to how it should be captured, stored, and analyzed. Structured data comes with a high degree of organization. The elements are addressable and the data typically reside in a database. These type of data are also highly specific. An example of structured data is the data in Excel sheets, comma-separated values (CSV) files, or relational database tables. In contrast, unstructured data have no predefined organization or specific format. Examples of unstructured data are .mp4 video files, Portable Document Files (PDF) files, .mp3 sound files, or plain text files. Semi-structured data is something in between structured and



unstructured data. This means data come with some degree of organization. For this type of data includes XML files, JavaScript Object Notation (JSON) files, and HyperText Markup Language (HTML) files, all of which include some degree of organization (Sullivan, 2020).

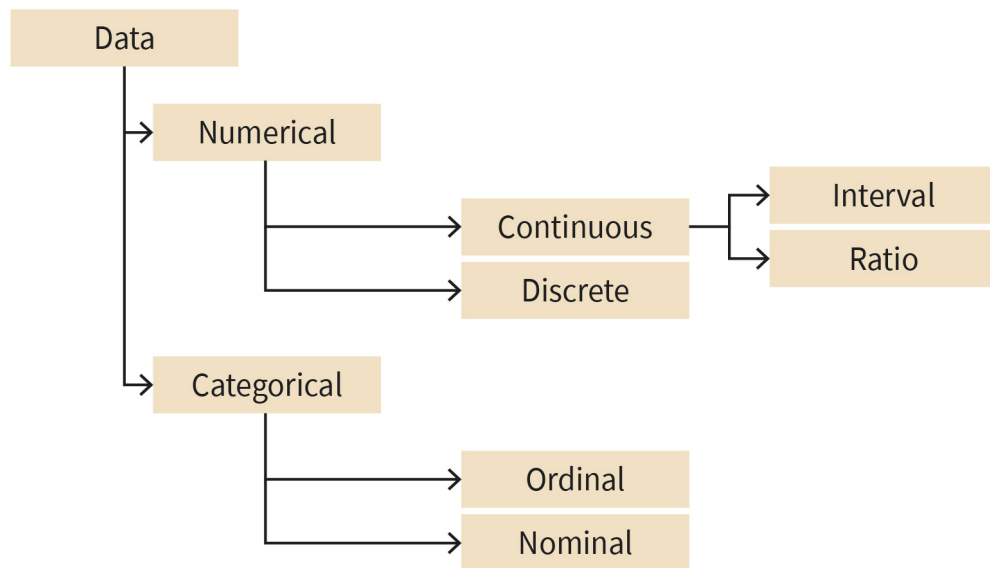
General Data Formats

Data can be stored and represented in different formats. Typical formats used in common data processing and programming languages are integer (342, 66, 1), character (a, !, ^), date (2022-12-25, 1981-08-30 06:15:54, **SYSDATETIME()**), real (1.43E5), long(-7424000322000914), short (3820), string ("text"), and Boolean (false, true). 

Data Types in Statistics

In statistics, there are two main types of data: numerical and categorical. Numerical data are made of numbers (e.g., age, weight, length of a road, size of a shoe). Categorical data are made of words and belong to a category (e.g., eye color, gender, ethnicity, blood type). The first data type (numeric) can be continuous or discrete. Continuous data has an infinite number of possible values that can be measured and comes in two types: interval and ratio. Discrete data are finite values that are not continuous and can be counted (e.g., the number of students in a class or the empty seats on a train). The second main data type (categorical) has two types as well: ordinal and nominal. Nominal data are also called qualitative data, which are not classified with an order or rank (e.g., names of students, nationalities of students, female/male, hair color). Conversely, ordinal data have a predetermined order (e.g., customer satisfaction rate with the categories "extremely dislike," "dislike," "neutral," "like," and "extremely like").

Figure 2: Types of Data



Source: Somayeh Aghanavesi (2024) 

Why are data important? Data allow visualization of the relationships between different facts, locations, systems, and entities. With data, the identification and differing of various entities or classes are possible. With the ability to identify the patterns, trends, differences, and classes, problems are identified and solutions to those problems emerge with a fact-driven or data-driven development approach.

1.3 Modeling



The goal of modeling data is to use historical data to decide future data. In other words, it finds the patterns and trends in raw data via algorithms. The algorithms examine the data and formulate them according to the characteristics of the raw data being examined. Descriptive and predictive approaches are the two most common types of data modeling.

Descriptive Modeling

Descriptive modeling is usually done as the first step to provide data properties and get a summary of how the data looks overall. It uses some common statistics to identify the clusters of identical objects and associations. It works on converting the data and summarizing it into a significant form that can be used for reporting the first line of the analysis or modeling the data further. Summarizing the data helps one understand it. By applying some association rules, it finds major relationships within large datasets where the data is consistent.

There are various tools and programming languages offering the employment of descriptive modeling. They often use descriptive analytics or descriptive statistics.



Python performs statistical processes to describe basic data features. Central tendency and dispersion are the properties of descriptive statistics, where the first (central tendency) identifies and characterizes the central value based on data distribution. The mean, median, and mode are considered central tendency measures. The other property, dispersion, investigates the distance between the members of the distribution from the center and from each other.

Central tendency

To find mean as a central location of the data in Python we first need to import the statistics module.

```
>>> Import statistics as stats
```

The example below shows how to calculate the mean as an arithmetic average. Average is calculated by adding all the values within a set (a_i to a_n) and dividing it to the number of all numbers (n) using the formula

$$A = \frac{1}{n} \sum_{i=1}^n a_i$$

where A is the arithmetic mean, n is the number of values, and a_1 is the dataset values.

To calculate the average/mean, a dataset, such as the following, is considered:

```
>>> numbers = [4, 23, 56, 12, 41, 41, 174, 12, 41, 0, 30, 41, 56]
```

To calculate the mean, the statistical module should be used together with the mean function.

```
>>> stats.mean(numbers)
```

To calculate the median, the number of items in a set must be identified. This is because, in a dataset, the median is considered as the middle value and the odd or even number of the items in the set determine how it is calculated. It follows the formula

$$\text{Median}(x) = \begin{cases} X\left[\frac{n+1}{2}\right] & \text{if } n \text{ is odd} \\ \frac{X\left[\frac{n}{2}\right] + X\left[\frac{n}{2} + 1\right]}{2} & \text{if } n \text{ is even} \end{cases}$$



— where X is the dataset ordered set of values, and n is the number of items in the set. Assuming the set of numbers created before, the median can be calculated using the below code in Python:

```
>>> stats.median(numbers)
```

Mode refers to the most frequent number. Assuming the set of numbers mentioned above, the number that occurs most often is 41. To calculate mode in Python, use the code below, which embeds statistics modules:

```
>>> stats.mode(numbers)
```

The mode function can be used with string data types as well. For this, the set should contain characters created with single quotes, such as `set = ['A', 'B', 'C']`.

Dispersion

Variance is a measure of dispersion that shows how spread out the numbers in a set are from their central value. It's the expectation of squared deviation of one random variable from the mean. If the meaning is a sample mean, then the variance is the sample variance. Similarly, if the mean is the population mean, then the variance is the population variance. It's calculated using the formula

$$S^2 = \frac{\sum (x_i - \bar{x})^2}{n - 1}$$


where S^2 is the sample/population variance, x_i is the value of the single observation, n is the number of observations, and \bar{x} is the mean value calculate using all observations.

The sample variance and population variance are calculated using different functions. Sample variance used the variance function from the following statistics module:

```
>>> stats.variance(numbers)
```

Population variance is calculated using the pvariance function:

```
>>> stats.pvariance(numbers)
```

Standard deviation represents the amount of variation that exists in the population or sample dataset. If the standard deviation is small, it indicates the values in the dataset are close to the average value. In contrast, if the standard deviation is large, data points are largely spread around from the average or mean. The calculation of standard deviation follows the formula 

$$\sigma = \sqrt{\frac{\sum (X_i - \mu)^2}{N}}$$

where σ stands for sample or population standard deviation, N is the length or size of the sample or population, X_i is the single data point from dataset, and μ is the average or mean of the dataset.

In Python, sample standard deviation and population standard deviation are calculated using different functions from statistics module. For sample standard deviation,

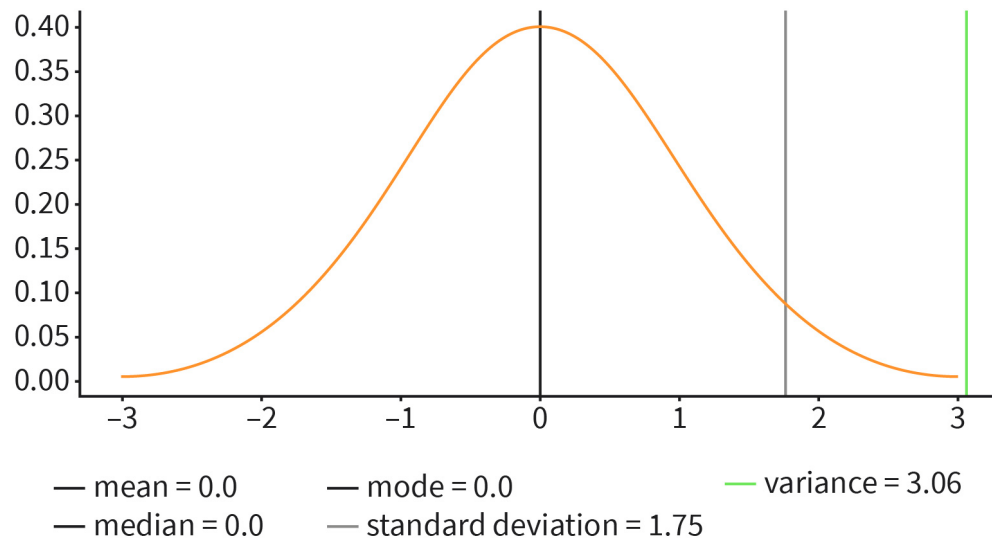
```
>>> stats.stdev(numbers)
```

while for population standard deviation,

```
>>> stats.pstdev(numbers)
```

To facilitate understanding the central tendency and dispersion, consider two examples. The first example is depicted in the graph below, which shows a dataset created from normal distribution. In a normal distribution, data are symmetrically spread around their means, while most of the values are clustered in the middle. The distribution line is shown in orange. The values (i.e., the mean, median, and mode) for this distribution are all zero. The variance, shown in green, indicates how much deviation the dataset has from the center or mean value. Standard deviation is shown in gray with a value of 1.75.

Figure 3: Normal Distribution



Source: Somayeh Aghanavesi (2024).

To create the above graph, the Python code below is used. First, the required packages are loaded. Then, the distribution of data points is created using the **three-sigma limit** method. Then the plotting of the distribution, its central tendency, and its dispersion are calculated. At the end, the measures are planned.

Three-sigma limit

This is a statistic method wherein 99.7% of the data are within three standard deviations from the mean.

```
import matplotlib.pyplot as plt
import numpy as np
import SciPy.stats as stats
import math

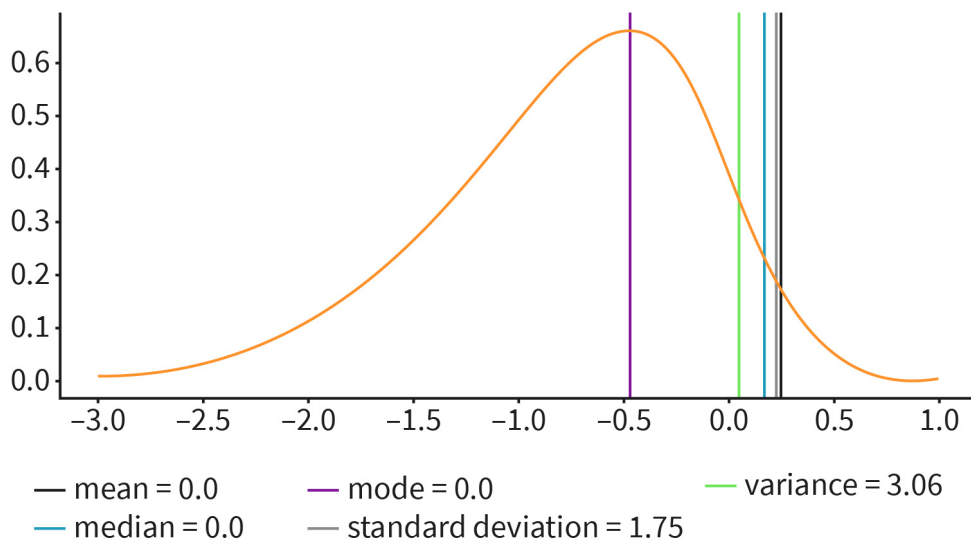
mu = 0
variance = 1
sigma = math.sqrt(variance)
p = np.linspace(mu - 3*sigma, mu + 3*sigma, 100)
plt.subplots(figsize=(12,4))
plt.plot(p, stats.norm.pdf(x, mu, sigma), color = 'Orange')
mean = plt.axvline(x=np.mean(p), color = 'black')
median = plt.axvline(x=np.median(p), color = 'black')
mode = plt.axvline(x=0.0, color = 'black')
standard_deviation = plt.axvline(x=statistics.stdev(p), color = 'gray')
variance = plt.axvline(x=np.var(p), color = 'green')
plt.legend(loc="upper left")
```

Negatively skewed

This means the distribution is tilted more towards the negative value.

In the second example, a **negatively skewed** distribution is created to illustrate different central tendencies and dispersion values. Five measures are shown this graph. This time the mean, mode, and median are not similar. However, the variance and standard deviation are close to each other.

Figure 4: Negatively Skewed Distribution



Source: Somayeh Aghanavesi (2024).

To create the above graph with central and dispersion measures, use the code below (written in Python). The packages needed to calculate the measures and visualize the graph are loaded. Function `skew_norm_pdf` is used to create the skewed distribution. As in the first example, the central and dispersion measures are calculated.

```
import numpy as np
import matplotlib.pyplot as plt
import SciPy.stats as stats
import statistics

SKEW_PARAMS = [-3]

location = 0.0
scale = 1.0
x = np.linspace(-3,1,100)

def skew_norm_pdf(x,e=0,w=1,a=0):
    t = (x-e) / w
    return 2.0 * w * stats.norm.pdf(t) * stats.norm.cdf(a*t)

plt.subplots(figsize=(12,4))
for alpha_skew in SKEW_PARAMS:
    p = skew_norm_pdf(x,location,scale,alpha_skew)
    plt.plot(x,p, color = 'Orange')
    mean = plt.axvline(x=np.mean(p), color = 'black')
    median = plt.axvline(x=np.median(p), color = 'blue')
    mode = plt.axvline(x=-0.47, color = 'magenta')
    standard_deviation = plt.axvline(x=statistics.stdev(p), color = 'gray')
```

```
variance = plt.axvline(x=np.var(p), color = 'green')
plt.legend(loc="upper left")
```

```
plt.legend([mean, median, mode, standard_deviation, variance],
['mean = 0.246', 'median = 0.167', 'mode = - 0.47', 'standard deviation
= 0.228', 'variance = 0.05'])
```

However, descriptive modeling includes as many analyses and visualizations as required to get an overview of the data. In Python, there could be lots of approaches to discover data sequences, identify associations, and visualize clusters. Below, we include another example that loads sample dataset of irises to visualize a descriptive outcome. This dataset contains data about the species expressed into four numerical columns and one class or categorical column. The below code loads the data, extracts the columns, and visualizes one feature, called “petal_len” (petal length), over its descriptive data (mean).



```
import pandas as pd
from sklearn import datasets
import seaborn as sns
import matplotlib.pyplot as plt

iris = datasets.load_iris() #load dataset

iris_df=pd.DataFrame(iris.data) #convert to data frame
iris_df['class']=iris.target #Form the class column
iris_df.columns=['sepal_len', 'sepal_wid', 'petal_len', 'petal_wid', 'class']
iris_df.dropna(how="all", inplace=True) # remove any empty lines

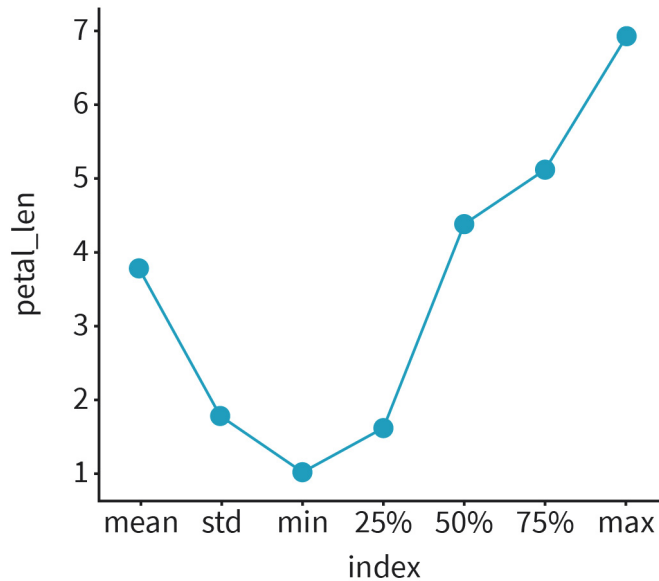
total_columns = iris_df.columns # all columns
num_col = iris_df._get_numeric_data().columns
cat_col = list(set(total_columns)-set(num_col)) # categorical columns
describe_num_df = iris_df.describe(include=['int64', 'float64'])
describe_num_df.reset_index(inplace=True)
describe_num_df = describe_num_df[describe_num_df['index'] != 'count']
```

To visualize the descriptive data:

```
#set up the plot
sns.factorplot(x="index", y="petal_len", data = describe_num_df)
plt.show()
```

The resulting plot is shown in the figure below.

Figure 5: Petal Length



Source: Somayeh Aghanavasi (2024).

Predictive Modeling

As a part of data mining, predictive data modeling uses historical or known data to predict unknown events. The models are often statistical techniques or algorithms. For example, assume that a number of dairy products are collected from different factories. Every product has its own specific character. They are different in form, texture, and flavor. When given the data about the product, predictive modeling uses a neural network or random forest model to predict the origin of a product. Data modeling, such as multi-class/binary classification, regression, and time series analysis, also fall into the category of predictive modeling.



Setting up a predictive model takes some steps. The data used in predictive modeling should be clean and sufficient. An important step is to explore the dataset. In Python the `info()`, `Describe()`, `shape()` functions can be used to get general information about the dataset.



There are two main types of supervised and unsupervised modeling for prediction of unknown values. With supervised modeling, one column of the dataset contains the desired output. The model gets trained using data and then with target values. Then, the model is tested using some unseen data that excludes the target values to predict the outcome based on what was learned in its last step. In unsupervised models, there is no target value, or it's not known. So, the model identifies the trends to find out the classes or the predictive outcomes.



Training and testing the data are essential parts in supervised modeling. Training the data should be done on the known features that have a good relationship with the target value. Data testing is performed in two ways. K-fold cross validation and held-out data (Stone,

1977). Cross validation is a resampling technique that uses different proportions of the dataset to train and test. K-fold refers to the proportion the data are divided by. The process is iterative, meaning one proportion out of K will be used for testing and the rest are used for training. The process repeats for about 100 iterations every time trying a random fold to train and test.

Hold-out data validation is conducted by splitting the whole dataset into a train and test set. This evaluation type is good for large amounts of data. Usually, 80% of data are taken for training and 20% are kept for testing.

Target value

This is also called a target variable, response variable, dependent variable, or outcome variable.

Target value is a column in the dataset containing the desired output when the case of predictive modeling is supervised. With an unsupervised modeling, the target value is unknown and calculated by the algorithm. Therefore, during the early steps of preparing the data, the important features should be identified and extracted from raw data.

The model is built by splitting the dataset into training and testing sets. The better the data preparation, the better the outcomes of the modeling. There are multiple predictive methods to be used, such as linear regression, logistic regression, decision trees, gradient boosted model, neural networks, random forest, and support vector machines.

Logistic Regression

Let's illustrate the predictive modeling further by describing the logistic regression (Huerta-Manzanilla et al., 2021). Logistic regression is a statistical and binary classification model, which means the target values are considered as binary values of 0 or 1. The model predicts the binary outcomes, such as yes or no, based on the prior observations and estimates the probability of the target value. Since the outcome is a probability, the target value is between 0 and 1. Logistic modeling is presented with the following formula.

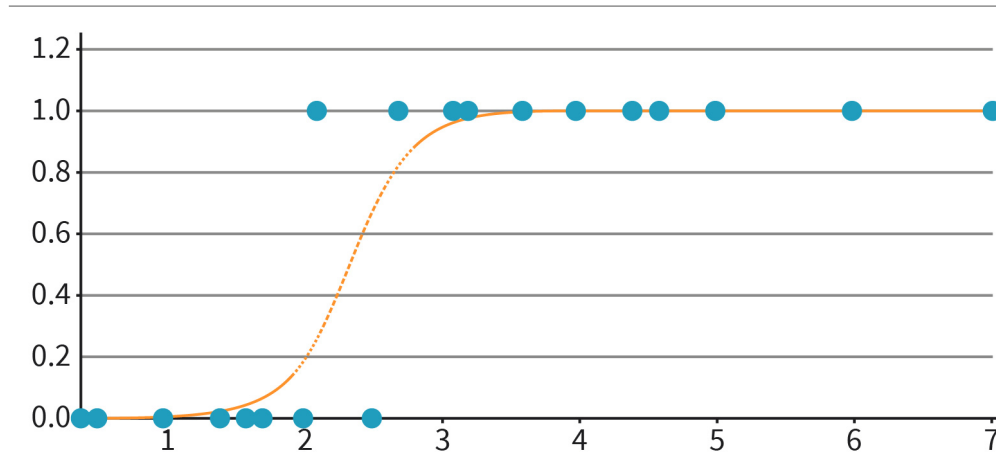
$$\log \left[\frac{p}{1-p} \right]$$

Odds

This is the measure of likelihood of a particular outcome.

where p is the probability of the event occurring with 0 or 1 cases. $1 - p$ is the probability of failure. The formula represents the logistic transformation that is applied on the **odds**, which is the probability of success divided by the probability of failure. It is also known as the link function. The transformation allows the model to be a nonlinear association between the two outcomes. Let's plot an example of logistic modeling for the case of predicting the pass rate of an exam when the data of study hours are analyzed.

Figure 6: Logistic Model



Source: Somayeh Aghanavasi (2024).

This plot includes 19 observations from students. The **x-axis** represents study hours and y-axis represents the exam outcome based on the hours. The logistic model (orange line) is fitted to the observations and represents the probabilities of failing or passing the exam considering the study hours. An important step is to map the business problems to data modeling tasks. Regressions models are good for predicting scores. They are usually used to estimate the effect of input variables on outcome. However, when predicting the categories, random forest data modeling would suit. Decision trees are often used to identify the variables affecting most of the categorization.



1.4 Evaluation

Model evaluation is an approach to quantify the performance of a model to ensure its quality. Depending on the data model task and its business-related objectives, there could be relative performance measures to be evaluated. If a data model performs a classification, then the measures of precision and recall are relative. If the model performs a scoring task, then the root mean square error (RMSE) would be an appropriate measure. Let's investigate the most used performance measures for classification and regression models.

Classification Model Evaluation



There are a number of ways to evaluate models, as will be shown in the following sections.

Confusion matrix

Classification model evaluation is performed using classification metrics. Classification metrics can be true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Assuming the table below, the columns represent the actual values, and the rows are the predicted classes.

Table 1: Confusion Matrix

Predicted values	Positive (1)	Negative (0)	
	TP	FP	Positive (1)
	FN	TN	Negative (0)
	Actual values		

Source: Somayeh Aghanavesi (2024).

True estimations occur when the model has predicted a positive or negative outcome correctly. False estimations by models are noted by *FP* or *FN*, indicating if positive or negative outcomes are incorrectly predicted. Having a confusion matrix, estimation of accuracy, precision, and recall/sensitivity scores can be calculated.

Accuracy is calculated by adding the true predictions from confusion matrix divided by all predictions. The formula below presents how accuracy is calculated. It measures how often the predictions are correct by providing a ratio of true predictions divided to all.

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

Precision is another evaluation score that measures how many of the predicted classes are positive. Precision is important for cases where *FP* is higher than *FN*. It's calculated using the following formula:

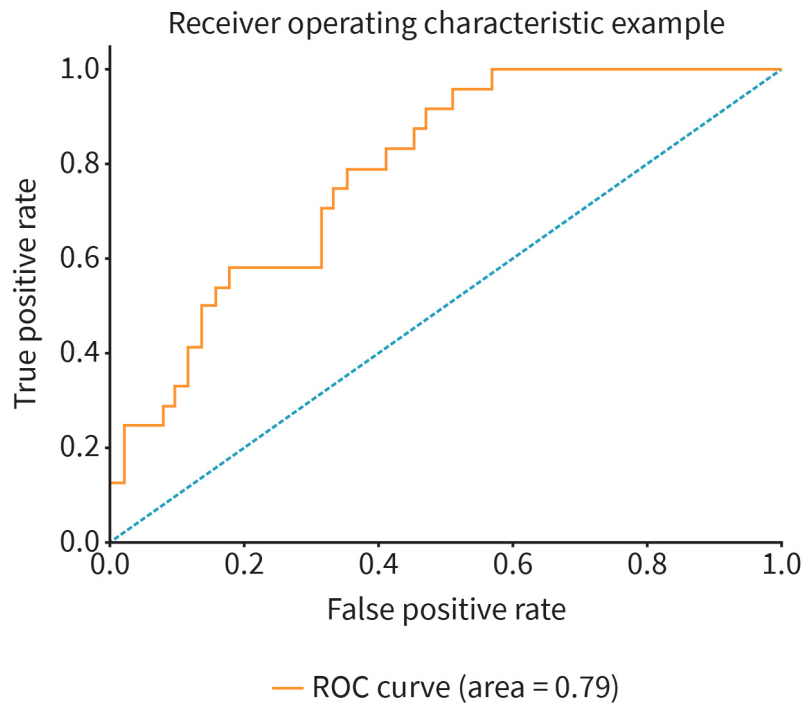
$$Precision = \frac{TP}{TP + FP}$$

The measure of recall or sensitivity score is calculated when a false negative is of concern. It calculates how many of the actual positive classifications were predicted correctly. This score is considered as important score in medical fields, where none of the correctly predicted scores should be missed. The formula below is used to calculate this score.

$$Sensitivity = \frac{TP}{TP + FN}$$

The receiver operator characteristic (ROC) curve is a measure that puts all the *TP*, *TN*, *FP*, and *FN* predictions together and provides a probability curve plotting the *TP* rate against the *FP* rate (scikit-learn, 2022). The plot is shown in the figure below.

Figure 7: ROC Curve



Source: Source: Somayeh Aghanavesi (2024).

Regression Model Evaluation

R squared is a measure to evaluate the regression models identifying how well the model fits the independent variables. It's the square of the correlation coefficient (R) presented with the formula below.

$$R^2 = 1 - \frac{RSS}{TSS}$$

Where RSS is sum of squares of residuals and TSS is the total sum of squares.

Mean square error (MSE) is also an important measure, as an absolute measure of goodness of the model fitness. It's calculated by the sum of square of predicted value errors minus predicted values, divided by the number of data points in dataset. See the following formula:

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

Both MSE and R^2 can be calculated in Python using Statsmodel and Sklearn packages.

1.5 Deployment



The processes in data mining are iterative. Once a solution is achieved, there could be another data mining process to start based on the new results deployed. Business-related questions trigger what kind of analysis should be performed. After running a round of analysis and deploying the results, there come new business-related questions to answer. It's important to keep track of the process and document the analysis together with the results step by step. We might try different approaches and apply various calculations. We should make sure to track the connections, from data to analysis, as this will prevent results from being lost.

To present data mining results, the audience needs to be identified. Is the audience from a data science background or are they from a business sector? Presenting results in technical way and using numbers and technical words to data science audience would be fine, but managers coming from a business perspective might find it difficult to grasp of numbers in detail. Therefore, the presentation of the results should be connected to the business questions and contexts the analysis is aiming to answer or relate to. The results should also be aligned with the business goals.

A common way to prepare a report to present the data mining results is to include the following steps in order (Cohen et al., 2013):

1. Provide an introduction of what the business question and the goal is.
2. Discuss the data of interest and include a description of the dataset.
3. Present the methods used to analyze the data.
4. Explain the important findings from the analysis together with data visualizations.
5. Provide the connection between the findings and the business context.
6. Explain the limitations of the method or data.
7. Provide either the next step to improve the results or the next business-related question to answer.



SUMMARY

Data play an increasingly important role for businesses. Traditional business decision-making included intuitional experiences and gut instincts, which could lead to mistakes and failures. New data science technologies help businesses to make decisions using facts obtained from data. To successfully make data-driven decisions and support businesses with insights from data, a comprehensive understanding of data and their modeling is needed.

There are various data types, including integers, double, and characters. In statistics, data are considered numerical or categorical. Understanding data types is essential to perform data analysis and modeling.

Prior to modeling data to extract information, their size, shape, and dimensions must be known. An early stage in data modeling is to perform descriptive data analysis. Python provides a wide range of packages to calculate central tendency and dispersion analysis using data. Central tendency analysis includes measures of mean, median, and mode, to identify the central points of a distribution. Dispersion is to identify the measures of variance and standard deviation to identify the variation and distance data has from its center. Data modeling includes predictive modeling as a measure to analyze historical data and provide insight for future data. To model data, an understanding of what variables are and whether they're dependent or independent is important. The model comes with better outcomes if the data are prepared beforehand. Data are prepared by making sure the related variables or features are included in the dataset and it is sufficient (i.e., there are not many null values and there are enough data points).

Data model evaluation ensures the quality of the model outcomes. Based on the model solving a classification problem or a regression problem, there could be different measures for its evaluation. For classification models, confusion matrix, accuracy, precision, recall, and receiver operator characteristic curve are used to measure the quality. For regression models, the two common measures of R squared and mean square error are used for evaluation.

To deploy the results from data analysis, there are necessary steps to be taken to ensure results are in line with business goals and questions. Technical terms and numbers in details are to be avoided when presenting the results to business managers who do not have knowledge on analysis methods. The report to present the outcomes of the analysis should include the purpose of the analysis, the connection to business goals, the questions to be answered, methods, a visualization of data, and the most important findings from data.

UNIT 2

DATA QUALITY AND DATA PREPARATION

STUDY GOALS

On completion of this unit, you will be able to ...

- identify and describe different data collection methods.
- understand the different approaches for selecting and sampling the data.
- apply the methods in Python to clean data prior to data analysis.
- describe what sparse and missing data are and how to handle them.
- describe what data consistency means.

2. DATA QUALITY AND DATA PREPARATION

Introduction

This unit focuses on the quality and preparation of data. Data quality is influenced right from the beginning, when they are collected. Questions, such as how many data are needed for collection and what type of data collection method should be used, are answered. Depending on the type of research, the type of data collection method is determined. Let's assume a research project about gathering the ideas or opinions of reviewers after watching a documentary. The purpose is to compile valid reviews and write an analytical article about them. Consequently, collecting their ideas through a survey or a face-to-face interview would be needed. The order and the design of the questions become important as well.

Conversely, what if the research is about examining the effects of drug on animals? Can the data be collected from a face-to-face interview? An accurate amount of the drug, the times the drug is to be given, and the number of animals participating become important. In this case, quantitative collection of data becomes necessary.

If data are already in databases, some sampling would be needed. A sample should be selected in a way that represents its entire population. So, the results of the analysis on a sample set can be generalized to its population. For the mentioned purposes, the important processes are data collection, or data selection.

Prior to using the collected, gathered, and sampled data in later processes (e.g., analysis, modeling, and data visualization) data must be clean and structured. Data cleaning techniques are provided by several libraries in Python. Besides cleaning the datasets, in this unit, handling of missing and sparse data is presented. Sparse data take space and add to the complexity of data analysis. Appropriate steps need to be taken to handle missing, sparse, and unused data.

2.1 Gathering Data

Data projects are composed of phases. Problem definition, data processing, modeling, evaluation, deployment, and conclusion. The collection of data and its planning becomes relevant in the early stages. To ensure the correct data are collected, and that there are enough of them, questions need to be asked. Data must be relevant to the business and project, and detailed questions should be based on the questions, metrics, and measures to be included. The next step is to identify what kind of data are needed or can be collected. Moreover, how much data would be enough to answer the questions asked? Basically, there needs to be enough data to see trends and patterns (Zozus, 2017).

From a statistical point of view, the amount of data to be collected depends on the sample size representing the population. It also depends on the expected variability, and the generalizability of the results. In other words, to make sure the results are not drawn by chance, statistical hypothesis testing is performed. To ensure the results are valid, a **p-value** is estimated, and it must be lower than five percent to ensure the quality of the results.

p-value

This is a statistical measure. It shows how significant are the results of the hypothesis testing.

In the process of collecting the data, the measurement methods to measure must be determined. After gathering the data, it is important to remember the following:

- the format of the data to be displayed
- the copies to be kept as backups
- its reporting format

Before looking into the different data collection methods, qualitative and quantitative data must be distinguished. Quantitative data refers to the data that can be counted and quantitatively measured or collected, whereas the qualitative data is about narratives, feelings, concerns providing context data. Quantitative data are countable, measurable, come in form of numbers, and are factual. They are analyzed using statistical analysis. Conversely, qualitative data are descriptive, dynamic, and subjective. They are analyzed by classifying data as meaningful or contextual.

Quantitative Data Collection

Quantitative data can be collected using different methods. An “experiment method” involves data collection by way of a controlled observation, such as a survey, poll, or **interview**.

Experiment

An experiment is a data collection method in which a researcher changes **variables** to monitor the effect of its change on other variables. The manipulated variable is the independent variable. The other monitored variables are dependent variables. For example, a researcher is testing the effect of a dose of fertilizer on a number of species in a greenhouse. Researcher decides to examine fertilizer at different dosages of 10 milliliters (ml), 20 ml, and 30 ml. In this example, the fertilizer dosage is the independent variable and the number of species is the dependent variable (Albert, 2022).

Interview

This is a structured conversation, where an interviewer asks questions for the interviewee to answer.

Variable

A variable is a changeable element in a situation or experiment.

The benefit of using experiments is exploring casual relationships not identifiable in observational study. The experiment method is applied, for example, in sociology, psychology, medical research, and agriculture. It is, however, an expensive way of collecting data.

Observation

The data collected using the observational collection method is acquired by observing any relationships present over the course of a study. There are four types of observational data collection: **cohort**, case-control, cross-sectional, and ecological.

Cohort

This is when a group of subjects have a shared



characteristic.

Cohort

In the cohort method, the subjects with similar characteristics are studied over a set period of time. There should be enough data for this method to collect. Cohort studies usually take place over a long period of time.

Case-control

In the case-control method, case and control groups are created and observed. Changes are made to the case group, while the control group is left untouched. It is observable that the two groups differ, revealing the effects of the changes on the case group. For example, when the consumption of food product is to be observed, participants willing to repeat the consumption many times are the case group. Participants in the control group do not consume the product. The effect of consumption over a period is observed and studied.

Cross-sectional

The cross-sectional collection method determines prevalence. Consider doctors in a clinic who wish to examine the **prevalence** of heart attacks among a defined population. They evaluate the people in that population with their locations, social backgrounds, ethnicities, and ages.

Prevalence

This is how much of something is present in a population.

Ecological

Ecological data collection is perhaps the cheapest way to collect data, as it draws on data that already exists in different populations. Unlike other data collection methods, which study individuals, the ecological method looks at populations. Consider a researcher who is interested in identifying the differences between the child growth rate in the Middle East and South Korea. They would collect samples from databases, including data from the two populations, to perform comparison studies.

Collecting research data

Collecting data for research requires minimum standards, including data quality and traceability. The standards are set to make sure the research and analytics conclusions can be supported by data. In other words, if data cannot be traced such that the conclusions from research are connected to it, then it cannot be used (Zozus, 2017).

Qualitative Data Collection

Data collected qualitatively describe and conceptualize the findings of questionnaires, interviews, surveys, and polls. This method is popular for thesis works. To collect data using a survey, questions need to be designed properly. To develop the questions for survey, the goal of the research and the target group needs to be determined. The researcher runs the interview separately for each group in focus. For example, if the researcher has set out to discover the learning effect of a programming language on a group of young students and a group of older students, the first goal is to identify the groups: Identify the programming language and what is meant by “learning.” Does learning refer to mastering

the language or to know it at intermediary level? This might be important, since the respondents can have different understandings of what they consider learning. Likewise, each question should be relevant and have a purpose. Every question should have an expected answer that is drawn from prior research and which is used to help draw conclusions. In case there are multiple choice questions, the choice of the answers should also be adjusted (Grand Canyon University, 2023).

With face-to-face or online interviews, the researcher asks questions in order and in a timely manner. It is beneficial to have a protocol for having equal conduct for every respondent. This could include keeping the time in between the questions, logging the interview, and recording the sessions after having the consent from respondent. This is all to make sure the data are of good quality for compiling answers and devising a conclusion.

Public data

Besides the mentioned main data collection approaches, public data are publicly available on websites. Public data might have limitations in terms of reuse or redistribution. Because of the danger of copyright violations, it's recommended to always check the details of public data usage. However, they are available to explore, visualize, and communicate. Numerous worldwide public datasets offer data about governments, online media, traffic data, health and welfare, education, social media, and many other topics. Below is the list of some famous public data offering platforms together with their links.

- Google's dataset search provides datasets of any type (Google, n.d.).
- The Crime Data Explorer provides public crime and law enforcement data (Federal Bureau of Investigation, n.d.).
- FiveThirtyEight is a platform providing data and predictions about politics, sports, science and health, economics, and culture (FiveThirtyEight, n.d.).
- New York's trip record data includes information about passengers, drivers, vehicles, and businesses (New York City Taxi and Limousine Commission, n.d.).
- The U.S. government's public data platform includes data about government budgets to climate data (General Services Administration, n.d.).
- HuggingFace is a platform with an extensible library that allows easy access and sharing of datasets for natural language processing (NLP, HuggingFace, n.d.).
- The World Health Organization (WHO) provides a global health observatory data repository (WHO, n.d.).

2.2 Data Selection

Data selection precedes the operation of data collection. Selection of data involves a process. It is defined as the process of determining appropriate data type and source, as well as suitable tools to collect data.

The research methodology should be aligned with the collected data collected for it. Data collection methods should be carefully selected to ensure the validity and relativity of the results. When selecting data collection methods, there are factors to be considered. These factors are the research goal, scope of the study, type of data and sample size (Mwita, 2022). A major factor determining the research tool to be employed in data collection is the research goal. The goal is in line with the scope of the research (i.e., the extent to which the study will explore its subject matter), which is another important factor to consider in choosing data collection methods. "Sample size" refers to the number of people or respondents from whom the study is expecting to collect the data. The type of data this unit refers to is primary or secondary data. Primary data are collected for the first time from sources such as interviews or surveys. Secondary data are gathered from databases. Deciding on what tool to use to collect data may be done using common sense (Mwita, 2022).

The goal of sampling data is to select a source of data that represents the entire population of interest. Samples can be drawn from different populations of humans, animals, documents, species, and observations. If the sample dataset does not represent the population according to its true characteristics, it may introduce research bias and compromise the integrity of the data. Having bias in the sample dataset limits the ability of the research to generalize the results to the wider population. Therefore, to obtain a representative sample of data, there needs to be a suitable procedure. The sample set should be evaluated before its use. There are number of procedures to reduce the probability of having bias in a sample dataset, namely simple random sampling, stratified sampling, systematic sampling, and cluster sampling.

Simple Random Sampling

Simple random sampling is a subset of data selected from a larger dataset or population. The subset is selected randomly, where all data points have same probability to be drawn from the population, for example, 100 students to be selected randomly from the population of all students in a university (Thompson, 2012).

Stratified Sampling

Stratified sampling is sampling from subsets of a population. The sampling is thus conducted per subpopulation. The subgroups are also called strata based on the characteristics that they share (Zhao et al., 2019).

Systematic Sampling

Systematic sampling is a sampling method in which data points are selected from a population at a regular interval (e.g., every 10th person from a population of 5000). If the population is in random order, this sampling method would be similar to random sampling, but when there is a low risk of data manipulation this method is preferred. Roper et al. (2020) demonstrated that the result of research can differ based on the two methods of simple random selection and systematic sampling.

Cluster Sampling

This method is similar to stratified sampling. The population is divided into multiple groups or clusters. The groups are selected using simple random selection methods or systematic sampling method. This method is planned when there are **homogeneous groups** which are **heterogeneous** internally (Cadima et al., 2005).

The difference this method has with stratified sampling is that groups are randomly selected and include data points in all of them. In stratified sampling, some data points of all groups are selected and included in sampling.

Researchers selecting sample datasets often set inclusion criteria to filter the data points and to ensure they selected data can be used to answer the questions. Inclusion criteria is a key feature of the target population that is used by researchers to answer questions. Some typical inclusion criteria are demographic, and clinical characteristics. For instance, a researcher working on estimating appropriate drug dosages for patients might add the following inclusion criteria: All patients should have the defined disease and no other disease is present. They should have consumed a dose of prescribed medicine for some time based on the research question outlining the needs. Stern et al., (2014) discussed how the development of inclusion criteria can be done considering the review questions. When a review question is developed, reviewers should be aware of the population's characteristics and accordingly ensure the review is sufficiently general (Stern et al., 2014).



Homogeneous groups
groups of the same kind
Heterogeneous
when a group has diverse types

2.3 Data Cleansing

The purpose of data cleansing is to prepare the data being used for analysis, visualization, reporting, and predicting algorithms. Data cleaning is about removing unnecessary data, filling in the missing values, structuring data, and refining them prior to the analysis. Some tools perform data cleaning automatically prior to the analysis or visualization. An example is Tableau. However, efficient data cleaning has a high impact on the quality of the analysis and the results obtained.



In Python, data cleaning includes operations like dropping columns, renaming the variables, renaming the variables, resetting the index of the data, data framing, and more.

Let's start experimenting with some available open data that can be found in Python. Installing the `vega_datasets` will provide access to a number of datasets for employment. It's relevant that the Numpy and Pandas packages provide suitable tools for data cleaning.

To install the `vega_datasets` package, run the code below in your Python interpreter.

```
!pip install vega_datasets
```

To load the package, run `import vega_datasets`.

There are a number of datasets in this package. They are not in raw format, but can be used for the to show examples of data cleaning techniques in this book. Running the `local_data.list_datasets()` will show what datasets exist in the package. Let's pick the ~~United States (U.S.)~~ employment dataset.

US

```
local_data.us_employment()
```

After importing the dataset, looking at its properties (can be done via `type(dataset)` code) will show that the dataset is already in a data frame format.

Dropping Unnecessary Columns or Rows

To perform data cleaning techniques, such as removing an unnecessary variable, two packages, Numpy and Pandas, are needed. To load to packages, blow code can be used.

```
import numpy as np
import pandas as pd
```

Note that the syntax in Python interpreters is case sensitive. Perhaps the columns to be dropped need to be identified first. To get the list of dataset columns, use the `columns` attribute of the data frame. This dataset contains 24 columns as the result of `dataset.columns` code.

```
Index(['month', 'nonfarm', 'private', 'goods_producing', 'service_providing',
      'private_service_providing', 'mining_and_logging', 'construction',
      'manufacturing', 'durable_goods', 'nondurable_goods',
      'trade_transportation_utilities', 'wholesale_trade', 'retail_trade',
      'transportation_and_warehousing', 'utilities', 'information',
      'financial_activities', 'professional_and_business_services',
      'education_and_health_services', 'leisure_and_hospitality',
      'other_services', 'government', 'nonfarm_change'],
      dtype='object')
```

Assuming the column “other_services” is not needed, we proceed to remove this column from this dataset.

```
to_drop = ['other_services']
dataset.drop(columns = to_drop, inplace=True, axis=1)
```

First, the column name is stored as a variable serving as a parameter that is used in drop function. The `inplace` parameter can have two values, true or false. Setting the value to true is to force the drop of the column to occur in the same dataset rather than creating a copy of the data frame after dropping. The axis can have two values of 0 and 1. “1” refers to the dropping column and “0” refers to the dropping row.

How about dropping multiple columns? Selecting the columns to be dropped, the additional columns can be added by their names (see below code).

```
to_drop = ['other_services', 'private']
```

Dropping rows is done similarly, except that the axis value should be changed to 0. Note that the default value is zero if the axis parameter is not used. Dropping rows can be done by locating a number of rows from the index or indicating a range of row numbers. The example below shows how to remove a range of rows.

```
dataset = dataset.drop(labels=range(40, 45), axis=0)
```

To drop single rows, use the following:

```
dataset = dataset.drop(labels=range(40, 45), axis=0)
```

Renaming Columns

The names of the columns are not always easy to understand when importing the datasets. They can contain characters or numbers that need to be modified. To relabel the name of the columns the function `rename()`. A dictionary including the old and new names of the columns of interest is created first. Then, using the data frame's `rename` function, the columns are relabeled (Jafari, 2022).

```
new_names = {'Month': 'Year_Month_date',  
            'Government': 'nonfarm - private'}
```

```
dataset.rename(columns = new_names, inplace=True)
```

Change DataFrame Index

Some datasets have a column containing unique identifiers that can act as the index of the whole dataset. In this example, the `US_employment` dataset contains a “private” column, in which all records might be unique and could, therefore, be the index. The uniqueness of the column should be determined before changing the index. The `isunique()` function is used for this purpose (Jafari, 2022).

```
dataset['private'].is_unique
```

This function returns two values, true or false. In this case, the function returned “true,” indicating that the private column contains all unique values. Therefore, it can be the index for this dataset. Of course, the column that is set to be the index should serve as desired (e.g., there is no other column serving as a better index).

```
dataset = dataset.set_index('private')
```

To ensure the code is in effect, we can check the first rows of the dataset using the `head()` function.

```
dataset.head()
```

The below figure shows part of the dataset loaded as result of running the above code:

Table 2: Change Index in DataFrame

	month	non-farm	goods_producing	service_providing	private_service_providing	mining_and_logging
private						
113603	2006-01-01	135450	22467	112983	91136	656
113884	2006-02-01	135762	22535	113227	91349	662
114156	2006-03-01	136059	22572	113487	91584	669
114308	2006-04-01	136227	22631	113596	913677	679
114332	2006-05-01	136258	22597	113661	91735	681

Source: Somayeh Aghanavasi (2024).

Cleaning data does not stop here. Extra characters may need to be removed from cells. It's also possible that the format on rows and columns will need to be changed. In fact, there are many operations to make sure data are clean and usable for further processes. Other possible cleaning could include writing a custom function to assess every cell and delete an extra character, which adds clarity to the data in cells. Depending on the status of the data, Python provides a number of libraries that are useful for data cleaning, namely, Numpy, Pandas, Matplotlib, Datacleaner, Dora, Seaborn, Arrow, and Scrubadub.



2.4 Sparse Data and Missing Values

Handling Sparse Data

A dataset or DataFrame can have sparse and dense data. Sparse data are the data containing mostly unused data or values that do not contain valuable information. An example is when a dataset contains mostly zero values. In contrast to sparse datasets, dense datasets contain nonzero values. The existence of sparse data in the dataset affects the performance of the analysis and the obtained results. In fact, sparse data can cause problems with adding to the space and time complexity. Sparsity can occur due to not having data to be recorded or stored in some cells in the dataset. To handle the sparse dataset, the zero values should be ignored. Only nonzero values should be extracted and used in a structural approach.

To identify the extent of sparsity in a dataset, the number of zero data points can be divided into the total amount of data.

$$\text{Sparsity} = \frac{\text{count zero data points}}{\text{total amount of data}}$$

Let's assume a small dataset and an array containing some zero element in the below example.

```
Data_array = [0,0,0,2,0,0,4,1,0,0,0,6,0,1,0,0,0,0,1,0,9,0,0,4,0,7,0,0]
```

The array contains 28 data points, of which 19 are zero values. The amount of sparsity is, therefore, 67 percent. To implement this in Python, we need to import a few packages, namely, the “array” and “count_nonzero” packages.

```
from numpy import array
from numpy import count_nonzero
# create a sparse matrix
Sparse_Matrix = array([[0, 0, 1, 0, 0, 1, 0, 0],
[0, 0, 2, 0, 1, 0, 0, 1], [1, 2, 0, 0, 0, 2, 0, 0]])
print(Sparse_Matrix)
# calculate sparsity
Sparsity = 1.0 - count_nonzero(Sparse_Matrix) / Sparse_Matrix.size
print(Sparsity)
```

In the example above, the two packages are imported. This time, the sparse dataset is a matrix. The lines starting with # are the commented lines, not a code line. The matrix rows are separated by a comma between each row defined using brackets. To calculate the sparsity, we use the count_nonzero() function, which counts the elements that are not zero. Subtracting the nonzero datapoints from 1 gives the number of zero elements. Dividing this result by size of the matrix provides the sparsity percentage. At the end, the print() function shows the sparsity measure.

There are techniques in Python to convert a sparse dataset to a dense dataset eliminating the zero values. For this purpose, the csr_matrix() function from Numpy Array can be used. Both the numpy and SciPy packages need to be installed and loaded. The two modules of array and csr_matrix need to be imported from the two packages.

```
from numpy import array
from SciPy.sparse import csr_matrix
# create a matrix as a dataset
Sparse_Matrix = array([[1, 0, 0, 0, 0, 1, 0, 0], [0, 1, 1, 0, 2, 0, 0, 1],
[2, 0, 0, 0, 0, 2, 0, 0]])
print('Sparse Dataset:')
print(Sparse_Matrix, '\n')
# convert to sparse matrix using CSR method to a dense matrix
```



```
Dense_Matrix = csr_matrix(Sparse_Matrix)
print('Dense Dataset:')
print(Dense_Matrix, '\n')
```

Running the above code, the sparse matrix is created first. Then, the content of the matrix is printed. Please note `\n` is used to print a new line. The `csr_matrix()` function uses the sparse matrix to create a dense matrix. The dense matrix containing only the nonzero items is printed out. Result is shown in below figure.

Figure 8: Creating Sparse and Dense Matrix

```
Sparse Dataset:
[[1 0 0 0 0 1 0 0]
 [0 1 1 0 2 0 0 1]
 [2 0 0 0 0 2 0 0]]

Dense Dataset:
(0, 0)      1
(0, 5)      1
(1, 1)      1
(1, 2)      1
(1, 4)      2
(1, 7)      1
(2, 0)      2
(2, 5)      2
```

Source: Somayeh Aghanavesi (2024).

Handling Missing Values

Missing values can refer to not a number (NaN) in numeric arrays, none (or null) in object arrays, not a time (NaT; a missing DateTime value), and not available (NA; generally a missing data point). In Python using Pandas package, there is a function `isna()` determining if the content of the given input is a NaN value or not by providing the boolean values of true or false response. Pandas treat NaN and none interchangeably as missing values. In the below example the function `isna()` is used taking a string as a non-null value. The result of this function is a Boolean value, which is false.

```
>>> pd.isna('not null value/ string')
False
```

`isnull()` is another function that can be used to determine the missing values in a dataset. This function returns a dataframe of Boolean values containing true for NaN values. In the example below, a dataset is created, converted to a data frame, and evaluated whether it contains a missing value. The entries where the values are missing are filled with `np.nan`.

```

import pandas as pd
import numpy as np

# dictionary of lists
list = {'Col1':[12.3, 9.4, 80.5, np.nan, 10.1],
        'Col2': [np.nan, 4.5, 15.9, 63.6, np.nan],
        'Col3':[np.nan, np.nan, 82.5, 80, 98] }

# creating a dataframe from list
DF = pd.DataFrame(dict)

# using isnull() function
DF.isnull()

```

Table 3: isnull() Function Output

	Col 1	Col 2	Col 3
0	False	True	True
1	False	False	True
2	False	False	False
3	True	False	False
4	False	True	False

Source: Somayeh Aghanavesi (2024).

To replace the null/missing values the function `fillna()` can be used alternatively. In a similar approach, using this function instead of `isnull()` will provide the below results. Using this function, the non-null values remain the same, but the missing values are filled with a 0. This is when the function is used with an input of 0. This means if the missing values needs to be filled with another value the function will take the desired input (i.e., `fillna(1)` to fill with value 1).

Table 4: fillna() Function Output

	Col 1	Col 2	Col 3
0	12.3	0.0	0.0
1	9.4	4.5	0.0
2	80.5	15.9	82.5
3	0.0	63.6	80.0
4	10.1	0.0	98.0

Source: Somayeh Aghanavesi (2024).



In a similar strategy using `fillna()` function, the values can be filled with a previous value in a column. For that purpose, the input of the function would be the method to be used for filling out the missing values, known as `fillna(method = 'bfill')`.

`Replace()` and `interpolate()` are the two other functions in Python that can be used to fill in the missing data. Below are two examples of how to employ them.

```
DF.replace(to_replace = np.nan, value = 0)
DF.interpolate(method = 'linear', limit_direction = 'forward')
```

2.5 Data Consistency

Data take a journey from databases (or from where it's collected/cleaned) to end users', processes, and end devices. However, throughout this journey the data format should not change. This is how data remain consistent. Data consistency is needed to maintain the accuracy and usability of data. It can directly affect the success of the business, whereas inconsistent data can lead to misinformed business decisions. Comparing data integrity and data consistency, integrity refers to correctness, accuracy, and quality, while consistency means data are correct in relation to other data. Data consistency ensures the users have the same view of the data, including the changes made by one or others.

There are three types of data consistency: (1) point in time consistency, (2) transaction consistency, and (3) application consistency. Point in time consistency refers to the uniformity of data in time. For example, in the event of a power failure, point in time consistency ensures data are restored to what they were at the moment of failure. Transaction consistency is often used in database systems. It refers to finding incomplete transactions to roll back the data in an event the transaction is not completed. Application consistency refers to promoting uniform formats between the applications. It works with transaction consistency.

Fuzzy
when uncertain information is given

In some areas, where the measurement is collecting **fuzzy** data using a scale of “low,” “high,” “very high,” and “medium,” it is important to measure the consistency of what's collected. Measuring the consistency of collected data, including fuzzy data, will make sure it collects the correct construct as it supposed to collect or measure. It's reliable if it measures the same thing every time. It is also internally consistent. The internal consistency of the collected data or a measure can be calculated by Cronbach's Alpha method (Adamson & Prion, 2013). Cronbach's Alpha is calculated using the following formula (DeVellis, 2005):

$$\rho_T = \frac{K^2 \bar{\sigma}_{ij}}{\sigma_x^2}$$

where ρ_T is the tau-equivalent reliability, ρ_T is the number of items, $\bar{\sigma}_{ij}$ is the covariance between X_i and X_j , and σ_x^2 is the item variances and inter-item covariances. Cronbach's alpha ranges between 0 and 1. An acceptable result is equal to or higher than 0.7. Simi-

larly, intraclass correlation coefficient is a method to calculate the consistency (Bartko, 1966). The Python code below shows a function created to calculate Cronbach's alpha according to the formula.

```
def CronbachAlpha(scores):  
    scores = np.asarray(scores)  
    vars = scores.var(axis=1, ddof=1)  
    tscores = scores.sum(axis=0)  
    nitems = len(scores)  
  
    return nitems / (nitems-1.) * (1 - vars.sum() / tscores.var(ddof=1))
```



SUMMARY

Data collection plays an important role in ensuring the integrity and quality of the knowledge extracted from data. Data sufficiency is determined by statistical hypothesis testing methods (p-values) to measure how much data would be needed to provide proper insights. Qualitative and quantitative data collection are the two main approaches of data collection. The quantitative approach provides numeric and countable data, whereas qualitative data are extracted from narratives through interviews and surveys.

Data selection occurs prior to data gathering. The aim of sample data is to possess a dataset representing its population and the population's characteristics. The knowledge obtained from conducting the analysis on sample data should be generalized to its population. Data selection is a process of determining what data to collect from what source and the tools should be used to do so. Some common methods to select data appropriately include simple random sampling, stratified sampling, systematic sampling, and cluster sampling.

Collected data from different sources can contain noise. Raw data are not easy to read or understand. They can, therefore, require cleaning prior to their use in analysis, visualization, or modeling. Data cleaning includes techniques in refining the data and making a clear structure of them for more clarity. If dirty data are used in later stages, it affects the inferences made from data and can lead to incorrect insights or decisions. Cleaning data includes removing extra characters from data, renaming the columns, and removing unrelated rows.

Some datasets include a significant amount of unused data, which makes them ineffective for further processes. In the same way, there are datasets containing missing values that makes it difficult to use it for extraction of useful information. Therefore, the application of techniques to handle the sparse and missing values are important. Python

provides some common operations for reducing the number of zero values of a dataset and converting it to a dense dataset. Missing values can be identified and replaced with the most desired values, such as zero, or closest value. Interpolation can be used as a function to fill the missing data in the datasets.

Data consistency refers to the correctness of data in relation to other data. It is needed to maintain the accuracy and usability of data. It can directly affect the success of the business, whereas inconsistent data can lead to misinformed business decisions. Three main types of data consistency are point in time consistency, transaction consistency, and application consistency. Calculation of consistency can be done by two methods, Cronbach's alpha, and intra correlation coefficient.

UNIT 3

DATA RETRIEVAL STRATEGIES

STUDY GOALS

On completion of this unit, you will be able to ...

- describe the approach for query driven data retrieval.
- describe what a data stream is and what its sampling approaches are.
- understand the challenges of large-scale data and the methods for sampling such data.
- describe the purpose of process mining.
- understand textual data and information extraction methods.

3. DATA RETRIEVAL STRATEGIES

Introduction

Data resides in different locations and with different forms. It can be formed of tabular data residing in databases or data stream form. It also comes in different sizes. These characteristics affect the adopted strategies for retrieving data.

In this unit, the different strategies to retrieve the data with various characteristics are explored. Query-driven data mining and retrieving data from databases using a query writing approach are major topics. Queries are used in web searches in search engines, as well as when data are requested from databases. But it must follow the standard specified, according to what data and what characteristics are to be requested. In the first part we discuss what query driven data retrieval is and how it is done using Structured Query Language (SQL) language.

In contrast to structured data residing in databases, a data stream is a volatile, continuous, and ordered sequence of instances. It can be read only using limited computing and storage capabilities. Together with these characters, as well as its infinity and nonstationary attributes, it requires special approaches to be retrieved. Stream sampling and filtering the samples are two main approaches for dealing with data. The approximation of the unique objects in a data stream can be done using the Flajolet-Martin explorer.

Big data mining faces challenges with data retrieval due to their volume, variety, velocity, and veracity. Therefore, a correct, accurate, and generalized data sampling is required. A sample set that can reflect the characteristics of the whole population is needed. The three approaches provide insight into retrieving data from large scales. Model fusion is an approach to collect data and integrate it from different data levels. To understand better data management, sparse, uncertain, and incomplete data mining will be discussed. Complex and dynamic data mining will also be showcased.

As part of this unit, which is mainly about strategies to retrieve data, process mining performs analysis based on event logs created from information and interactions in a business. Its goal is to gain a view into the business-level processes. It analyzes the operational processes and enables the business to gain a holistic view of their processes. Having this view, the business can identify inefficient points and improvement opportunities. Process mining helps businesses to uncover the root of operational challenges.

When it comes to textual data as un/semi-structured data, special techniques are needed to extract meanings and information. Information extraction has a main role in (NLP). One technique is to use a template with a specified list of slots filled with substrings extracted from text documents, websites, and databases. Python provides a useful library called spaCy, which is used for this purpose as well. Part of speech (POS) and morphology are part of spaCy library to discover the words, their position, and their root, as part of information extraction from textual data.

3.1 Query Driven

A query is an extracted word from a question. In computing, domain query is used to retrieve data or information. Queries are used to extract and structure data. There are different types of queries, such as search queries and database queries. A search query places keywords in a search engine and retrieves information about those keywords. A database query retrieves data records and statistical summaries of data from databases. They can be used for modeling, structuring, or updating the data in a structure. Queries might run in many cases without the direct action of an operator, like in dynamic websites. Most of the queries in dynamic websites run automatically when there is a new page visit. There are background functions in running applications that perform the queries based on the input. The basis of almost all queries is to receive an answer (Dam & Fritchey, 2009).

The data mining query language (DMQL) is based on SQL, which is also a relational query language. DMQL can support interactive data mining and provides commands to specify the primitives. SQL works with databases and data warehouses. The elements in SQL include keywords, identifiers, clauses, expressions, predicates, queries, and statements (Coding, 2020).

A simple example of a SQL query is in the below statement.

```
SELECT * FROM database_name;
```



NOTE

The statements in SQL end with a semicolon. All SQL statements start with a keyword. The commands are case sensitive. The table and database names must match the existing and original sources.

Every SQL query starts with a keyword. Keywords and their purposes are listed below:

- SELECT: to select from a database/table
- UPDATE: to update the content of a table/database
- CREATE TABLE: to create a table or database
- DROP TABLE: to drop a table from a database
- ALTER TABLE: to modify the content(value/attribute) of a table
- DELETE: to delete data from table/database

SQL recognizes the following data types: integers, character, variable character, numerical values, floating values, date, time, and time stamps. The below table shows an example of each data type in SQL.



Table 5: **Data Types in SQL**

SQL data type	Syntax in SQL	Example in SQL
Integer	Int	Create table 0_01(Reg_No Int);
Character	Char(n), n>0	Create table 0_01(Name char(10));
Variable character _ type 1	Varchar(n)	Create table 0_01 (ID Varchar(10));
Variable character _ type 2	Varchar2(n)	Update table 0_01 (address Varchar2(10));
Numeric	Numeric(P,S) P is precision. S is the scale, the number of digits to the right of the decimal point.	Height(5,2)
Float	Float(n) with six decimal places for precision.	Float(22)
Date	YYYY-MM-DD	2022-Oct-25
Time	HH:MI:SS	13:13:23
Time Stamp	YYYY-MM-DD HH:MI:SS	2022-Oct-25 13:13:23

Source: Somayeh Aghanavesi (2024).

Operators in SQL are +, -, *, /, and %. For comparison, the operators <, >, =, <=, =>, and <> are used. To compare the expression of binary and bitwise, the operators &, ^, and | are used to perform AND, XOR, and OR. The logical operators, including LIKE, AND, IN, BETWEEN, OR, and NOT, are to be used between the queries or join conditions, or to extract specific values.

Below are examples of SQL statements using comparison, arithmetic, logical, and bitwise operators.

- arithmetic: `Select 8 + 10;`
- comparison: `Select * from table 0_01 where age > 18;`
- bitwise: `Select 3 | 5;`
- logical: `Select * from table 0_01 where age Between 18 AND 40;`

SQL is important when working with data mining, extraction, analysis, and visualization. Python provides the possibility to connect to databases directly within the compiler and query the data using SQL statements right in the Python integrate development environment (IDE). Querying data using this approach facilitates having the process of extracting data and processing them in one place rather than fetching them from other environments, such as the Microsoft SQL server, and transferring it to Python IDE for processing. For this purpose, some libraries are needed, such as Pandas and `mysql-connector-Python`.

To connect to the database of interest, use the following code.

```
Connection = mysql.connector.connect(host=host_name, user=user_name,
passwd=user_password)
```

After a successful connection, a function can be defined to be executed every time a query needs to be executed. The function can be defined as seen below.

```
def excute_query(connection, query):
    cursor = connection.cursor()
    try:
        cursor.execute(query)
        print("The query is executed")
    except Error as err:print(f"Error: the query did not execute
successfully ")
```

The function takes the two input values of successful connection and the defined query. The query can be defined as below, with the function to be called using the name of the query and the connection that was made.

```
Query = "Select * From Table 0_01 Where ID <> 0"
excute_query(connection, query)
```

The query is calling for all records in table 0_01 where their identification (ID) is not equal to zero. The function takes in the two input parameters of connection, and the query runs and returns the results. The results can be saved in a new variable to be processed further in the Python IDE.

3.2 Mining Data Streams

Prior to elaborating on the data stream mining, let's investigate what the data stream is. The data stream is the continuous and ordered sequence of instances. It can be read only using limited computing and storage capabilities. Examples of data streams are computer network packet traffic, conversations using phone, transactions, and sensor data. Since stream data are **volatile**, they need to be processed before they disappear. The data arrive at high speed and actively storing the data is costly. In other words, stream data are infinite and **non-stationary**, and their management is important (Matthes & Lutz, 2021).

There are three main challenges with data streams in data mining: sampling data in a stream, filtering the streams, and counting distinct elements in a stream (Matthes & Lutz, 2021). Sampling from data stream is the process of collecting data that are representative of the elements of a data stream. The sample is usually smaller than the entire stream but contains the its important characteristics. Therefore, it can be used to estimate many important aggregates on the data stream (Lahiri & Tirthapura, 2009).

Volatile
Once the data is mined or analyzed, it is discarded.
Non-stationary
when the mean, variance, and covariance change over time

Data Stream Sampling

There are different ways to sample data streams. The simplest method is the sliding window. In this method, a queue of samples from the stream is taken. The queue has the size of n . There is also a subsampling factor of k , which is supposed to be larger than 1. The window slides across the data stream based on a specified interval. For example, a sliding window with a length of 20 seconds and a sliding interval of eight seconds contains queues that arrive within a 20 second window. In another method called unbiased reservoir sampling, there is a container to fill the data with the first n points from the stream. At the time point $t > n$ a randomly selected item is replaced which has the acceptance probability of n/t . This probability acceptance rate is taken from the original formula designed to calculate the items probability when there are n items. This method leads to a container in which the elements have a similar probability of being selected.

In the biased reservoir sampling method, the probability of the items residing in the container is updated. It decreases the data points mined from a longer history. It means the data points with sooner history have higher probability of residing in the container. A fourth method is called histogram method to sample data from streams. In this method, a histogram is made of the observations of the data stream. The data points from those observations are sorted into containers (Matthes & Lutz, 2021).

Filtering the Streams

Data stream filtering is used to remove the data or observations that contain errors or are otherwise not desired to be included. For this purpose, a specific criterion needs to be defined and the data filtered accordingly. The data stream matching the criterion is to be kept for further processes and rest discarded. A method performing this filtering is bloom filtering. Data points that are selected are hashed into buckets to form bits which later will be set to 1. Hash buckets are used to divide out data items for sorting. Its purpose is to loosen the links in a list to facilitate the search for specific data items to be found within a shorter amount of time. An example of usefulness of data stream filtering is when web-based applications use filtering to divide out the list of user choices so that it becomes manageable and usable in further processes.

Count Distinct Elements in a Stream

Assume a web-based business such as Amazon is about to tally the number of customers entering the web site. How many distinct products are purchased in a region? For this, the Flajolet–Martin (FM) algorithm, created by **Philippe Flajolet**, is used to approximate the number of unique objects in a stream (Flajolet & Martin, 1985). Assuming the stream contains an n element, out of which m are unique, the FM algorithm runs in $O(n)$ time and $O(\log(m))$ memory. The FM method follows four main steps for estimating the distinct number of elements. The steps are as follows:

1. Select a hash function $h(x)$ to map each element from the stream to a string with at least $\log_2 n$ bits.
2. For each element x , the length of the zero stream in the function $h(x)$, the hash function that is mapping the input identifiers.

Philippe Flajolet
the owner of this method,
a French computer scientist
and senior research scientist
in the field of computational
complexity and algorithms

3. Calculate R as maximum value in $r(x)$ set
4. Calculate 2^R for first hash function.
5. Calculate the median value between the results for all hash functions.

To put the steps in practice with an example, let's assume the following: x is a stream set of [1, 3, 2, 1, 2, 3, 4, 3, 1, 2, 3, 1]

$$h(x) = (6x + 1) \text{ mod } 5$$

$$|b| = 5 \text{ as the number of bits.}$$

The $h(x)$ will become the set of [2, 4, 3, 2, 3, 4, 0, 4, 2, 3, 4, 2] after taking the x items into account. Then, the binary value of the elements in $h(x)$ is [00010, 00100, 00011, 00010, 00011, 00100, 00000, 00100, 00010, 00011, 00100, 00010] and the $r(a)$ is the number of zeros after the first appearance of 1. In case there is a one in the binary number, all zeros to be counted. $R(a) = [1, 2, 0, 1, 0, 2, 5, 2, 1, 0, 2, 1]$.

$R = \max(r(a)) = 5$ and $2^5 = 32$ is the first value. Repeating the process for all hash functions and calculating the median value from the resulting values, the yields the distinct item count of the data stream.

3.3 Large-Scale Data Mining

Mining big data requires approaches and techniques to retrieve relevant and demanded data from large datasets. The approaches enable acquiring useful information from databases that include huge amounts of data. In this section, we will explore the challenges of collecting data from large-scale data so that they can represent the characteristics of the population. Imagine a large-scale dataset like a large dinosaur, and several people with blindfolds around their eyes are trying to understand the dinosaur's size or shape. One person is located at the head, another is on top, and another is at its tail. They try to get a sense of the dinosaur by touching the point they are at. The problem is that each person has a picture limited to the area where they're located. To further the complexity, the dinosaur is growing, and its figure is changing size. Moreover, each person speaks a different language when describing what they see from their subjective perspective. The exchange of information is relative and heterogeneous. Discovering how the dinosaur looks in shape and size is difficult with respect to what each person describes from their point of view, language, and even their concerns about security, if one is not willing to describe everything.

In a similar manner, big data mining faces challenges due to their volume, variety, velocity, and veracity. On the other hand, data can be distributed. This means they are not available in one place and it can be a challenge to bring data from heterogeneous sources to a centralized repository. In addition, when the data come in large quantities, owning and maintaining the hardware, software, storage, and servers becomes costly (Gandomi & Haider, 2015). Let's explore three approaches for tackling big data mining challenges.

Model Fusion for Multiple Data Sources

We noted that aggregation of distributed data into centralized sites is not encouraged due to its high cost and privacy issues. Data mining can be carried out at distributed sites, but it needs to remain concrete, correct, and consistent to enable the right decisions to be made from data analysis. Or, like the previous metaphor with the dinosaur, the information given by each person needs to be compiled to achieve a big picture of the whole figure. In such circumstances, information sharing and fusion procedures need to be enabled for a big data mining system to make sure that all distributed information and data can work together for achieving a universal picture. For this purpose, universal mining with two main steps can be featured: The first is local mining, and the second is the universal correlation to be processed at the levels of data, model, and knowledge.

At the data level (the lowest level), local sites calculate data statistics from local sources and exchange the statistics between the local sites to get a global view concerning data distribution. At the model level, which is the middle layer, each site carries out local mining on the local data sources to identify local patterns. The patterns are exchanged between multiple sources, forming a new aggregated pattern that represents a new global pattern. The process so far has been performed on two data and model levels. Models are generated from different data sources to determine the relevance of the data sources. At the third level, the knowledge level, the correlation between models is built to reveal the relevance between them. This supports the formation of accurate descriptions. This process is also called local learning (Wu et al., 2013).

Sparse, Uncertain, and Incomplete Data Mining

Only a few data points from sparse data can be included for a reliable conclusion; this is due to the high dimensionality presented by sparse data. With a high dimension of data, finding trends becomes a challenge. There are some dimensions reduction approaches to reduce the data sparsity, such as some general unsupervised methods or principal component analysis (Artoni et al., 2018).

Uncertain data contain noise or are biased and not deterministic. Uncertainty can occur with the nature of this data, or it can relate to the technology or application collecting the data. For example, some technologies for collecting Global Positioning System (GPS) data can include some error (one-meter error per location). Moreover, in some applications recording private data, users might include some random errors intentionally to protect their privacy. For example, when some users provide information about their salaries, they provide the range of the salary instead of the exact number. So, in such cases, the error is included in the data from distribution, not in a single value. Therefore, to have a solution, the whole distribution needs to be considered. There are some error-aware data mining solutions, like the naïve Bayes model or decision trees, which use the mean and the variance of the distribution considering every single value

Incomplete datasets include missing values. The missing values require imputation methods for replacement. These methods use mean, median, or frequency of the data.

Complex and Dynamic Data Mining

In contrast to tabular data, cloud servers, social networks, communication networks, time series sequence data, and biologic sequences are considered complex data. Although it is difficult for learning systems to capture their structure, they can provide exciting information that simple data are not capable of. One example is detecting some events from analyzing scratched data from social media. Another is when the data are based on people's queries in search engines to identify the outbreak of some diseases. Mining complex and dynamic data requires preprocessing (Wu et al., 2013).

The news on X, pictures in Flickr, and the news on the internet are good examples of intrinsic semantic associations in the data. Mining this complex association from text, image, or video can significantly help improve the performance of the application. In the context of big data and the complex relationship networks, there are relationships involving big data from Facebook, X and LinkedIn (Wu et al., 2013). With the case of time series complex data, dimension reduction approaches are used to transform time series data into numerical data. This is to have subsequent matching data that are like the original and represents the time series data.

Biological sequences are long sequences of nucleotides. The purpose of mining such sequences is to find the features of human deoxyribonucleic acid (DNA). To compare the alignment of the biological sequences, the data need to be mined. For this purpose, the degree of similarity between the nucleotide sequences is measured. The degree of similarity can be measured using different methods, such as dynamic time warping. This method aligns the signal of the time series and calculates the distance between the two signals and provides the degree of similarity between the two-time series (Skutkova et al., 2013).

3.4 Process Mining

Process mining is a technology used to gain deeper views into business level processes. It analyses the operational processes and enables the business to gain a holistic perspective concerning their processes. With this view, business can identify inefficient points and opportunities for improvement. Process mining helps businesses to uncover the root of operational challenges. In fact, analysis is performed on processes that are based on event logs extracted from companies' databases, and customer management system, or business management software. Perhaps one of the differences between data mining and process mining is that data mining looks for the data in the resources while process mining is more interesting about how the data are created based on the processes.

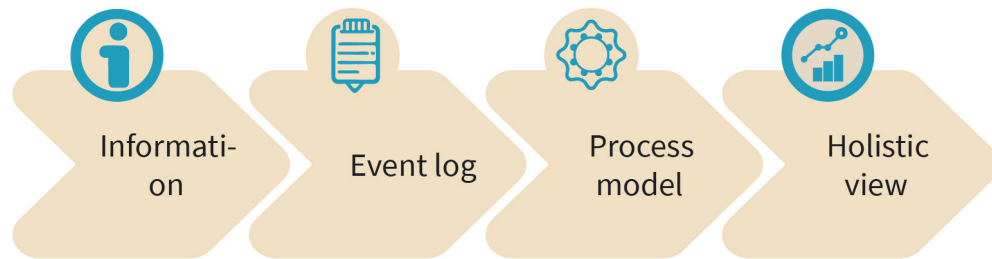
Process mining is becoming trendy. Microsoft has made a move towards this technology, and the market is expected to grow to \$11 billion by 2028 (Fagan, 2022). The adoption of process mining is increasing in businesses (Dilmegani, 2022).

Early research into process mining lead by Wil van der Aalst (2011) concluded that "process mining provides an important bridge between data mining and business process modeling and analysis" (p. 172). The article also highlighted business activity monitoring and

complex event processing. Business activity monitoring refers to technologies enabling the real-time monitoring of business processes, whereas complex event processing refers to technologies processing a large number of events utilizing them to monitor, steer, and optimize the business in real-time (van der Aalst et al., 2011).

The below figure gives a general overview of how process mining works.

Figure 9: Process Mining



Source: Somayeh Aghanavesi (2024).

The activities or the interactions with information technology (IT) systems are made and create digital records. It's like receiving an order, entering information to create a case, or submitting complementary documentations. The digital records are transformed into event logs. They include the timestamp, identification numbers, and records of the activity. Some process mining software creates a visual map of the activities. This is to facilitate understanding the process. With the process analytics, some methods such as key performance indicators (KPIs). KPI development can be adopted to uncover the potential improvement areas. Some process analytics use machine learning to detect hidden patterns and dependencies.

Before process mining, there were only interviews, which could be held in businesses to help understanding and analyzing the performance of the processes. This was slow, time consuming, high-effort, and subject to error. The emergence of process mining allows businesses to leverage automation and identify what happens with processes and their performances in the real world without having an accurate view of businesses processes. Process mining provides accurate and fast insight into business processes.

The adoption of process mining is significantly increasing. It is estimated that process enhancement applications will reach 42% and exceed process discovery in 2022 (Fagan, 2022). Van der Aalst identified three main areas in process mining: process discovery, conformance, and enhancement (van der Aalst et al., 2011). Process discovery takes up to 38% of the basic process mining (Fagan, 2022). Process discovery uses the event logs including historical information to discover how the business processes work. Then, adoption of conformance as part of the process mining identifies how the event logs reflect what happens in real world. Conformance takes up to 34% of the basic process mining (Fagan, 2022). With what's learned during the process discovery and conformance, enhancement identifies the improvements that can be made and the inefficiencies that can be eliminated in business. Enhancement takes up to 34% of basic process mining (Fagan, 2022).

However, process mining has some limitations. It runs on historical data and the performances conducted in the past. It is not enabled to monitor the processes in real-time or on an ongoing basis for possible anomaly alerting. In some cases, it might fall short in performance with complex scenarios with large number of variations.

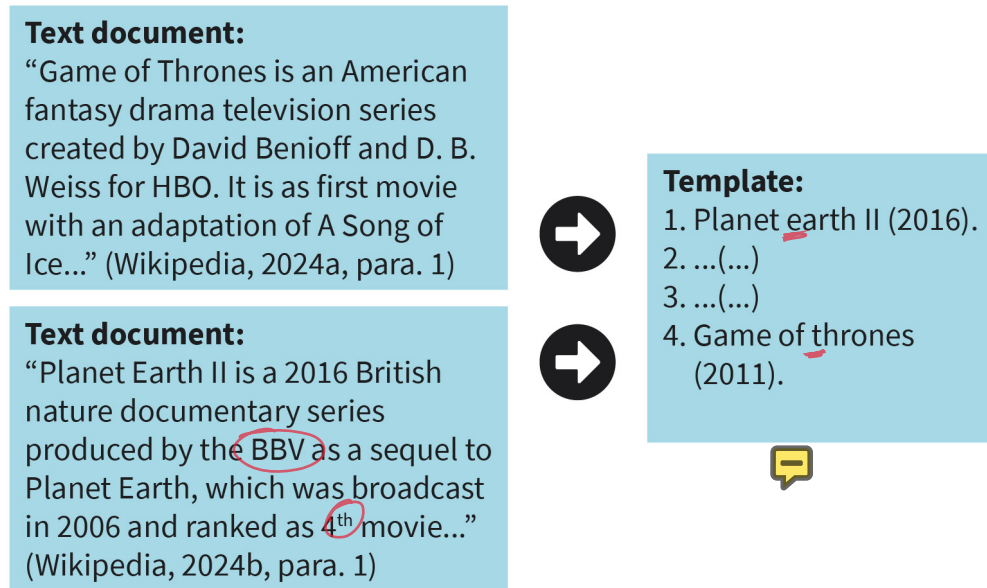
3.5 Information Extraction

Information extraction is a process to extract information from semi-structured or unstructured data, such as textual data. Extracted information is used to classify and store information in the databases (Wilks, 1997).

Data mining, known as knowledge discovery, assumes that the information is mined in the same form as they are in the databases. Data mining assumes that the data are already in the form of a relational database. However, there are differences between data mining and information extraction. Information extraction is about locating specific pieces of data from semi or unstructured sources, such as natural language documents within structured data, and extracting information from them. Information extraction has a primary role in natural language processing and information retrieval.

There is usually a template that specifies a list of slots to be filled with substrings that are extracted from text documents, websites, and databases. For example, consider a group of text documents that list television (TV) shows or movies based on the ranking of TV programs their popularity details. The template would have slots corresponding to the name of the program, its rank, producer, and the year it was released. For the information extraction system, it would be enough to understand the documents to find the corresponding data for filling the slots in the template, based on the input criteria given to system. The below figure shows an outline of how information is extracted from the documents into the template.

Figure 10: Information Extraction From Documents



Source: Somayeh Aghanavesi (2024), based on Wikipedia (2024a; 2024b).

Information extraction techniques can be applied on semi-structured data as well. This refers to an information extraction system restoring structure throughout documentation. For example, this could include extracting a table from a document, the information from that table to be extracted in a structured format, or the commands from the content of the document (Milosevic et al., 2019).

Knowledge base
This means to retrieve information from complex structure or unstructured information.

Extracting information from a logical perspective refers to **knowledge base**, which is about filling a database with facts from the documents. For example, going through information extracted from sentences of a document, information extraction may find the type of numbers or numerical expressions that have been used or the place names or some specific words that are special to that document. It can also refer to identifying similar words between documents or relationships between the entities (Nguyen & Verspoor, 2019). Let’s assume that one document can include the place of the employees of an organization. Another document may discuss the type of work those employees do at that organization, while a third might include the history of those people. By extracting the relationship between documents, information such as place, history, and the role of the employees is linked and might be found useful for further processing.

Dudas et al. (2009) performed information extraction from simulated data by applying data mining techniques. The purpose was to optimize manufacturing processes. They provided support for operators learning how various process parameters affect the optimization criteria (Dudas et al., 2009).

In Python, spaCy is a popular library that supports identifying and finding words of interest and the position of speech (POS) tagging. There are eight different POSs in English, namely noun, pronoun, verb, adjective, adverb, preposition, intersection, and conjunction. With POS, the specific function that each word in the sentence has in relation to the

meaning is determined. For example, the words “row” in the sentences “they will row the boat” and “they sat in the front row” have different meanings. It is a verb in the first sentence, and it is a noun in the second sentence. This shows how important it is to identify the meaning of the words in a sentence. Using the spaCy library in Python, the code below shows a simple example with identifying the noun in a sentence. See the following example, where the attribute `.pos_` is identified and extracted (Vasiliev, 2020).

```
for token in doc:
    # POS token
    if token.pos_=='NOUN':
        # print
        print(token.text)
```

Now, the attribute can be changed to identify a verb, adverb, or other positions of the words in a sentence. However, the POS from this library does not support the detection of a subject or an object. For this, we need to find how the words relate to each other, also known as finding the dependency. We can import the `displacy` package from the spaCy library. Please note the below example.

```
from spacy import displacy
displacy.render(document, style='dep', jupyter=True)
```

The code outputs a dependency graph representing the relations between the words from the sentence given as input (Vasiliev, 2020).

In addition, morphology is another feature provided by the spaCy library. Inflectional morphology is a process to modify the root form of a word by adding prefixes or suffixes to specify its grammatical function (Sylak-Glassman et al., 2015). The process does not change the word’s POS. The root form is called “lemma,” and the modified or combined parts are called “inflected.” These two are the morphological features to create a surface form. For example, in the sentence “I was reading the paper,” morphology captures “reading” as the surface, “read” as lemma, and POS as a verb.



SUMMARY

Data retrieval strategies include the main strategies to retrieve data considering different technologies, size of the data, and the logic behind extraction of some information. SQL, as a query driven approach, is an important tool that is used widely for mining structured data from databases. It is a query language with simple structure enabling the extraction, creation, modification, and saving of data.

Data streams have different structures. They can be volatile or continuous, and they are the ordered sequence of instances. They are also infinite and non-stationary. Stream sampling and filtering the samples are

two main approaches for dealing with streamed data. When it comes to large-scale data, it's important to collect in a way that reflects the population's characteristics. This should also take into account volume, variety, velocity, and veracity of the big data at scale. Model fusion is an approach to integrate data on a large scale.

Besides mining raw data, process mining provides insights into the business-level processes that are based on operational processes and event logs. Having this view, the business can identify inefficient points and improve the processes. Likewise, Information extraction is used with textual data to retrieve information. Information extraction is mainly used in un/semi-structured data.

One technique is to use a template with specified list of slots filled with substrings extracted from text document, websites, and databases. Using the spaCy library in Python, the position of a word in a sentence and POS can be found together with its relation to other words in a sentence.

UNIT 4

TYPES OF DATA SOURCES

STUDY GOALS

On completion of this unit, you will be able to ...

- describe what an API, flat file, and unusual data structure are.
- illustrate the structure and architecture of relational databases.
- understand the differences between un-relational and relational databases with examples.
- explain stream data and the tools used for their processing.
- identify and describe an open data source and its characteristics.

4. TYPES OF DATA SOURCES

Introduction

In this unit, we will investigate the structure of various data sources together with understanding their structures, importance, and use cases. The purpose of data sources is to connect users and applications to related services. Relevant technical information is gathered in one place to make data valuable. It is important to know the sources of data, the structure, architecture, characteristics, and the tools to access data. It's also crucial to know how data are hosted and transported between various different applications and the role of application programming interfaces (APIs, with the emphasis on "interfaces") for connecting components. The source of the data, as well as their architecture in terms of relational and un-relational databases, will be discussed as well.

The tools and the characters of streaming data and public data will be further explored in this unit. Flink as a useful tool for processing streaming data will be introduced. In addition, the importance of open data and how they differ from public data will be discussed. Open data are a source offered for purposes such as clarity, transparency, improving quality, and identifying the opportunities based on the levels the data are offered in. By "levels," we mean whether the data are governmental or organizational.



4.1 APIs, Flat files, and Unusual Formats

A number of formats will be addressed in the following sections.

Application Programming Interfaces

APIs are used to build distributed software systems with loosely connected components. APIs are the connectors and communicators between software systems. This means APIs help programs to communicate with each other. They act as the middlemen between machines and the businesses owning those machines to make web products available across the internet. They accept requests from applications and return the desired services, data, or functionalities. Representational state transfer (REST) is a common architectural style that APIs work in, to enable interacting via **Hypertext Transfer Protocol** (HTTP) protocol. They're also called RESTful APIs (Biehl, 2016). However, the interface is not something visible to the end users. The only people who are interacting with the interfaces are the developers. The architecture of REST API's includes elements of a client, a server, and a resource. The client is the software that runs on the end user's device (e.g., a smartphone, laptop, etc.) and initiates communication by sending an HTTP request. The server offers the API providing access to the desired data or functionalities by generating an HTTP response. Resource is the actual content that is offered by server, which can be a text or video file (Jablonski et al., 2013). The rest of the request includes the HTTP method describing what to do with the resource. There are four basic methods, namely POST to

Hypertext Transfer Protocol

This is used to transmit hypermedia documents.

create a resource, GET to retrieve a resource, PUT to update every source, and DELETE to delete the resource. These four basic methods are called create, read, update, delete (CRUD) operations (Biehl, 2016).

API structures started to develop in 1991 with Common Object Request Broker Architecture (CORBA), Raster Data Access (RDA), Extensible Markup Language remote procedure call (XML-RPC), and Simple Object Access Protocol (SOAP; Jablonski et al., 2013). REST replaced SOAP around year 2000 and more APIs such as JavaScript Object Notation remote procedure call (JSON-RPC), Open Data Protocol (ODATA), Graph Query Language (GRAPHQL), and google Remote Procedure Calls (gRPC) were developed. RPC, SOAP, REST, and GRAPHQL make up the four main API architectural styles. REST is one of the preferred styles since its compliant with many architectural constraints. It supports XML, HyperText Markup Language (HTML), JSON, and plain text. RPC supports as many formats as well, but is perhaps not the most famous one. REST has a large community, and public APIs are using this style because of the simple resource-driven nature of it. REST and RPC are the most comparable API architectures. The command line and action-oriented APIs use RPC style. It's also used in high performing communication systems (Jablonski et al., 2013).

Flat Files

Flat files are collections of data that are stored in a two-dimensional (2D) dataset, including columns as one dimension and rows as another dimension of the dataset. These types of data are stored alphanumerically with almost no formatting. A common example of a flat file is comma separated values (CSV) file. In a CSV file, the columns and rows show this by tabs or commas, and a flat-file database usually comprises of a single table. There is no limit to determining how much data can be saved in a flat file. It can depend on the memory of the operating system. To manipulate the content of a flat file database, tools such as column sorting and searching are used. Sorting helps to arrange the data in ascending or descending order based on the contents of the column. Searching helps to find specific pieces of data text or numbers throughout a flat file.

Flat files are considered easy to create, work with, and maintain. applications including spreadsheets like Excel or Google Sheets are of such kinds. They are also widely used on the Internet of Things (IoT), data lakes, and data warehouses since they provide little **overhead** as well as accessibility to store a large mass of information that needs to be maintained in its natural form.

In contrast to relational databases, flat files databases are represented using a data dictionary. Moreover, the flat file database consists of a single file with no structured relationship while, in a relational database, there are multiple entities that are only presented using a schema (Carpenter, 2010).

Unusual Formats

Data types usually fall into common categories: observational, experimental, derived or compiled, simulation, reference, and canonical. The data comes in different formats of text, numbers, media, models, software languages, discipline specific format, and instrument specific format.

Overhead

This is excess computation time, memory, or bandwidth required to perform a task.

Among the many unusual data formats here, the two data formats of structures (structs) and map data are discussed.

Struct data type is a collection of elements from different data types. This format has an associated schema defining the structure of the data. The schema determines the number of elements and their names in the data. Processing and modifying destruct data requires the inclusion of functions that can manipulate in this schema of struct data. An example is shown below:

```
Schema[element_name1:data_type1[,element_name2:data_type2,...]]
```

Struct data example:

```
[street_number:34,street_name:Bulevard Ave, city: Lancaster City,  
state:CA,zip:52231]
```

Map data type is a collection of key value pair elements that are not ordered. The elements in map data are key and value pairs, mapping one item to another. To read and write map data in complex files, spark engine can be used. As an example, a map data format can include an integer key and a string value to map customer identification (ID) to the customer's name. The following shows the map data format and applies it in an example

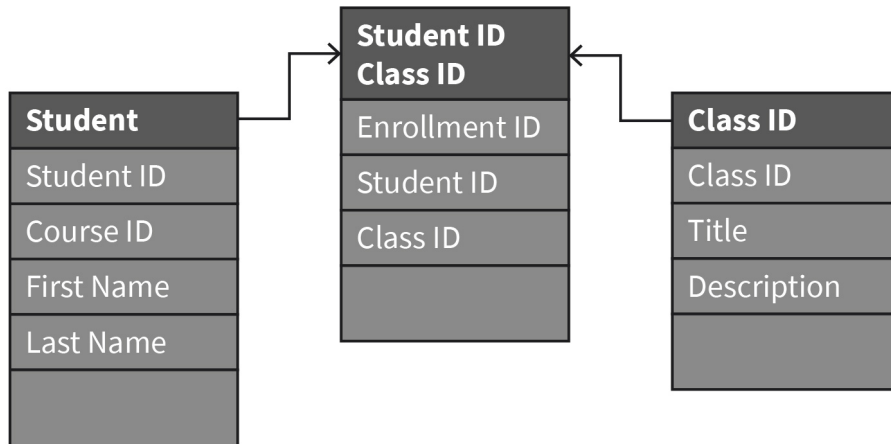
```
custid_name <integer, string>  
<46231,'Eric Sharafi'>
```

4.2 Relational Databases

The term “relational database” was first defined in the 1970s by Edgar Frank “Ted” Codd (1970) as a theory of data management. Codd defined what is meant by “relational.” He showed that the complex relationships in data can be presented with simple concepts. The common definition of what constitutes a relational database system includes the 12 rules from Codd’s research works.

A relational digital database is based on the relational model of data according to Codd. With the relational model the data is organized into tables with a unique identifying key per row (records). The table represents the entity type (e.g., a customer or a product). Each row in the table has its own unique key. Rows in one table can be related/linked to two rows in another table. To have this connection, a column is added to the table that includes the unique key of the linked rows from the other table. This column is called foreign key. It refers to the field in the relational table matching the primary key column of another table. The column in the table including unique identification numbers per rows is the primary key used to access the table. The primary keys are unique, and this allows only one row with a given primary key value to disable (Juba & Volkov, 2019).

Figure 11: Schematic View of Primary and Foreign Columns



Source: Somayeh Aghanavasi (2024).

This figure shows the number and the names of the columns existing per table. There are three tables: one for students, one for classes, and one in the middle that serves as a joining table. Each field represents a column name containing values. The columns containing primary key in students and class tables are noted within a star (*). The middle table contains the foreign keys used to connect the student and class tables. Having the tables with their primary keys, foreign keys, and defining relations, stored procedures can be written and executed. They are used to collect and customize common operations, such as inserting data in relation, collecting data, and performing calculations from tables. A relational database management system (RDBMS) maintains relational databases. Many such systems are equipped with Structured Query Language (SQL) for maintaining the database and querying data.

PostgreSQL, also known as Postgres, is a free and open-source management system for relational databases. It has complied with SQL. It is the default database for Mac OS servers and is available in Windows, Linux, FreeBSD, and OpenBSD. Connecting to applications, PostgreSQL includes built-in interface libraries and embedded sea system. Third-party libraries to connect to PostgreSQL are available in many programming languages including Python, Node.js, C++, and Java. PostgreSQL has been adopted by many large organizations.

PostgreSQL is available to common operating systems and to download from its website. The installation is usually in two parts: the PostgreSQL server and a graphical user interface such as pgAdmin. In some cases, Stack Builder is also used to download and install the drivers and additional tools. Then, PostgreSQL and pgAdmin need to connect to enable exploring the databases and querying them. The pgAdmin interface allows creation and modification of the databases right from the interface. The same thing can be done using the SQL queries. The below query can be written instead.

```
CREATE DATABASE "STUDENTDB"  
WITH  
OWNER = postgres
```



```
TEMPLATE = postgres
ENCODING = 'UTF8'
TABLESPACE = pg_default
CONNECTION LIMIT = -1;
```

Admin is one of the most used graphical user interfaces (GUIs) by Postgres users. It makes databases easy to find on the left-hand menu. So far, there have been four major versions of this GUI. It's an open-source application supporting all PostgreSQL's features. Some of its strengths include its ability to run as a web application, making it able to be deployed on any server, including the computers. This makes it suitable when a Postgres distributed database runs across multiple servers, enabling PgAdmin to be included on each. Perhaps one of the limitations of PgAdmin is the installation for SQL developers who are not familiar with command line. The other alternative GUIs are Navicat, DBeaver, HeidiSQL, Data-grip, OmniDb, and IntelliJ IDEA – JetBrains.

PostgreSQL is also available in Google Cloud. You can connect cloud SQL PostgreSQL from a cloud shell. Please note that the billing account needs to be created before starting an instance with SQL querying in PostgreSQL. When you have a graphical user interface installed with PostgreSQL, using SQL is straightforward.

4.3 Non-Relational Databases

In contrast to traditional relational databases, non-relational databases do not use tables or tabular format with rows and columns. Non-relational databases use storage models instead. They are specific with the type of data stored and queried (e.g., graph data, or time series data). For example, non-relational data can be stored in a document format. Non-relational databases are more flexible than relational databases since they can digest and organize various data types of information. The NoSQL term used for non-relational databases refers to data stores not using SQL for queries. When there is a large amount of diverse data to be organized, non-relational databases are the better choices for the mentioned reasons. For example, a large database in a company can contain customer information stored in document format. The documents can contain names, addresses, credit information, and order history. Different formats of data can be stored in the same document (Tejada, 2022).

Queries in relational databases search over several tables, and every time the data changes, the query must change to look for the right data. Non-relational databases generally perform faster since they do not deal with SQL queries. They support applications that rapidly change and require dynamic databases containing complex and unstructured data (Sharma, 2021). Scale and speed are two important characters in non-relational databases. They are flexible in expansion even when the new data is of new types and at different granularity. Major categories of non-relational data stores are document, columnar, key/value, graph, time series, object, and external index.

Document data store contains a set of named string fields and object data values in an entity that is referred to as a document. The data can be stored in the form of JSON, YAML, BSON, or plain text. The field values can be of numeric or string format. The document contains an entity. The entity contains, for example, a customer and order or both. Each document is to be identified with a unique key which is used for document retrieval and is often hashed key is used to distribute the data evenly. The below figure shows an example of a document store (Tejada, 2022).

Table 6: Non-Relational Document Data Store

Key	Document
1101	<pre>{ "StudentID" : 9092 "Courses" : [{"CourseID" : 31001, "Course_date" : 2021, "Course_teacher" : Helen Jones" }, {"CourseID" : 37349, "Course_date" : 2022, "Course_title": "Data mining" }], "Entrance_year" : "2018/01/01" }</pre>
1102	<pre>{ "StudentID" : 9093, "Courses" : [{"CourseID" : 35302, "Course_date" : 2019, "Course_teacher" : Reza Salimi" }, {"CourseID" : 35202, "Course_title" : "Data Analysis" }], "Entrance_year" : "2021/08/01" }</pre>
1103	<pre>{ "StudentID" : 9095, "Courses" : [{"CourseID" : 31057, "Course_date" : 2021, "Course_teacher" : Anna Normand" }, {"CourseID" : 37349, "Course_date" : 2022, "Course_title" : "logic and mathematics" }], "Entrance_year" : "2019/01/01" }</pre>

Source: Somayeh Aghanavesi (2024).

The data in non-relational databases can be stored in columnar data stores. The data are stored in columns and rows. It appears very similar to relational database, conceptually. Columnar data store uses a denormalized approach to structure sparse data. The difference here is that the columns are divided into groups known as column families. Each column family contains a set of columns that are logically related. The data that are accessed separately can be stored in a separate column family. The rows in a column family can be sparse, which means a row doesn't have to have any value for every column. The below figure shows an example of columnar data. Note that an entity of data has the same row key in each column family. The Rows in a column family can dynamically vary, so they are suitable for varying schemas. Key/value data store format is like columnar data; however, there are differences on level of consistency, strategy (optimistic in contrast to pessimistic), access pattern, and indexing. Furthermore, in contrast to columnar data, they cannot include many columns or attributes (Tejada, 2022).

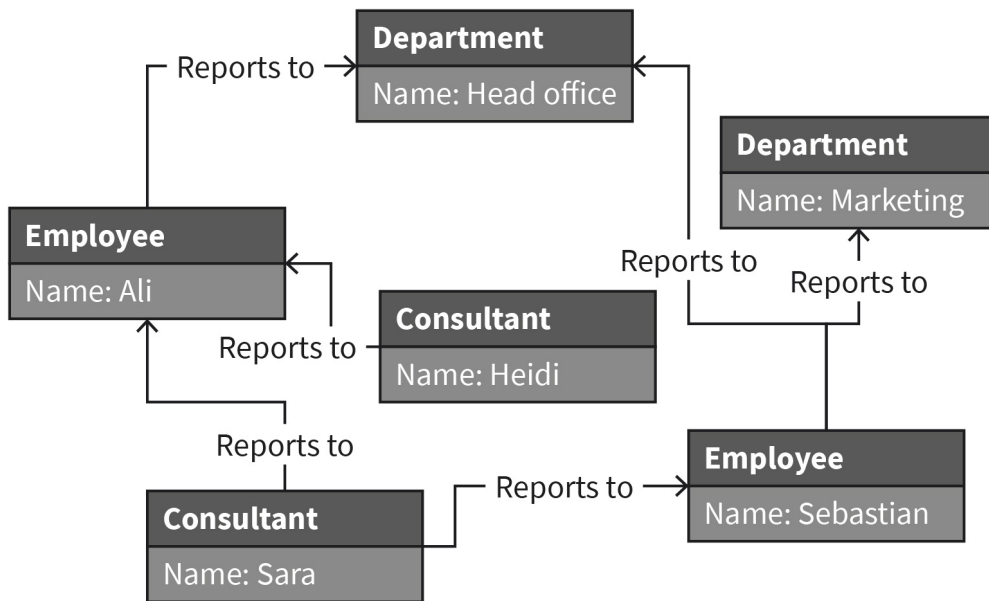
Table 7: Non-Relational Columnar Data

StudentID	Column family: Identity
0001	First name: Jack Last name: Jonsson Suffix: Jr
0002	First name: Ali Last name: Forsblad Title: Dr.
0003	First name: James Last name: Ekberg Title: Prof.
0004	First name: Mohd Last name: Englund
StudentID	Column family: contact info
0001	Phone number: +46-72031
0002	Phone number: +92-481080 Email: name@air.com
0004	Email: name_01@exp.com

Source: Somayeh Aghanavesi (2024).

Another data type suitable to be stored in non-relational form is graph data. Graphical history can manage two types of information: nodes, and edges. Nodes represent the entities, and edges are specified relative to the entities. Edges have directions, which indicate the nature of the relationships. Having graph data in this form allow queries to traverse and analyze the relationships between the entities for an efficient result. The example below shows an organization's personnel data structured in non-relational graph data. The arrows show the form of relations between the nodes (Tejada, 2022).

Figure 12: Graph Data



Source: Somayeh Aghanavasi (2024).

4.4 Streaming Data

This section investigates stream processing with Apache Flink, as streaming technologies to process data. In many application domains, streaming data are generated. For example, data could be generated from IoT sensors and devices, transactions from banking systems, feeds provided by users, and much more. Traditionally, data streams were stored in databases and then processed. However, businesses require real-time processing of such data, enabling them to get insights from data directly without spending much time on the processing. Real-time processing is supposed to be close to data sources. Flink, an open-source framework, is useful for larger scale, distributed, and fault tolerant processing of data streams (Qadah, 2018).

The building blocks of the Flink pipeline are the input, processing, and output. It provides real time data (event streams such as **Automated Teller Machine transaction** [ATM transaction] alerts) analytics of streaming data specifically when continuous extract, transform, load (ETL) is considered. The operations on data streams are mapping, filtering, grouping, updating state, joining, defining windows, and aggregating. Flink works with both Datastream and datasets. In another world when the data is in finite and infinite boundaries. For dimensioned data boundaries, Flink has a rich set of APIs, through which developers can perform the mentioned operations. The APIs are supported by Java and Scala programming languages (Hueske & Kalavri, 2019).



ATM transaction

These include any actual or attempted use of an ATM, including to cash withdrawals, cash deposits, fund transfers, and checking account balances.

The ecosystem of Apache Flink includes four stages of storage, deploy, Kernel, APIs, and libraries. Flink has the capability of reading and storing multiple data, such as the following:

- Hadoop distributed file systems (HDFSs), S3, local files
- databases such as mongoDB, RDBMS(MySQL, Oracle)
- Kafka, RabbitMQ streams

Flink conducts the deployment in local mode, cluster mode (standalone, YARN, MESOS), or on a cloud (AWS, GCP).

To start with a local cluster to execute a streaming application, follow the steps below (Hueske & Kalavri, 2019). The application converts and **aggregates** randomly generated temperature sensor readings by time. For example, your system needs Java 8 to be installed. The application needs to run in a Linux environment. If you are running windows, you can get help having Linux via a virtual machine installation or use a windows subsystem (introduced by Windows 10). Start a local Flink cluster and submit an application for execution:

Aggregation

This is a whole formed by combining several separate elements.

1. Download Hadoop-free binary distribution of Apache Flink 1.7.1 for Scala 2.12 from Apache Flink's webpage.
2. Extract the archive file using the following code:

```
$ tar xvfz flink-1.7.1-bin-scala_2.12.tgz
```

3. Start a local Flink cluster:

```
$ cd flink-1.7.1
$ ./bin/start-cluster.sh
Starting cluster.
Starting standalone-session daemon on host xxx.
Starting taskexecutor daemon on host xxx.
```

4. Open the Flink's web user interface (UI; <http://localhost:8081>) in a browser. Some statistics will be shown about the local Flink cluster. The available task manager and task lot will be shown.

4.5 Open Data Sources

Open data are allowed for reuse. The purpose of having open data is to provide public oversight of organizations and governments with the hope to reduce corruption by adding greater transparency. For instance, open data makes it easier to monitor organizational or governmental activities by looking over public budget expenditure data in relation to environmental policies.

Governmental open data enables public service improvements. It can be done when citizens use open data to contribute to public governmental planning, or by providing feedback to organizations' key parties or governmental ministers on service quality. Open data is data which are public and can be reused, promoting a key resource for public innovation and economic growth. It provides new opportunities for collaboration between governments and citizens, as citizens with access to public services will be able to evaluate them. Entrepreneurs and knowledge experts in businesses use open data for understanding potential markets and the opportunity for building new data-driven products. Through having data public and available, governments or organizations reduce the cost of data acquisition, redundancy, and overhead. This is due to it being easier and less costly for them to discover and access their own data or data from organizations (Foth et al., 2015).

Public data can be open data and vice versa. However, both public and open data can have limitations. It's recommended to always check the policy of the data for reuse. If the data are explicitly noted as public but not open, it indicates that there might be limitations to reusing it. Open data can be offered by country, sector, or topic (Janssen et al., 2017).

There are various technologies used to develop open data platforms. Open data catalogs that are web-based systems are used to make data available to the end users. A data catalog is a set of datasets available for open data initiatives. The essential elements of a data catalog include searching, metadata, license information, and access to datasets. A platform provides online access for the users to access the available resources from open data initiatives. A platform includes a data catalog together with the information and the services. They usually include online forums, support, and feedback (Charalabidis et al., 2018).

We mentioned that the purpose of having open data is to add to a service's **transparency** and quantity. Determining the quality needs a clear definition of what measures are to be taken for open data. There are six relevant quality dimensions that are defined as the statistics to identify or measure the quality of open data. The quality dimensions are relevance, accuracy and reliability, timeliness and punctuality, accessibility and clarity, comparability, and coherence. In addition, open data are supposed to be specifically public, accessible, described, reusable, complete, timely, and interactive (Janssen et al., 2017).

Transparency

This is the ability to easily access and work with data no matter where they are located or what application created them.



SUMMARY

Application programming interfaces (APIs) are the connecting points or the communicators between different software systems. This architecture style of APIs is the REST API, which works with the methods of POST, GET, PUT, and DELETE. These four methods are called CRUD operations. Flat files are 2D datasets, mostly in the form of columns and rows, that store data. A common flat file is a CSV file. Flat files are represented using a data dictionary.

Further relational and non-relational databases were described in this unit, along with their differences. Relational databases involve the relations that are defined between the data. There are primary keys and for-

foreign keys to show such relations in databases. A common language to work with relational databases is PostgreSQL or Postgres. PostgreSQL usually comes in two parts. PostgreSQL and pgAdmin. In contrast to relational databases, non-relational databases do not work with defined relations in tables. They come with different storage models with embedded identification of the relations inside the data. Non-relational databases come in the forms of document data store, columnar data, and graph data.

In this unit, we explored how to process streaming data and introduce Flink as a common toolkit. The building blocks, ecosystem, and introductory level of how to work with Flink were discussed. This tool provides local, cluster, and cloud base data deployment.

Finally, we observed the differences between open and public data. We looked into the purposes and benefits that governments and people can get from accessing open data. To ones who cannot collect the data on their own, having access to open data with the possibility of reusing them is beneficial. However, it is important to verify the policy of the data before reuse.

UNIT 5

DATA MINING TECHNIQUES

STUDY GOALS

On completion of this unit, you will be able to ...

- describe common statistical methods used for data mining.
- explain the general concept of machine learning together with linear regression and principal component analysis.
- identify the architecture of operational data store in data warehousing.
- apply event processing to use cases and describe its architecture.
- identify real-time and near-real-time processing.

5. DATA MINING TECHNIQUES

Introduction

The techniques and data mining include methods from statistics, artificial intelligence, and machine learning. It requires understanding the structure and architecture of the systems that store and process data, as well as reporting the results made from them. To gain a sufficient understanding of the above-mentioned concepts, this unit will look into some common methods used in statistics for analyzing data. The two common tests are t-tests and chi-square tests. T-tests are used for measuring the significance of an effect when a hypothesis is assumed to be rejected or approved. A chi-square test also includes a hypothesis to compare two sample sizes.

The importance, usage, pros, and cons of using machine learning are introduced in this unit with practical examples. Linear regression and principal component analysis are the two subjects discussed in many machine-learning concepts. This unit continues with data warehousing, event processing, and real-time processing. As part of data warehousing, operational data store (ODS) is a central database providing a snapshot of the latest and updated data from the transactional systems. Event processing provides a perspective of how the events are in normal life and how they can be digitalized together with their identification and their processing. Reviewing the events in a system facilitates its enhancement and enables solutions to solve the bottlenecks in complex event analysis.

5.1 Statistical Methods

You already are familiar with different distributions (e.g., normal, negative skewed, and positively skewed distributions) to introduce different types of populations or probability distributions from the statistical point of view. Of course, distributions are not limited to only those; there are also Bernoulli, uniform, binomial, poisson, and exponential distributions. In data mining, it's essential to be familiar with statistical distributions, so let's briefly describe each of the remaining distributions.

Bernoulli distribution has only two possible outcomes: 1 as success and 0 as failure. Each observation is a single trial: 1 is the probability of success, and the value 0 is the probability of failure (q), or one minus success ($1-p$).

Uniform distribution has the outcome of one to six, just like when rolling a dice. The probability of getting each of the numbers is equally likely, making the basis for uniform distribution. Unlike the Bernoulli distribution, all the observations have an equal outcome.

With binomial distribution, the outcomes are of only two values success or failure, gain or loss. This is when the probability of each of the two values are similar or 0.5, shaping a binomial distribution. But the outcomes do not need to be equally likely and each of the trials are independent from the previous outcome with trying a toss.

An example of Poisson distribution is the number of customers arriving at a salon in an hour. It is applicable in situations where events occur at random points of time and space. With Poisson distribution a successful event does not influence the outcome of another successful event. The probability of a successful event during a short interval is equal to another successful event during a long interval. When the interval becomes smaller the probability of success in that interval gradually becomes 0 (Woolfson, 2008). In Woolfson (2008), experimental examples are provided with real case scenarios to understand the probabilities with better sense. An example of exponential distribution is the length of time between arrivals of cars at a gas station.

The type of statistical methods can be viewed as four methods of parametric inferential statistical methods, nonparametric inferential statistical methods, predictive statistical correlation methods, and predictive statistical regression methods. The parametric inferential methods carry out the following search and parameters on data: The t-test is considered a parametric inferential method. It's a statistical test used to compare the means of two groups. T-tests are quite common with hypothesis testing. Hypothesis testing is a test to determine whether a solution has an effect underpopulation. It's also used to compare the two groups. AB testing is one of the examples of hypothesis testing. It's used when a hypothesis, or an assumption, is made about the relationship between two groups of data. The test is used to approve or deny the relationship. In Python, a t-test has three main types: (1) one sample t-tests, (2) two sample t-tests, and (3) paired tests. Let's try one sample t-test in Python.

T-Test

This test is used to compare the meaning of a sample set with a specific value. This specific value is assumed as the meaning of the population, which is also called the hypothesis here. For example, the watches produced by a factory should run for a year on average. We want to check whether the average diameter of a watch from the random sample picked from the production line differs from the known size. The null hypothesis here is that the sample mean is equal to hypothesized or known population mean. The alternative hypothesis is that the mean of the sample is not equal to the known population mean. Another alternative hypothesis is that the sample mean is larger or smaller than the known population mean. In Python, we can use SciPy library with the code shown below to try this t-test.

```
from SciPy import stats as sp
from bioinfokit.analys import get_data

# load dataset
df = get_data('t_one_samp').data
df.head(3)

# output
      size
0  5.739987
1  5.254042
2  5.884379
```

```

# t test
size = df['size'].to_numpy()

# use parameter "alternative" for two-sided or one-sided test
sp.ttest_1samp(a=size, popmean=5)

# output
Ttest_1sampResult(statistic=0.36789,pvalue=0.71453)

```

Chi-Square Test

The chi-square test is primarily used for the examination of two categorical independent variables. It's a statistical hypothesis that is used when sample sizes are suitable for the analysis of contingency tables to see whether they are related. This test can be used to examine whether there is a relationship between two variables of the population. A contingency table is used to summarize the relationship between the variables. Again, a null hypothesis can be made. Usually, the null hypothesis assumes there is no relation between the variables. The alternate hypothesis assumes there is a **significant relationship** between the two. The significant factor is provided by the p-value which is an alpha value of 0.05. The alpha value denotes the probability of rejecting the null hypothesis. If the p-value is greater than the alpha value, then the null hypothesis is true; otherwise, it is rejected. For chi-square, if the calculated value is less or equal to the value of chi-square, then the null hypothesis is true. There is an expected value for the table including the calculated or expected values. It's calculated with the below formula:

$$\text{Total of rows} \cdot \text{total of columns} / \text{grand total}$$

Chi-square table is calculated as below:

$$(\text{Observed value} - \text{calculated value})^2 / \text{calculated value}$$

Putting this into practice with Python, we can use SciPy library again for a chi-square test to compare a few records given in two arrays as columns of a table.

```

from SciPy.stats import chi2_contingency

# Defining the table
Table = [[217, 202, 239], [253, 231, 242]]
stat, p, dof, expected = chi2_contingency(Table)

# Interpreting the p-value
alpha = 0.05
print("p value is " + str(p))
if p <= alpha:

```

Significant relationship
This is when correlation analysis is performed and the p-value is less than 0.05.

```
print('Dependent (reject H0)')
else:
    print('Independent (H0 holds true)')
```

The p-value is 0.10231, and, therefore, the H0 holds to be true since the p-value is larger than alpha.

5.2 Machine Learning

Perhaps one of the main differences between data mining and machine learning (ML) is that data mining is about extracting the rules from large quantities of data while ML is about how a computer learns about the data and comprehends the given parameters. In other words, data mining is used on existing data to find patterns. ML is used to put the data as input into an algorithm learning about the patterns and replicating the next steps.

Predictive modeling and ML go hand in hand. Predictive models include ML algorithms. The models can be trained over time to respond more accurately to new data delivering the desired results and use cases in businesses. Predictive modeling largely overlaps with ML concepts. But why ML is important? ML is part of artificial intelligence (AI), which has been a concept for quite some time. The learning ability of the algorithms emphasizes the importance. Algorithms can learn and make predictions on data.

ML has become important since it can solve problems at a speed and scale which cannot be performed by the human mind alone. Machines can be trained to identify the parents and their relationships between the variables and automate the processes beyond human capability. For ML, data are the key and AI is the goal. ML can be used in data security, finance, healthcare, fraud detection, and retail. The training methods in ML differ. As is mentioned before, there is supervised learning when there is a defined target value. With unsupervised learning, there is no target value, and the machine is supposed to create the trends and identify the parents or classes in the data. There's also another method, the semi-supervised method (also called reinforcement learning), which rewards the outcomes. This method interacts with its environment by producing actions and discovering errors or rewards. One of the important characteristics of reinforcement learning is the trial and error searching and the delayed rewards (Kotu & Deshpande, 2019).

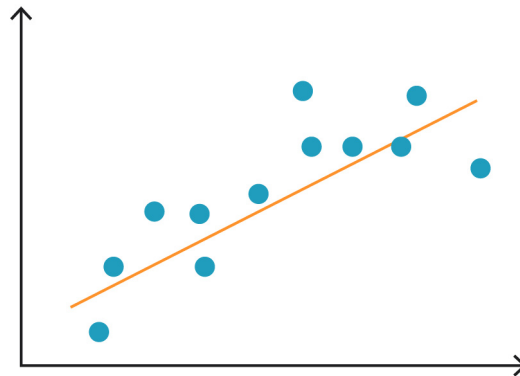
There are a few things to note about ML. We said the key is the data, and that means ML, therefore, is not based on knowledge. ML models take effort to be trained. If there are issues in the data, the ML models get directly affected. The errors in the data can cause overfitting or errors; for such reasons ML often becomes biased. It's considered a kind of **black box**, where it takes data as input and provides results without providing a view of what happens entirely in the algorithm. Therefore, it is critical to make sure how the data fits into the method, what happens to the data in the entire algorithm, and how the results are achieved to avoid having black-box ML learning (Kotu & Deshpande, 2019).

Black box

This term is used when the entire process is unknown, and it acts like a box providing output to a given input.

Another instance of supervised learning is linear regression. Linear regression models the relationship between the input value and the target value. The input value is also called the independent variable, and the output for the target value is the dependent variable. The goal of linear regression is to identify the pattern by drawing a line over the observations and finding out the trend and giving the next set of new data the algorithm to find the next target values. Below is a figure of how linear regression finds the line matching the observations (Brandon, 2019).

Figure 13: Linear Regression



Source: Somayeh Aghanavesi (2024).

Other instances of supervised learning are K-nearest neighbors, decision trees, random forests, and support vector machines. With unsupervised learning, there are dimension reduction algorithms, density estimation, and clustering. A famous method for dimensionality reduction is principal component analysis (PCA). PCA can be used to reduce the dimensionality from n components to the configured number of desired dimensions. It's a fast and flexible unsupervised method. In Python, the libraries NumPy, Matplotlib, seaborn, and sci-kit learn can be used for PCA.

5.3 Data Warehousing

Data warehousing is identified as including methods of organization and compilation of data into one database. There's a difference between data warehousing and data mining. Data mining includes the methods for fetching important data from databases (Bhatia, 2019).

Operational Data Store

In the context of data warehousing, an operational data store (ODS) is a central database providing a snapshot of the latest and updated data from the transactional systems. It usually stores and processes data in real-time. It is connected to multiple data sources and pulls the data into a central location. You might ask why ODS is needed. It enables organizations to combine data from various sources into one single destination. This helps make

it available for business reporting while it contains up-to-date information. Information is integrated from operational sources, and it enables the support from business intelligence tools facilitating data-driven decision making (Bhatia, 2019).

To reflect on the differences between ODS and a data warehouse, ODS is designed to perform simple queries on small sets of data, while, on the contrary, a data warehouse is designed to perform complex queries on large datasets.

The architecture of ODS includes three pillars. The first is the extraction of raw operational data from small production data sources. The second is the transformation of operational data systems into staging. The third is to load the staging original data into the data warehouse. The way operational data stores work is comparable to the extract, transform, and load (ETL) process.

For data warehousing, the architecture starts with getting the data from operational systems and external data into transaction databases. After preprocessing cleaning and transaction, ETL processes data together with metadata residing in data warehouse databases (Bhatia, 2019).

Having data warehouses means that data marts can be generated. A data mart is a subject-oriented database. It is usually a partitioned segment of an enterprise data warehouse. The data that are held in the data mart specifically are aligned with a specific part of the business unit, such as finance, marketing, or sales. It's considered a shortened version of the data warehouse. The three different types of data marks are independent, hybrid, and dependent.

Independence data mart uses this top-down approach to be created from an existing data warehouse. With the top-down approach, it means that it begins to wear storing business-related data in one central location, then extracting portions of data needed for a section in the business. Contrary to the dependent data marts, the independent data mart is created without any dependency on a data warehouse. Data is extracted from a data source, transformed, and loaded into the data mart repository. Maintenance of independent data marts can be challenging, and they add to the complexity as the business grows more with independent data marts. Hybrid data marts are the combination of data from warehouse and operational source systems. It uses the top-down approach to figure out the focus on the end user, and the advantages of enterprise level integration of the bottom-up method (Bhatia, 2019).

5.4 Event Processing

Let's first look into the definition of an event: It is an occurrence within a domain or system. It's something that has happened, and in programming, an event presents an occurrence in the computing system.

Even in our normal life, we encounter a lot of events. Some events are considered basic: getting emails or answering phones. Some events are unexpected: robberies or a late train. Not all the unexpected events are negative: having an old friend calling you, getting a good score in a course, or winning the lottery all have positive connotations. Some events can be deduced from the prior ones. For example, if after three weeks you consider that the amount of bread consumption at home has increased since every week the amount that was purchased is the same but it has run out sooner than the week's end. You thus start buying more bread in your weekly shopping. Not only has the "running out of bread" event occurred, but a higher-level event has also occurred, the "bread consumption has increased," which is a deduction from an earlier event (Niblett & Etzion, 2010).

The purpose of learning about events is to identify the opportunities to react to the ones that occur out of normal situations. It will be the same for event processing in computer systems. The main purpose of event processing is to detect and report situations so that they can be reacted to. The reaction might be a simple action. As in our example with purchasing extra bread, it is important to realize that the consumption has increased.

To provide an example of event processing in computer systems, we can refer to metro trains that enable commuting in a city. There are multiple trains on multiple paths with a schedule to stop by stations and start leaving the stations on a timely plan. Part of the plan is to monitor the distances between the trains. There are sensors placed at the end of the trains measuring the amount of distance by sending wireless signals and detecting the distance to the next train. The signals get processed, and distance is measured and monitored. Usually, every train has the normal defined distance kept with other trains. If, all of sudden, this distance decreases, it can endanger the safety of the train and the passengers. In this case, the event is to identify if the distance between the trains is getting shorter or longer than what is defined or what is considered to be normal. There are five categories of event processing applications:

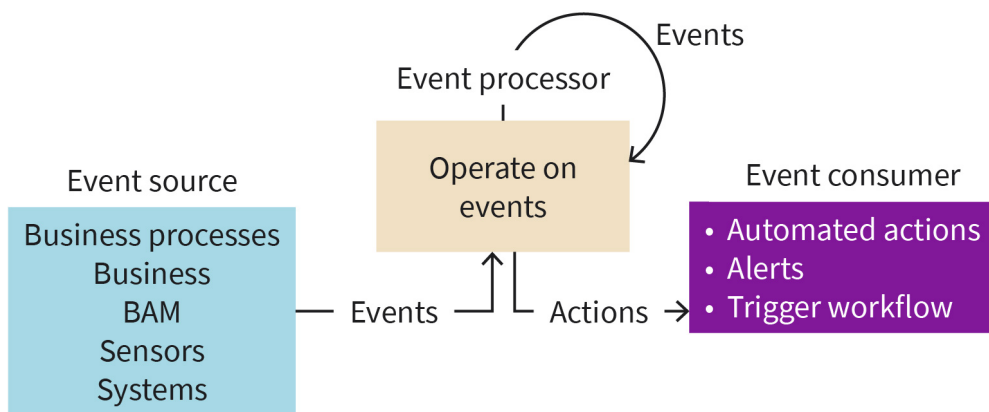
1. Observation: This is when event processing is used for monitoring a system or identifying exceptional behaviors and generating alerts.
2. Information dissemination: The reason for this event processing is to deliver the right information to the right entities, with the right granularity level, and at the right time. It's also called personalized information delivery. An instance of this case is a personalized bank system allowing customers to set up alerts for cases such as when a sum withdrawn from the account goes beyond a certain number, or even with a certain investment portfolio that gains more than 5% during a trading day.
3. Dynamic operational behavior: This is an event processing that often uses actions conducted by systems dynamically to react to specific incoming events. For example, consider an online trading system matching buys and sells requests in an auction. The events are the buy and sell requests, and a system needs to dynamically match the process using patterns such as setting the priorities based on orders and setting risk to buyers or sellers based on their trading history.
4. Active diagnostics: These are all in this event processing to diagnose a problem, based on what's observed as symptoms. For example, consider a manufacturing plant management system that is about to diagnose mechanical failures that are observed as

symptoms. The events are the symptoms, and the events describe what is not working. The purpose is to find the root cause of symptoms. Thus, a set of events is observed, collected, and then discussed to identify the root cause.

5. Predictive processing: The goal of this event processing is to identify events before they happen. This mitigates problems before they happen. Serious cases like fraud detection involve predictive processing (Niblett & Etzion, 2010).

The architecture of event processing is based on the interactions between the components. The components are the event source, event processor, and event consumer (IBM, 2021).

Figure 14: Event Processing Architecture



Source: Somayeh Aghanavesi (2024), based on IBM (2021).

Event sources could be the signal/data generators, the flow of business information between systems, or radio frequency identification sensors. The event processor can add a timestamp or information about the source of the event to the data. It processes the events against patterns to produce a new compound event or knowledge about the events. This can also be called complex events processing (IBM, 2021).

5.5 Real-Time Processing

Real-time processing considers the methods of processing the data immediately. As soon as the data arrives at the processing unit, the output is available. With this characteristic, a continuous data flow would be required. The term “real-time” is, in fact, a myth. In reality, it is true that the analytics are getting closer to the data; however, while the gap between the data and analytics is becoming smaller, there is still a small amount of time necessary for performance. It’s impossible in practice to eliminate this gap since the computation, operation, and network latencies are near-to-real time rather than real-time (Saxena & Gupta, 2017).

What are the high-level expectations from real-time analysis? The data are produced from multiple sources including structured and unstructured data. The processing engine is supposed to handle ultra-fast and complex logics. It's expected to generate accurate reports that can be reverted to the ad-hoc queries in less than a second, as well as render the visualizations and dashboards with no latency (Saxena & Gupta, 2017). Touching the system-level expectations in terms of data, processing, and output, the systems are designed to process billions of transactions applying complex AI and ML methods in real time. The below ~~figure~~ ^{table} illustrates computation time.

Table 8: Real Time Analytics

Real-time analysis			
> 1 hour, high throughput batch Adhoc queries Monthly active users relevance for ads	> 1 second, approximate online Ad Impressions count Hashtag trends	< 500 milliseconds, latency-sensitive online Deterministic workflows Fanout tweets Search for Tweets	< 1 millisecond, low latency near real time Financial trading

Source: Somayeh Aghanavasi (2024), based on Saxena & Gupta (2017).

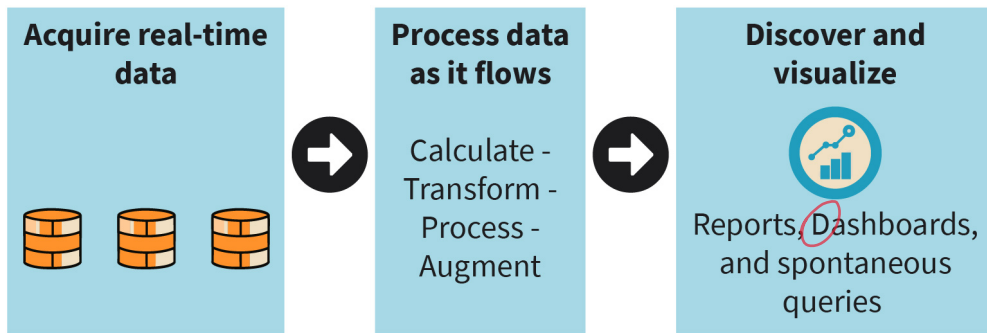
Ad-hoc queries
 These are single questions or requests for a database written in SQL or another query language by the user on-demand.

Ad-hoc queries larger than zetabytes of data take up hours of computation time and are represented typically as batches. Some use cases, such as ad impressions, hashtag trends, determinist workflow or Tweets (noted as online in the figure), considerably reduce the amount of processing time and the volume of data to be processed. Data volume would be about a few megabytes to be processed every 500 milliseconds. With financial data, the volume is low, the arrival rate is high, and the processing is high as well. With a low computing latency, this results in a time window of a few milliseconds (Saxena & Gupta, 2017).

To reflect the reality of real time analytics, the critical aspects introducing latency to the total turnaround time are the time lapse between occurrences of an event to the time actionable insight is generated using it. There is some time lapsed in data travel from diverse geographical locations over the telecom channels to the processing hub. With the processing, first, due to security aspects they generally land on an edge node and are ingested into a cluster. Since it's big data, the data veracity is to be catered for, by eliminating the incorrect data before the actual processing. Then data messaging and enriching, binding, and enriching the transactional data with dimensional data occurs. Some time would be spent for the actual processing and storing of the results (Saxena & Gupta, 2017).

To build a scalable, sustainable, and robust real-time solution, a high level near real-time solution is proposed in the below figure. It contains a simple data collection funnel, the distributed processing engine, and some other components like cash, stable storage, and dashboard plugins.

Figure 15: Near Real-Time Solution Architect

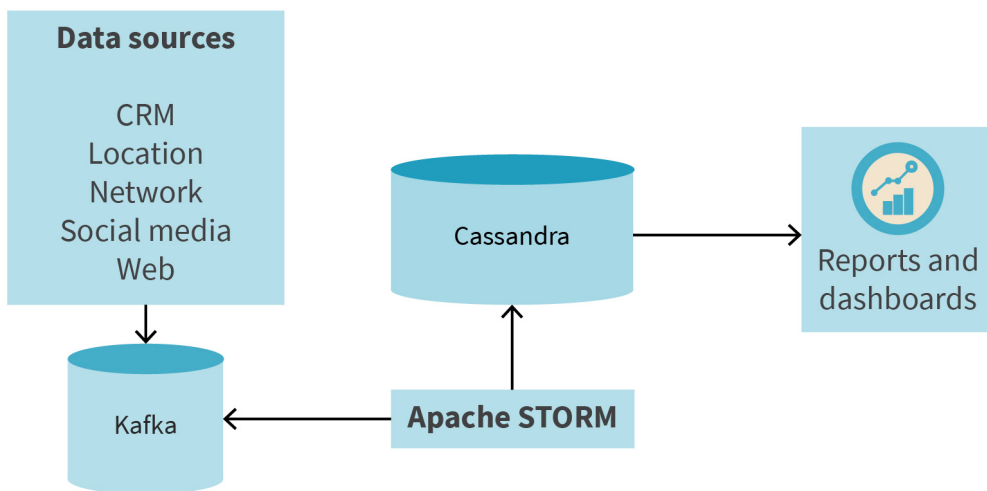


Source: Somayeh Aghanavasi (2024), based on Saxena & Gupta (2017).

The above figure illustrates the basic infrastructure of near-real-time processing; however, the process in the middle can include other solutions, such as distributed processing.

With the Storm solution, the high level streaming data in real-time is captured and routed it through some queuing systems, such as Kafka or RabbitMQ. Then the distributed processing is handheld through storm topology. After computing the insights, the results proceed to a fast write data store such as cassandra or Kafka for further real-time downstream processing.

Figure 16: Near Real-Time Processing: Storm



Source: Somayeh Aghanavasi (2024), based on Saxena & Gupta (2017).

Spark provides a very high-level solution with data flow pipelining, which is like Storm architecture. What's critical with the spark solution is that it leverages Hadoop Distributed File Systems (HDFSs) as a distributed storage layer. It also introduces Kafka to ensure decoupling into the system between the sources agents. The spark streaming component itself provides a distributed computing platform to process the data before writing out the results to unstable storage unit dashboard or Kafka queue.

However, high level solutions that are fast and appropriate are difficult to implement with simple systems. High performance hardware would be required. In the case of system failure, an overload of data is likely.



SUMMARY

The two common statistical tests used in data analysis and data mining are the t-test and the chi-square test. T-tests measure the significance of an effect to reject or approve a hypothesis. There are three kinds of t-test: (1) one sample t-tests, (2) two sample t-tests, and (3) paired tests. Chi-square tests are used to compare two sample sizes. The two concepts of linear regression and principal component analysis are covered. Linear regression, as an example from supervised learning, identifies the pattern by drawing a line over the observations finding out the trend. Giving the next set of new data the algorithm to find the next target values. Principal component analysis is used to reduce the dimensionality in datasets.

Operational data storage is an essential context in data warehousing. It provides a snapshot of the latest data from the transactional systems. It stores real-time data. It helps organizations to combine data from various sources into one single source. Operational is comparable to ETL process.

Event processing aims to identify the opportunities to react to the ones that occur out of normal situations. Five categories of event processing are observation, information dissemination, dynamic operational behavior, active diagnostics, and predictive processing.

Real-time processing includes the three main stages of data collection, processing, exploring, and visualizing data. The Storm solution with this construction as base provides distributed processing with use of Kafka or RabbitMQ queueing systems.

UNIT 6

WEB MINING

STUDY GOALS

On completion of this unit, you will be able to ...

- identify information retrieval models.
- evaluate methods in web content mining.
- describe web structures and how usage mining is performed.
- identify the purpose of web search spamdexing and differentiate content spam and link spam methods.
- present the purpose of a data lake and how it differs from a data warehouse.

6. WEB MINING

Introduction

The World Wide Web serves huge amounts of information, which is distributed throughout the world and provides global information in the form of news, advertisements, financials, education, and e-commerce. It is a dynamic environment containing vast amounts of information about hypertext structures, multimedia, hyperlink information, usage information, and access information. It also provides opportunities for such sources for data mining. To discover patterns, structures, and knowledge from the web, the application of data mining techniques is required (Han et al., 2012). This is called web mining. Web mining is organized in three main areas: (1) web content mining, (2) web structure mining, and (3) web usage mining. This unit will provide the meaning and concept of information retrieval. It will also discuss the following main topics: web search and spamdexing, access, and mining data lakes. Web search and spamdexing is about identifying the approaches taken to manipulate the search engine evaluation when indexing the pages to be shown in a list as output. Data lakes provide wide availability and scalability to data services, enabling access to various data types and formats.

6.1 Information Retrieval

The popularity of e-commerce is rising. This makes data mining a critical technology with the applications and the competencies it brings to online businesses. Enterprises expand as they grow. Together with their growth, the amount of information they provide on their websites grows as well. Their websites provide meaningful sources for exploring their activities, innovations, and business plans (Kasemsap, 2017).

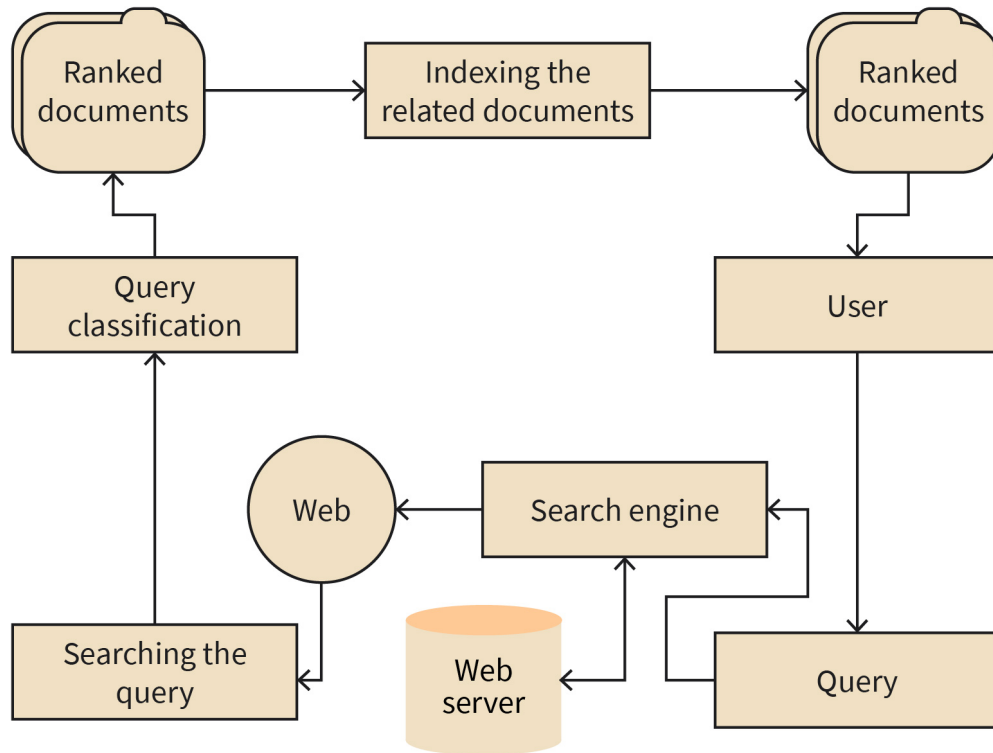
Information retrieval (IR) is the science of searching for documents and information in documents, through the metadata that describes data, and in the databases containing texts, images, and sounds (Luk, 2022). Retrieving data that satisfies requirements and constitutes important services is the goal of IR. IR is a fundamental component organizing the available information to guide and improve the retrieval applications dealing with complex data.

A challenging task to search the information is the increase of data volume on the web together with their heterogeneous characteristics. Therefore, when designing IR systems, they must be able to respond to the goals and to the domain knowledge of the users (Kasemsap, 2017).

Some traditional ranking models for IR systems can make a distinction between a relevant or a non-relevant document. The distinction is successful when both kinds of documents have similar representations with user's query (Kasemsap, 2017).

According to (Srinaganya & Sathiaseelan, 2015) the second kind of ranking model for IR is interactive IR. The general framework for IR can be explained in the figure below.

Figure 17: Interactive IR



Source: Somayeh Aghanavasi (2024), based on Srinaganya & Sathiaseelan (2015).

In a research study by Srinaganya & Sathiaseelan (2015), the authors focused on IR from web to rank and distinguished different documentations. They found that IR is one of the best techniques for web mining. The authors claim that “web query classification and web page classification are a leading major role to classify the web documents in IR using classification algorithms, based on the user query. The classified documents are indexed using ranking algorithm” (Srinaganya & Sathiaseelan, 2015, p. 5). They used machine learning (ML) algorithms, such as support vector machines, to achieve better accuracy and relevant documents from the web. The page rank algorithm was used to index the retrieval relevance of documents. This algorithm reflects the importance of incoming links from the page going out to other pages. Page rank algorithm is used often. To solve the complicated problems in relevancy and accuracy as well as ranking, the web documents would provide a promising design of ranking system (Srinaganya & Sathiaseelan, 2015).

IR system is a software system that provides access to books, journals, and other documents. It stores and manages those documents. An example is web search engines, one of most visible IR applications. General applications of IR are in searching media, whether images, blogs, videos, speech, or music.

The types of models for IR are categorized into two dimensions: (1) the mathematical basis and (2) the properties of the model. Mathematical basis models include some sub models such as set-theoretic, algebraic, probabilistic, and feature based models. In the set-theoretic model, documents are represented as sets of words or phrases. Set theoretic operations are applied on those sets to identify the similarities. The standard Boolean model, extended Boolean model, and fuzzy retrieval are instances of set theoretic models (Rehma et al., 2018).

The algebraic model represents the document and queries as vectors, matrices, or tuples. When applying algebraic models, the similarity of the query vector and document victory is identified and represented as a scalar value. Algebraic models include the generalized vector space model, the topic-based vector space model, the extended Boolean model, and latent semantic indexing/analysis (Singh & Singh, 2015). A probabilistic model works with the process of document retrieval. Using probabilistic inference, the probability of the similarities and relevance of the documents for a given query are computed. For instance, the probabilistic relevance model, uncertain inference, language models, and the binary independence model include probabilistic theories (Najar & Bouguila, 2022).

In the second dimension, the property of the model includes different interdependency inclusions in three general types.

1. A model without term-interdependencies treats different terms as independent. It represents a vector space model by assuming **orthogonality** of term vectors or, if it's in a probabilistic model, independency of term variables.
2. A model with immanent term interdependencies allows for the representation of interdependencies between terms. The degree of interdependency between two words is defined by the model.
3. A model with transcendent term interdependencies allows a representation of the interdependencies between words. The definition of the interdependency between the two terms relies on an external source not an internal one.

Orthogonality

In mathematics, for example, two lines are orthogonal if they are perpendicular. Change in one does not affect the other.

6.2 Web Content Mining

Web content mining is the process of data mining that automatically discovers and extracts information from the documents and services in web. The main purpose of web content mining is to find the pattern of usage for websites and retrieve useful information from their content and relations to other sources. It consists of several types of data, text, image, audio, and video. It can provide interesting and effective views about user needs. Text mining is used to scan documents and mine the text and images to retrieve useful information. ML and natural language processing (NLP) are related to text mining (Han et al., 2012).

One example of web content mining is analyzing social media data to understand consumer sentiment about a particular product or brand. This involves collecting large amounts of data from various social media platforms, such as X, Facebook, and Instagram, and then using NLP techniques to extract insights from the data.

For instance, a company might use web content mining to analyze customer reviews on their website, social media posts, and other online forums to gain insights into their customers' experiences and preferences. The data can be used to identify common themes and trends in customer feedback, such as product complaints or feature requests, and to develop targeted marketing campaigns or product improvements.

Another example of web content mining is analyzing web traffic data to optimize website performance. This involves collecting and analyzing data on user behavior, such as page views, click-through rates, and bounce rates, to identify areas of the website that need improvement. The data can be used to optimize website design, navigation, and content to improve user engagement and increase conversions. Common web search engines, such as Google and Yahoo, use web mining. The more mining the search engines provides, the more they can improve the identification of relative content.

Web pages can reside on two main levels, either on the surface web or on the deep web. Typically, search engines index the surface web. The deep web, also called the hidden web, refers to the content of the web that is not part of the surface web. The content from deep web is provided by connected databases (Han et al., 2012).

Mining information from the web can uncover the links between multiple web pages and other resources. It also provides a potential for inappropriate personal information disclosure. Therefore, studies on securing data mining sources reflect this concern to avoid any breaches of personal privacy during development of techniques (Han et al., 2012).

The degree of difference in data mining and web mining is the application, target users, access, structure, problem type, and tools. Starting in order, the application of data mining is in web page analysis, while web mining is applied to websites and e-services. The target users in data mining are data scientists and data engineers. Web mining works with data scientists along with data analysts. The access in data mining is considered based on access control definition, while web mining targets public data access. Data mining is more about clustering, classification, regression, and prediction. Web content mining investigates problems such as the content or the structure of the web to be mined. The skills that are used in web mining include application-level knowledge, data engineering, mathematics, statistical, and probabilities knowledge.

Web content mining investigates different forms of structured, unstructured, and semi-structured web pages. The approaches that are used in structured data extraction are wrapper generation and wrapper induction. In the first approach, the extraction program for each website is manually written, based on the pattern that is observed from the website. This approach can be time-consuming and difficult to scale for a large number of sites. The wrapper induction approach first labels the set of trained pages first. A learning system is used to generate the rules from the training pages. Then the rules that are resulted from learning apply to extract the target items from the web pages.



Research conducted by Brauner et al. (2022) analyzing web content mining e-scooter crash causes and implications in Germany. They collected 1936 crash related reports over two years based on a systematic web content mining process. The purpose of using web content mining was the lack of comprehensive and factual database analysis of e-scooter crashes and their causes. The work was in four major phases.

1. Resource finding, which included the publication of police reports.
2. Information selection and data cleaning, which included extracting URLs, removing duplicate URLs, excluding unrelated reports, **lemmatizing** using the report, and pre-processing the lemmatized text.
3. Generalizing, which included the vectorizing text and identifying the 400 most frequent words, selecting keywords, and creating an adjacency matrix based on those keywords.
4. A sentiment analysis of plain article texts, a network analysis based on adjacency matrix, and driving the clusters from network analysis.

Lemmatization
Identified by a lemma, this is a process of grouping inflected forms of words to analyze them as a single group.

Based on the results and findings, they could provide recommendations for driving e-scooters. In addition, they identify some causes of crashes, which provides ideas for securing the environment for e-scooters and drivers (Brauner et al., 2022).

6.3 Web Structure and Usage Mining

The structure of the web pages usually follows a hierarchical order: It starts from the main page, from which one navigates to the main sections depending on the web content and what's offered (e.g., information only, products, online shop). Web structure mining extracts patterns from hyperlinks, connecting a web page to other locations in a web page. Hyperlinks are the structural components of the web pages. Web structure mining is the process of using graph theories and methods to analyze the structure of the web and find out about the nodes and the connections. It can also be used to mine the structure of a document within a page. It's like analyzing the tree structure of page structures describing HyperText Markup Language (HTML) or Extensible Markup Language (XML) tags. The purpose of web structure meaning is to help one understand web contents and transform web contents into relatively structured datasets.

Web usage mining is the process of extracting user-related information, such as user click streams, from server logs. It finds patterns related to (1) users in general or particular groups and (2) users' searches, trends, and associations. Web usage mining also predicts what users are looking for on the internet. The information can be used to improve search efficiency and effectiveness. It can also help promoting products or related information to different groups of users at the right time. On a routine basis, web search companies conduct web usage mining to improve their quality of service (Han et al., 2012).

To discover user navigation patterns, the following methods may be employed. Association rule mining is one of the basic rules of data mining and is often used for web usage mining. The rules are in the form of statements such as $X \Rightarrow Y$. This statement means that if

transactions contain items in X, they may also include items in Y. Such association rules are used to find the relationships between the pages that appear frequently next to other pages in user sessions.

Sequential patterns are used to find out subsequences in sequential data of large volume. These sequential patterns are used to identify the user navigation patterns that appear frequently. They may seem to be like association rules, but they include the time. This means the sequence of events is of importance. The two types of algorithms that are used for sequential mining patterns are (1) the algorithm that is based on association rule mining (e.g., Apriori algorithms) and (2) sequential pattern mining algorithms (e.g., WAP-mine).

Clustering techniques can be used for the diagnosis of identifying groups of similar items. They use distance functions measuring the degree of similarity between the items. Using clustering algorithms in web usage mining involves identifying the groups with similar characters. Two types of clustering can be done: user clustering and page clustering. An example of cross selling methods are genetic algorithm and K-means algorithm.

In a research study performed by Svec et al. (2020), web usage mining was performed with empirical logits that analyzed the data of commercial banks and accesses of stakeholders to the selected part of the website. They modeled the time dependent behavior of the web user. The modeling of the probabilities of the access to web categories based on time was done using multinomial logit model. Nonhuman access was found to the web portal, which significantly influenced the obtained knowledge. Therefore, it was emphasized that the preprocessing phase in the process of web usage mining is critical and needs to be done appropriately so as to not affect the quality of the acquired knowledge (Svec et al., 2020).

6.4 Web Search and Spamdexing

Indexing in search engines is an approach by which one may organize information. Searching through every page for keywords and topics would make the process very slow for search engines to identify the relevant information. Therefore, search engines use an indexing approach as a system together with a process called tokenization to reduce the words and bring them to the essential meaning they contain. This is to make it faster for listing all known documents that could be relevant to the entered keywords and characters.

To determine the relevancy, ranking search engines use varieties of algorithms which rank pages. Some of the algorithms check whether the outcome is in the form of a web page, Uniform Resource Locator (URL), or text. As part of the process, search engine operators check for non-relevant items listed to be removed from the results. To affect the results of the search engines, for example, to appear in the topmost related items in search results, spamdexing was introduced in mid-90s (Marsden, 2018). There are two common spamdexing techniques: content spam and link spam.

Content Spam

Content spam uses a vector space model representing text documents as vectors of identifiers. Every technique in content spam involves altering the logical view that the search engine has about the content of the page. Here we mention the common techniques of content spamming.

One common technique is keyword stuffing, which involves the keywords defined on a page. The manipulation of the keywords affects the count, variety, and density of the page. It's to make it more likely to be found. For example, when a promoter wants to attract web viewers to a page, hidden text including the keywords representing a piece of popular music gets placed to mislead the viewers to particular but non-relevant content. Nowadays, most modern search engines can detect this by analyzing the pages for keyword stuffing, and they thereby determine if the page frequency is consistent with the other pages. Moreover, a similar technology called meta tag stuffing involves repeating keywords in the meta tags. This involves using meta keywords, keywords unrelated to the content. Similarly, some try to inject hidden or invisible text by disguising the text via using the same color as the background or hiding the text in the HTML code of the page. The HTML section such as “no frame” or “no script” has been used for this purpose.

Some other techniques are having a gateway or doorway page to be created with very little content and instead stuffed with very similar keywords and phrases to become attractive. They have no purpose for their visitors looking for the information and generally include a link such as “click here to enter” on the page which does the auto-forwarding for this purpose.

Article spinning or machine translation are the other two techniques. Article spinning is about rewriting the existing articles, in contrast to merely scraping content from the sites. This is to escape the penalties conducted by search engines when detecting duplicate content. Likewise, machine translation is used by some sites, which renders the content to different languages using no human editing. This results in text that does not make sense but still is indexed by search engines, thereby attracting traffic.

Online reviewing is becoming common. However, not all online reviews are truthful and trustworthy. A study used a novel classification approach using ML to distinguish spam from truthful reviews. They experimented with data from Yelp.com. They coupled multi-view learning with deep learning to associate both the reviews' content and the reviewers' behavior. The results achieve a high accuracy to detect spam reviewers (Andresini et al., 2022).

Link Spam

This method works on the links that are between pages. It works based on a method called link-based ranking algorithm. This method provides a higher rank to a website that is linked to other highly ranked websites. Another link-based ranking method is the hyper-link-induced topic search (HITS) algorithm.

For the same mentioned purpose (attracting viewers and appearing first in search results), a spammer can create multiple websites with different domain names and create a link between the websites to create a fake network of links. To increase the link popularity, spam users put hyperlinks where the visitors cannot see them. If a text link is highlighted, it can help the web page rank higher because of matching that specific phrase. This approach is used in Sybil attack.

Basically, websites that can be edited by users can also be edited by spam users. They insert a link to a spam site that remains until it is detected and eliminated. Comment spam as a form of link spam allows dynamic user editing (Wiki, blog) and can be problematic.

Nowadays, search engines use many new techniques to filter out spam pages and to provide the most realistic list of content related to the searched keywords. In a research study by Sangers and van Gijzen the second eigenvector of the Google matrix and its relation to link spamming was examined (2015). While the dominant eigenvector of the Google matrix determines the PageRank value, the second **eigenvector** was used to detect link spamming. They provided an algorithm to compute a complete set of independent eigenvectors for the second eigenvalue. The algorithm was to be used for the detection of link spamming (Sangers & van Gijzen, 2015).

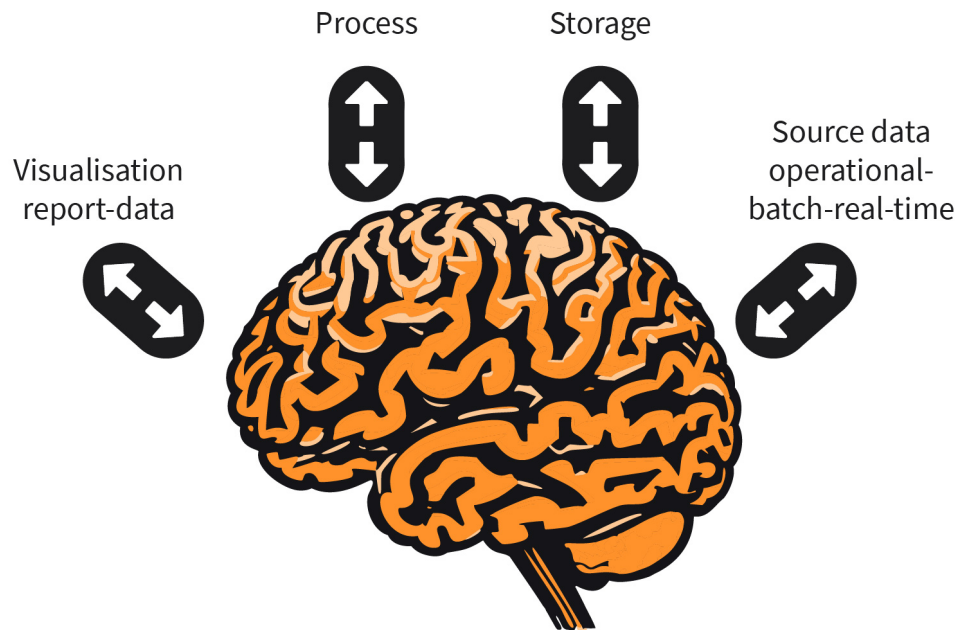
Eigenvector
This is a characteristic of a linear transformation. It's a nonzero vector changing by a scalar factor when linear transformation is applied.

To summarize, spamdexing is the unethical act of forcing a website to appear as a top search result. A webpage owner might now ask what legal and legitimate approaches to enhance their web page visibility are. Search engine optimization takes similar approaches, including (1) having a link-worthy site, (2) making sure the meta data is correct, (3) updating the content, and (4) using tags. But these approaches enhance the web page search by including the relevant content and information, rather than by injecting false texts. Being inline, correct, and legitimate makes a significant difference.

6.5 Access and Mine the Data Lake

What is a data lake? A data lake has been compared to a human brain (Dash, 2021). The brain receives input information from multiple different sources, processes the information, stores it in short- or long-term memory, and provides the output. Moreover, the brain stores the information that needs to be leveraged for access in future.

Figure 18: Data Lake Compared to a Brain



Source: Somayeh Aghanavesi (2024), based on Dash (2021).

A data lake receives different data sources including real-time, batch, or a mixture of both. The data could even be structured, unstructured, or semi-structured. Dash identified data lake architecture in two tiers. The first layer is the extract, transform, and load (ETL) from the source to the raw layer of data lake. The second layer uses ETL to move data from the previous layer to be consumed further, such as by moving the data to a data warehouse for reporting and analytics. The advantage of having this architecture is having multiple data sources residing in one place. Data would always be available from those multiple sources, which makes it easy to work with. The storage and computer are in different sections, which makes the processing faster. However, there could be some disadvantages to it too. Having multiple ETL tiers can make the process complex. Having these different tiers might need high maintenance. The state of the data wouldn't be clear, and data governance and data quality might face some challenges (Dash, 2021).

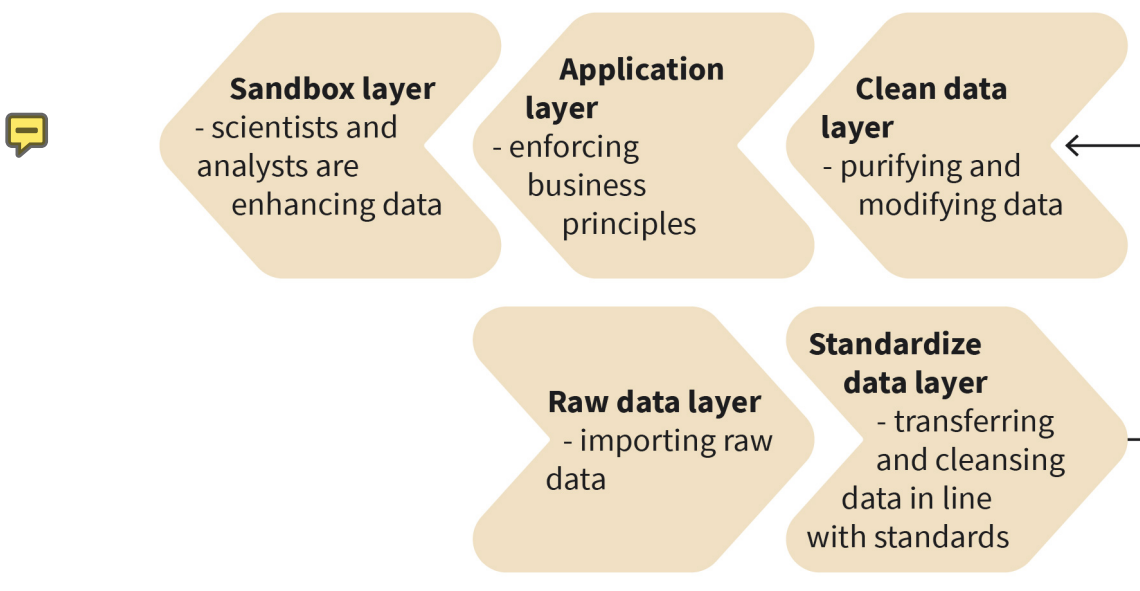
The difference between a data lake and data warehouse is that they are both widely used for storing big data, but they cannot be exchanged with each other. The data in the warehouse is clean and processed and ready for analysis based on the business needs. However, the purpose for a data lake is not clearly defined. It is a huge pool of raw data. Perhaps the ideal lake is for deep analysis. A data scientist who needs to perform advanced analytics on all available data may prefer to work with the internet. They can have access to all raw data and do multiple and iterative processes to analyze them. However, the data warehouse is ideal for operational users since it's well-structured and easy to use. In fact, the purpose of constructing a data lake is to have access to a wider range and variety of data while not limited to only structured data (as in data warehouse).

There are several important considerations in design and implementation of the data lake storage, as presented by (Han et al., 2012):

1. Data lakes are the centralized data repository for entire enterprise. The storage is supposed to be exceptionally scalable.
2. They are supposed to respond to wide ranges and varieties of queries and analytic tasks with data robustness.
3. Data lakes are supposed to address the diversity of data in enterprises supporting different types of data in various formats.
4. Data lakes support different kinds of queries, analysis, applications, and fixed data schemas. The independence to fix data schemas allows for maximum scalability in both data and applications.

Data lake layers are summarized in the figure below. It includes three mandatory layers of raw data layer, a cleansed data layer, and an application data layer. The optional layers of a standardized data layer and a sandbox data layer were also included (Han et al., 2012).

Figure 19: Data Lake Layers



Source: Somayeh Aghanavasi (2024), based on Han et al. (2012).

The main objective of the standardized layer is to facilitate high performance in data transferring and cleansing. Data scientists and data analysts have access to the raw data, and they can come up with experiments and new analysis which may lead to creating new datasets. The purpose of the sandbox data layer is to store the datasets created from those projects.

A data lake serving as a centralized data repository of the enterprise provides access to data scientists, analysts, and data engineers. Most of the accesses are to discover the datasets that are used for analytic purposes. search engine supports data discovery. To facilitate better data utilization in data lank, data models and dictionaries together with business rules are employed. They act as a domain for business knowledge base in the enterprise search engine, supporting search of datasets related to business.

There are many applications that could be built on top of the data services that are provided by data links. Application Programming Interfaces (APIs) are used for such a buildup. Moreover, some regular analytics and reporting services can be developed and maintained (Han et al., 2012). Amazon Web Services (AWS), Microsoft Azure, Google Cloud, Alibaba, and IBM provide solutions to enterprises establishing data links. As an example, AWS can provide fully managed services for high performance analytics pipeline. Amazon S3 buckets can be used for data landing, where heterogenous data sources can reside. It has related services, like AWS Cloudwatch, for monitoring and security (Cloud Experts, 2018).



SUMMARY

Information retrieval is the science of searching for documents and information in documents, the metadata that describes data, and databases containing texts, images, or sounds. It is a fundamental component organizing the available information to guide and improve the retrieval applications dealing with complex data.

Web content mining includes techniques that automatically discover and extract information from the documents and services on the web. Web mining finds out the pattern for usage of websites and retrieves useful information from their content and relations to other sources.

Web structure mining extracts patterns from hyperlinks, connecting a web page to other locations in a web page. Hyperlinks are the structural components of the web pages. Web structure mining is the process of using graph theories and methods to analyze the structure of the web and find out about the nodes and the connections. Web usage mining is the process of extracting user related information.

To determine the relevancy of the identified targets per search keyword, ranking search engines use a variety of algorithms which produce page rank. Spamdexing is an approach to affect the results of the search engines in order to appear in the topmost related items in search results. There are two common spamdexing techniques: content spam and link spam.

As part of web mining, it's essential to gain enough information about the data stores in web enabling access to data. In addition to data warehouses providing access to structured data, data lakes were introduced. A data lake has four essential characteristics enabling diverse and scala-

ble data access. Moreover, data lakes support different kinds of queries, analysis, and applications. It acts as a centralized repository to store data and provide multiple level data access to a variety of users.

UNIT 7

DATA ECONOMY

STUDY GOALS

On completion of this unit, you will be able to ...

- identify types and levels of data aggregations.
- understand what data monetization is.
- describe Internet of Things components and the journey data take from being collected to being monetized.
- describe the history of Industry 4.0 and how data play as role in 21st century innovation.
- present the importance of big data and their characteristics in data economy.

7. DATA ECONOMY

Introduction

Alongside land, capital, labor, and oil, data have become an important factor in the contemporary world. It's the key component of artificial intelligence (AI), the application of which is a powers innovations from self-driving cars and drug personalization to ad targeting and credit provisioning. Data support the economy from various directions. They are used for efficiency and factful decision-making. Using data and analyzing them helps corporations and businesses design and develop new products, services, and processes.

The data economy is connected to some topics in data mining. This unit gathers the information related to data economy to provide a clear view about those topics and their roles in the data economy. The first topic is data producers and aggregators, which will inform you concerning how and at what levels the data are aggregated.



Then the focus goes to the topic of data monetization to provide the definitions and commons senses of it. This unit includes the topics of the Internet of Things (IoT), data mining in Industry 4.0, and big data. These topics provide an overview of the current status of the Industry 4.0 and how significant data is in today's economy. Moreover, the relevant challenges are also discussed.

7.1 Data Producers and Aggregators

Businesses consequently and frequently gather a large amount of data to get critical insights about their customers and other important matters. In the case of providing a product or service, aggregated data can provide statistics about customers, their behaviors, and the trends that are important to be aware of for them. For this purpose, the aggregation of data plays an important role. As an example, a telecom company's aggregation of data about the number of purchases, subscriptions, or the location of customers offers valuable information relevant to decisions about the next stages in business and product offerings (Cloud Experts, 2018).

The data producer is accountable for collecting and processing data, identifying the data domain, storing and monitoring data, and ensuring data quality. Data aggregation is the act of compiling the information that is fetched from databases. The intention for data aggregation is to prepare a combination of datasets for the processing of data. For example, the collected raw data can be aggregated over a given time to provide statistics such as mean, minimum, maximum, sum, and count (Cloud Experts, 2018).

Levels of Data Aggregation

The time interval of the data aggregation can be based on (1) the reporting period, (2) granularity, and (3) the polling period. The first-time interval is the period in which the data are collected for presentation. It can be raw data. For example, the data can be collected and processed into a summarized format and stored in a database. This data can be collected in one day from a networking device like a wireless access point. The aggregation of the database on granularity refers to the period in which the data are collected for aggregation. As an example, assume that some of the points for a specific product or customer are collected for a period of one hour. At this point, the granularity would be one hour. However, the value of the granularity can vary from an hour to years depending on the reporting (Han et al., 2012).

Aggregating data based on polling period refers to the frequency in which the resources or customers are sampled. For example, imagine a situation where you are collecting data about customers polled every hour. The period for polling and granularity would be under what's called spatial aggregation.

The application of data aggregation is useful for analyzing data over time. Let's say for a business the campaign is about to run for a particular time or a particular cohort or a particular platform. This can be done in three phases: extract, transform, and visualize.

In retail and e-commerce industries, data aggregation plays an important role. An instance is the monitoring of prices. The company is supposed to collect details of the pricing, offers, and campaigns of their competitors and other companies to be able to perform comparative analysis. Referring to the competitor's website, the aggregation of data can obtain desired data.

Likewise, in travel industries, data aggregation has a significant impact. Travel companies need to perform research about their competitors to get insights into their approaches to marketing, the status of their customers, and their relations to the other industries affecting their businesses. Capturing an image from travel websites, including the customer sentiments, helps identify the satisfaction level. Again, the aggregation of the collected data can lead to valuable insights and business values.

With the given illustrations, the business analysis requires careful data aggregation and should provide a summary of the data. The statistics of the data help correct false information and provide products or services which can lead to higher values and customer satisfaction.

Web Data Integration

Web data integration is crucial for aggregating data accessed from websites. It involves a process that manages data collected from different websites, generating a standardized workflow. The process includes accessing the data, transforming them into suitable formats, mapping it appropriately, and ensuring their quality before generating valuable business insights.

However, integrating web data presents challenges. Evaluating data involves measuring and monitoring progress, which may require costly expert assessments that are often inaccessible. Another challenge is dealing with hybrid data, which combine textual and structured data. When searching for data, both forms must be considered, including any structured elements within the query.

Additionally, the heterogeneity of web data exists at both the schema and data levels. When using data from different sources simultaneously, it is crucial to ensure that overlapping data do not affect the results. Care must be taken to avoid counting redundant records twice or analyzing them incorrectly when summarizing data and counting records (Herzig, 2014).

In SQL, functions like `count()`, `sum()`, `min()`, `max()`, and `avg()` are used for data aggregation. For example, assuming access to a data table named “survey,” the following command can be written to count and sum the records with a condition (Carpenter, 2010).

```
SELECT COUNT(*), SUM(weight)
FROM survey
Where weight > 1.5
GROUP BY Product_id
Order by count(Product_id);
```

where `*` is the symbol to collect all data records, the command “group by” aggregates the data per product identification number, and the command “order by” orders the records based on the identification number in ascending order (by default; Carpenter, 2010).

7.2 Data Monetization

Data monetization refers to the process of earning and benefiting from the sale or use of stored data. Data within the business ecosystem holds value that can be leveraged. Effective data monetization should be driven by the dynamics of a functioning business model. Therefore, data should not only possess inherent value in its passive historical context but also possess dynamic value derived from analysis. The dynamics of the data, subject to verification and monetization, provide additional value (Jabłoński & Jabłoński, 2020).

Data monetization can be categorized as internal or external. Internal monetization involves utilizing data to extract value from customers, such as tailoring offers based on customer/user behavior. On the other hand, external monetization refers to collaboration between businesses to achieve synergy effects through the sharing and utilization of data (Opher et al., 2016).

Monetization strategies can vary based on the unique needs and characteristics of each business. There is no one-size-fits-all approach. To derive value from business data, specific tools and methods need to be employed. Some existing monetization strategies include marketing, finance, sales, and revenue management. Each of these areas relies on

analysis to support decision-making. However, integrating these existing strategies and continuously developing new ones into scalable analytical solutions can pose a challenge for organizations.

A real-world example from Unilever illustrates this point. Unilever has a team called Consumer and Market Insights (CMI) that focuses on building monetization strategies for different business units. They have developed analytic solutions such as Growth Scout and Growth Cockpit to drive better decision-making through monetization strategies. These solutions have a broad impact on decision-making across the organization and deliver economic benefits through the actions they recommend. The following excerpt is taken from the article by Chiang and Wells (2017).

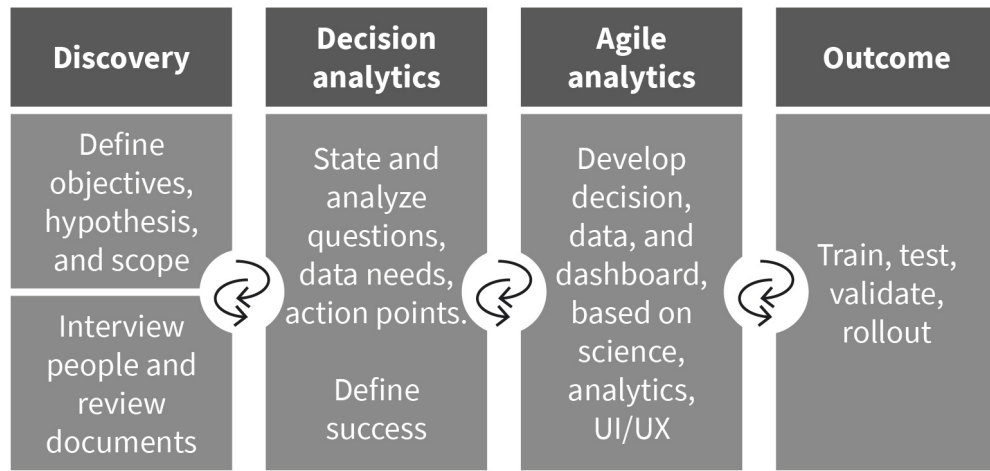
A typical example is assessing the impact of increasing the market share of shower gels in Thailand by 10%. This information could help Unilever identify growth opportunities and determine where to invest in marketing or product development. Unilever's CMI home-care team has recently used Growth Scout to find potential profitable markets for their detergent brands by identifying demographic groups with low market penetration. CMI also utilizes other tools to answer questions about product benefits, effective advertisements, budget allocation, and optimal pricing. CMI plays a key role in tracking marketing performance and advising on necessary adjustments. Once decisions have been made on market strategy, Growth Cockpit, a custom-built software, provides a comprehensive overview of a brand's performance relative to the market. This tool quickly displays important metrics such as market share, penetration, pricing, and media spending, helping managers identify growth opportunities (Chiang & Wells, 2017).

The above excerpt provides good examples of how tools can effectively drive decision-making. In these cases, the growth scout or growth cockpit assist business managers in making higher-quality decisions by implementing monetization strategies. The aim is to equip authorities with the tools needed to develop strategies for companies (Chiang & Wells, 2017).

Architecting Decisions

Consider the following statement: “Companies in the top third of their industry in the use of data-driven decision making were, on average, 5% more productive and 6% more profitable than their competitors.” (McAfee & Brynjolfsson, 2012, p. 1). This statement suggests that data-driven decisions should prioritize processes that promote data monetization and data-driven decision making. Focusing solely on the data alone would miss the driving force behind effective action, which directly affects the quality of decisions. Factors such as understanding the interdependencies, defining ownership, determining risk tolerances, and possessing the necessary skills are essential for putting performance at the forefront of operations. The decision architecture methodology, shown in the figure below, includes decision analysis in the second phase. It's important to note that there are iterative cycles between decision analysis, agile analytics, and outcome, with ongoing back and forth movement between these phases.

Figure 20: Decision Architecture Methodology



Source: Somayeh Aghanavesi (2024), based on Chiang & Wells, (2017).

According to Chiang and Wells (2017), there are four components for developing a monetization strategy: monitorization guiding principles, competitive and market information, business levers, and requirements gathered from decision analysis. Depending on the specific strategy being developed, these components may be developed to different degrees. The figure below provides an overview of the high-level picture of a monetization strategy and how the components fit together.

Table 9: Monetization Strategy Aspects

Monetization Strategy			
Decision analysis	Market information	Business levers	Monetization
	Industry competitors, market trends		

Source: Somayeh Aghanavesi (2024), based on Chiang & Wells, (2017).

Business levers are actions taken to address specific business challenges, with the aim of achieving faster results. It is similar to using a physical lever to move large objects with less force.

The goal of developing a monetization strategy is to help managers generate revenue and reduce costs. To do this, it is important to understand what business levers are available and which tools can be used. For example, if the goal is to optimize asset utilization in a company with few assets, it may not be appropriate to use the same method as for a company with a large number of assets.

A good starting point for developing monetization strategies is to analyze the company's profit and loss statement. This helps identify the factors driving growth, costs, and ultimately profit. This information is then used to determine which business levers can be used (Chiang & Wells, 2017).

Data visualization is a powerful tool that aids in understanding the bigger picture. It should be incorporated early in the decision analysis phase to present information visually. Effective data visualization requires special modeling and structuring techniques (Chiang & Wells, 2017).

7.3 Internet of Things

In today's digital world, data are as valuable as physical assets like property, equipment, inventory, and cash for companies. Companies collect, analyze, and report large amounts of data, which have become a crucial measure for survival in the digital revolution. Data can come from various sources, and their collection and analysis are improving thanks to IoT devices. These devices enable the exchange of more data within and between companies, making data a key factor in their financials.

An example of how data can be utilized in the data economy is through an IoT wearable device. This device collects data such as temperature, sleeping patterns, sleeping orientation, and breathing data from patients at rest. These data are collected by a device worn by the patient. Different parties can then use this data for their benefit. For instance, the device manufacturer can monitor the data to identify anomalies, provide diagnostics, and sell the data to drugstores for medicine recommendations. In another use case, the device manufacturer can aggregate the data from the device and sell insights to retail stores that specialize in beds or clothing.

A framework has been developed for the data economy to describe companies and their roles, as well as their capabilities in the data economy (Opher et al., 2016). Data presenters use user interface and user experience to explore and understand user engagement. Insight providers primarily utilize statistical and computational methods, such as semantic models, and develop algorithms and logical rules using machine learning (ML) in analytics libraries. Platform owners create environments using APIs for connectivity, leveraging clouds for their apps and device discovery. Data aggregators and data custodians ensure common transmission and collect heterogeneous data from various devices. They achieve this by accessing, controlling, and collecting data through IoT and big data-focused solutions (Opher et al., 2016).

The figure shows that data producers collect data from both IoT and traditional big data sources. IoT data sources include embedded chips, sensors, wearables, mobile phones, accelerators, and gyroscopes (Opher et al., 2016). Companies have various options in the data economy framework. They can move horizontally across different layers, vertically up or down the stack, or expand or compress the stack.

Moving across the layers allows companies to enhance their current capabilities by adding more material and content to a specific layer. For instance, a company producing fitness trackers can incorporate data on steps and heart rate to their existing sleep monitoring feature. Moving up or down the stack enables companies to take on new roles and capabilities in different layers, leading to further differentiation. This can involve closer consumer interactions or improved interoperability.

The process of compressing and expanding the stack involves moving between different layers, such as data presentation and data production, to enhance data offerings. For instance, companies such as Google and Amazon have created platforms that simplify the search and purchase of products. They use data to optimize the user experience and, as a result, people are more likely to revisit these platforms, often driven by factors like product reviews. These platforms also allow smaller companies to participate and engage with customers, generating additional data and maintaining platform relevance (Opher et al., 2016).

IoT helps companies' financial performance into main ways: cost containment and value attainment. Cost containment does the following:

- reduces building optimization energy usage
- lowers maintenance costs by balancing resource costs, idle time, breakdowns, and management costs
- improves asset utilization by monitoring the equipment
- reduces manufacturing defects and design flaws by embedding sensor data to conduct failure pattern and failure risk analysis
- improves safety by synthesizing equipment, personal, and location data
- increases workforce productivity through new automation improved visibility and early identification of anomalies via usage and excursion analysis

In contrast, value attainment involves the following:

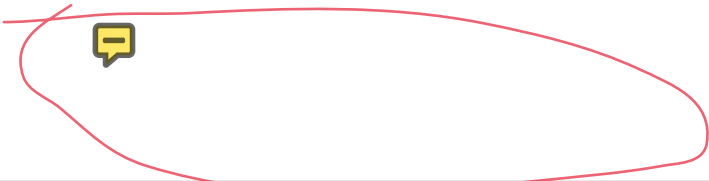
- data monetization via selling company data or licensing **taxonomies**
- product placement, store layouts, marketing, improve customer experience with new data and analytics
- pay-per-use/as-a-service pricing by collecting usage data
- increasing product margins by integrating data into processes
- enhancing products and services by analyzing data stored from purchases and products
- identifying new applications for current and existing products and services

Although there are billions of datasets and sources of IoT data collected worldwide, IoT data can be categorized into several main types. These include location, environment, machine, living, event, attribute, motion, and orientation (Opher et al., 2016). The table below describes each type.

Taxonomy

This is the practice of categorization and classification.

Table 10: IoT data types

Data type	Description
Location	<p>Data that is geographical position based on Global Positioning System (GPS), Wi-Fi, and beacon. Example:</p> <ol style="list-style-type: none"> 1. GPS coordinates of a vehicle for fleet management. 2. Wi-Fi hotspot locations for location-based marketing. 3. Beacon proximity data for in-store customer tracking.
Environment	<p>Data measurements and states of environmental variables. Example:</p> <ol style="list-style-type: none"> 1. Temperature and humidity levels inside a greenhouse for smart agriculture. 2. Air quality data in urban areas for pollution monitoring. 3. Water level and flow rate data in rivers for flood forecasting.
Machine	<p>Automatic data created from computers, equipment, and machines without intervention of human. Example:</p> <ol style="list-style-type: none"> 1. Operating temperature data of a manufacturing machine for predictive maintenance. 2. Energy consumption data of a building for optimizing energy usage. 3. Server log data for monitoring system performance and identifying errors.
Living	<p>Data collected by sensors monitoring vital states such as blood, heart rate, temperature. Example:</p> <ol style="list-style-type: none"> 1. Heart rate variability data for remote patient monitoring. 2. Blood glucose levels data for diabetes management. 3. Body temperature data for fever monitoring.
Event	<p>A data point at which an event/occurrence is transpired. Example:</p> <ol style="list-style-type: none"> 1. Timestamped data of a vehicle collision for insurance claim processing. 2. Timestamped data of a stock trade for financial analysis. 3. Timestamped data of a customer transaction for fraud detection.
Attribute	<p>Characteristics of an object that can be categorized and/or counted. Example:</p> <ol style="list-style-type: none"> 1. Product stock keeping unit (SKU) data for inventory management 2. Object weight data for logistics tracking 3. Customer demographic data for targeted marketing. 
Motion	<p>An object or human movement. Example:</p> <ol style="list-style-type: none"> 1. Accelerometer data from a wearable device for fitness tracking 2. Pedestrian traffic data for urban planning and design 3. Vehicle traffic data for traffic management
Orientation	<p>Relative and rotational movement of an object. Example:</p> <ol style="list-style-type: none"> 1. Gyroscope data from a drone for flight stabilization 2. Rotational speed data from a wind turbine for performance optimization 3. Orientation data from a robotic arm for precise positioning.

stock keeping unit
 This is a unique identifier code assigned to each product or item in a company's inventory, which helps to track and manage inventory levels, monitor sales, and simplify ordering and restocking processes.

Source: Somayeh Aghanavasi (2024) based on Opher et al. (2016).

7.4 Data Mining in Industry 4.0

During the second industrial revolution, significant advancements in production were made through the use of assembly lines and electricity. Prior to that, in the 18th century, the First Industrial Revolution occurred, which utilized water and steam power to transform manufacturing and society. The manufacturing revolution then progressed to the third generation, which incorporated computers and automation.

Currently, we are experiencing the Fourth Industrial Revolution, also known as 4IR or Industry 4.0. This revolution is characterized by rapid technological changes, impacting various industries and societal patterns in the 21st century. Industry 4.0 aims to introduce machine interconnectivity, automated decision-making processes, and improved data analytics to enhance productivity and efficiency. As of 2023, some of the top trends in industry 4.0 include the IoT, advancements in three-dimensional (3D) printing, improved efficiency in 5G technology, enhancements in AI, better utilization of cloud technology, and optimized production methods with reduced energy consumption.

Data science involves using and understanding data and requires the expertise of mathematicians, statisticians, and computer scientists. Data providers and data users need to work together and have a clear understanding of how to effectively use the data. In Industry 4.0, information technology (IT) skills, business awareness of data's role, and data analytics using math, statistics, ML, AI, and deep learning are key to adding value. ML techniques have been around for a while, but the availability of large amounts of data has made their use more prevalent. Although AI and ML may sometimes be seen as a black box, businesses can benefit from a clearer understanding of their capabilities, which requires knowledge of statistics and mathematics. Mathematics and statistics have much to offer to data science in Industry 4.0 and should be taken seriously as a growing field (Coleman, 2019).

One major challenge in Industry 4.0 is the lack of data sharing between different platforms and departments, also known as siloed data. For effective business intelligence and analytics, every department must share its data and insights. Otherwise, the business may face obstacles in progressing in the right direction. Another related challenge is data system redundancy, where having multiple enterprise planning systems makes it difficult to utilize data analytics and reporting effectively. It is important to have a centralized repository of enterprise data to avoid this redundancy. In addition, talent shortages, security and data access issues, and compatibility problems between digital tools hinder smooth data flow and record exchange (Information Resources Management Association, 2021).

7.5 Big Data

In this section, we will discuss how big data impact the economy. Big data play a significant role in increasing economic activity and living standards by greatly impacting our knowledge of the world. According to the McKinsey Global Institute, big data have the

potential to generate an additional \$3 trillion in value each year across seven industries. This value would primarily benefit customers by reducing traffic jams, improving price comparisons, and enhancing search capabilities (Kennedy, 2022).

Big data will affect the economy in several ways. Firstly, they will optimize business processes. Secondly, they will enable targeted marketing, allowing customer feedback to shape product design. Thirdly, they will contribute to better business management. And finally, big data will facilitate the production of new products and services (Kennedy, 2022).

In the field of sustainable development, researchers have been studying the quality of big data. In a previous study, the researchers reviewed existing literature to find evidence of how cyber-physical production systems influence social sustainability performance through technology. They found 119 sources and recommended future research to investigate if Industry 4.0-based manufacturing technologies can ensure sustainability in big data-driven production systems by using IoT sensing networks and deep learning-assisted smart process planning (Andronie et al., 2021).

Another study, conducted by Wamba et al. (2018), aimed to understand the dynamics of information quality in a big data environment and how it relates to business value, user satisfaction, and firm performance. The researchers collected data from 302 business analysts in France and the United States of America (U.S.). From their analysis, they found that information quality in big data analytics has significant and positive impacts on (1) completeness, (2) currency, (3) format, and (4) accuracy.

It is worth mentioning that around 2012, privacy regulations underwent significant decision-making processes in response to emerging technologies like big data. While data usage brought about various innovations, efficiency, and productivity improvements, concerns about privacy started to arise. The concept of personal privacy, including limited access to personal information and control over one's data, faced a major challenge due to the advent of big data. As a result, regulations such as the General Data Protection Regulation (GDPR) were introduced. GDPR requires end-users to give consent for their data to be accessed, stored, and presented.



SUMMARY

Data aggregation involves compiling information from databases to combine datasets for data processing. This can include aggregating raw data over a specific period to calculate statistics such as mean, minimum, maximum, sum, and count. The time interval for data aggregation can be determined by the reporting period, granularity, and polling period. Data integration, on the other hand, involves the process of aggregating and managing data collected from different websites, particularly for web data integration purposes.

Data monetization should result from a dynamic business model, where data are not just attractive in its passive historical context but are also analyzed to generate insights. Developing a monetization strategy aims to assist managers in driving revenue and reducing costs, which involves identifying the business levers and understanding the available tools.

In the context of data collection and analysis, the IoT plays a crucial role in the data economy and the value derived from data usage. IoT data sources include embedded chips, sensors, wearables, mobile phones, accelerometers, and gyroscopes. The use of IoT data can impact a company's value in two main ways: cost containment and value attainment. The utilization of big data, which significantly impact global knowledge, can lead to increased economic activity and living standards.

Industry 4.0 refers to the period of significant technological advancements, changes to industries, and shifts in societal patterns and processes in the 21st century. This transformation is driven by smart automation and improved connectivity. The objective of Industry 4.0 is to enable machine interconnectivity, automate decision-making processes, and enhance data analytics. Ultimately, the aim is to boost productivity, efficiency, and value.

UNIT 8



LEGAL REGULATIONS AND USAGE POLICIES

STUDY GOALS

On completion of this unit, you will be able to ...

- understand the necessity of applying the General Data Protection Regulation (GDPR).
- identify what personal data and information are.
- identify the legal grounds for personal data processing.
- describe the data processing model and the transparency needed to communicate between the main entities in the model.
- evaluate compliance with GDPR.

8. LEGAL REGULATIONS AND USAGE POLICIES

Introduction

The protection of data is an important issue. Some laws and regulations protect users' data, known as the General Data Protection Regulation (GDPR). It's important to understand the basic elements of GDPR, its principles, and the parties involved in it. Understanding the definition of personal data and what it includes is essential to comply with GDPR. Additionally, every data controller and processor must comply with the rules of GDPR, which include the legal basis for data processing. The data protection model outlines the main entities involved and their interactions, as well as their responsibilities to the Information Commissioner's Office. This unit also covers compliance with copyright laws and the rights of owners if these laws are violated.

8.1 General Data Protection Regulation

GDPR is a significant step in protecting the rights of EU data subjects. It has had a profound impact on industries, shifting from a process-driven approach to a risk-based one. GDPR has also influenced data privacy laws worldwide since 2016, not just in the ~~European Union~~ (EU). Its effects have been felt in technology-reliant fields such as investigations, application development, eDiscovery, and emerging technologies like artificial intelligence (AI), machine learning (ML), and blockchain (Taal, 2021).

GDPR consists of four fundamental building blocks: principals, individual rights, responsibilities, and oversight and enforcement. These components aim to ensure appropriate handling of personal data (Reichel et al., 2021).

Two essential elements of personal data processing are location and duration. Location refers to where the data is processed, which may differ depending on whether it is stored within the EU or in another location, such as the United States of America (U.S.), where relevant laws may also apply. Duration pertains to how long an organization can store a data subject's information. Data cannot be kept indefinitely, and generally, it should only be stored for as long as necessary for a specific purpose. Some organizations inform data subjects about the duration of data storage and whether they can request longer retention periods.

8.2 Personal Information

According to the regulation, “personal data” refers to any information that relates to an identified or identifiable natural person (referred to as the “data subject”). An identifiable person is someone who can be directly or indirectly identified, based on factors like an identification number or personal characteristics such as physical, physiological, mental, economic, cultural, or social identity (Taal, 2021).

The concept of personal data is broad and encompasses information in various forms. Even work-related data can fall under the category of personal data if they relate to or impact an individual, including opinions expressed, internal documents, or emails concerning a specific person.

Some examples of personal data include the following (Quinn, 2021):

- name, address, email, telephone
- age, gender, marital status
- identification (e.g., driving license or registration number)
- national identity or tax number
- passport number
- car registration number
- photographs
- video (e.g., CCTV)
- voice
- fingerprint, facial recognition images
- travel cards or tickets
- education and training information
- student numbers
- grades, exam results, certificates
- **testimonials**, references
- health information, medical reports
- family and lifestyle details
- financial details or bank statements
- card numbers
- online identifiers, Internet Protocol (IP) addresses
- cookie identifiers, radio frequency identification (RFID) tags

Testimonials

These are statements or comments from customers, clients, or users about their experiences, satisfaction, or feedback regarding a product, service, or company.

According to Quinn (2021), personal data can be categorized into factors that relate to different aspects of a natural person's identity. Some examples include:


- physical factors: This includes photographs that may reveal information about a person's health or appearance.
- genetic factors: This refers to records of a subject's DNA profile.
- mental factors: This pertains to records of a subject's mental health or individual abilities.
- cultural factors: This encompasses the ethnicity or cultural background of the person.
- social factors: This relates to the socioeconomic class of the subject.

In terms of identified personal data, data subjects have the right to erase their data and to prevent their processing if certain conditions are met. These rights are outlined by Quinn (2021):

- withdrawal of consent by the data subject
- objection to data processing based on the organization's legitimate interest or public interest. In cases where there is no overriding legitimate interest and direct marketing or profiling is based on legitimate interest, the individual has the right to object.
- collection of personal data in relation to information society services provided to a child with parental consent. The personal data were collected in relation to the information society services offered to a child when the consent was given by parents (Quinn, 2021).


8.3 Legal Basis for Data Processing

When an organization wants to process personal data, it needs to have a valid legal reason for doing so. In other words, they must explain why they are processing personal data. There are several types of legal reasons, including the following:

- 
- getting consent from the individual for a specific use of their personal data
 - having a contract between the organization and the individual that requires the processing of personal data
 - fulfilling a legal obligation that the organization has
 - protecting the individual's life in a dangerous situation
 - having a public task or duty that requires the processing of personal data
 - having a legitimate interest in processing personal data

To ensure that the data processing is lawful, the organization must have at least one of these legal reasons for each specific process. The legal reason cannot be chosen simply for convenience – it must be specific to each processing activity.

The most flexible legal reason for collecting and processing data is a legitimate interest. This means that the organization has a valid reason for processing the data that is not based on a specific purpose or consent. According to the United Kingdom's (U.K.'s) Information Commissioner's Office (ICO), a legitimate interest is a legal reason that is not focused on a particular purpose and does not require consent (Taal, 2021).

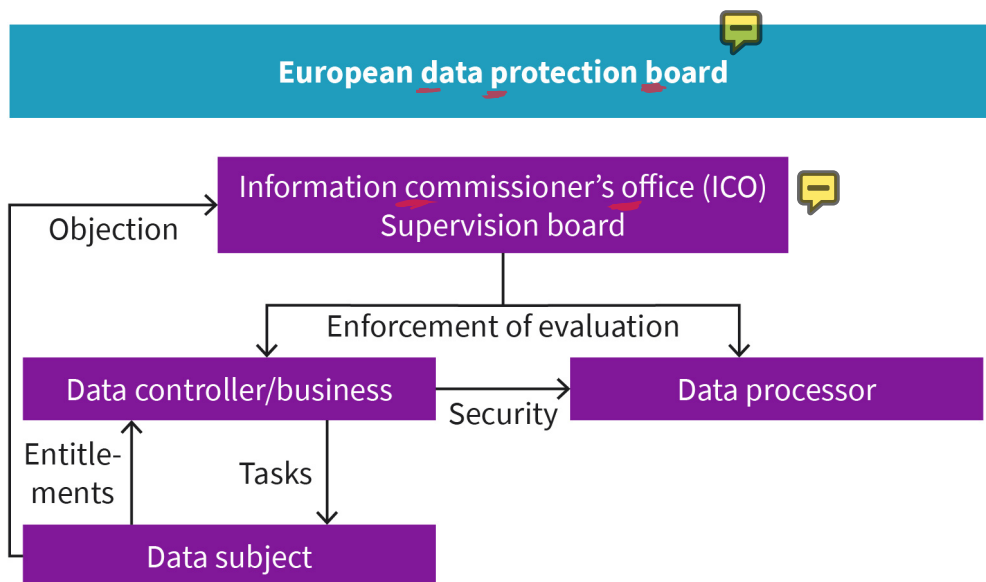


8.4 Data Protection and Transparency

Data protection, private data protection, and privacy are closely related terms that are often used interchangeably in discussions about protecting data and personal information. GDPR establishes legal requirements regarding the right to data protection and how it relates to other rights. According to GDPR, any processing of personal data by employers must always comply with the law.

The diagram below illustrates a general data protection model based on GDPR. Each EU member state has designated a supervisory authority responsible for overseeing data protection. In the U.K., the designated authority is the ICO (Taal, 2021).

Figure 21: Data Protection Model



Source: Somayeh Aghanavesi (2024), based on Taal, (2021).

The data protection model includes a board that represents its components. The ICO is the supervisory authority responsible for enforcing assessments of data controllers. Data controllers, which are businesses, have responsibilities towards data subjects and must comply with their rights. They are also required to ensure the security of data processes. Third parties are responsible for informing subjects about the disclosure of their data, and data subjects themselves must comply with the ICO.

In Germany, the supervisory authority is called Die Bundesbeauftragte für den Datenschutz und die Informationsfreiheit. In Belgium, it is the Commissions de la Protection de la Vie Privée. France has the commissions Nationale de l'Informatique et des Libertés (CNIL), and Ireland has the Data Protection Commissioner. There are 27 EU member states and the U.K. plans to remain even after withdrawing from the EU (Taal, 2021).

Accountability and transparency are often connected because transparency is seen as a way to provide accountability and demonstrate that things are being done diligently. However, transparency is not a one-dimensional concept. It encompasses various expectations and challenges. Transparency includes the idea of verifiability, where someone can verify the truthfulness and accuracy of information, as well as the ability to explain and inspect. Having a complete understanding of transparency enables accountability and auditability.

To achieve accountability, organizations need to be transparent. Transparency is both a social and organizational concept. It involves showing how data are processed, but the level of transparency can vary. The opposite of transparency is opaqueness, which occurs when there is an overwhelming amount of information that users cannot effectively analyze. This presents a challenge between transparency as information and transparency as performance.

Transparency also encompasses the ability of a system to interact with other systems and for individuals and society to engage with those systems. This can include interoperability and engagement. (Hallinan et al., 2021)

Transparency is influenced by legal, regulatory, and organizational factors. In the legal context, GDPR mandates transparency as a fundamental principle. However, the focus is primarily on providing information, with little guidance on how developers should implement transparency requirements. Some authors have proposed a “transparency by design” approach to establish guiding principles.

8.5 Copyright Compliance

Quinn (2021) suggests using the following applications for data controllers or processors to comply with GDPR. It is the enterprise’s responsibility to implement operational controls for these processes (Quinn, 2021):

- to have a lawful basis for each processing activity
- to be able to demonstrate compliance with the six principles and accountability
- to ensure processing complies with the GDPR and the security is appropriate to the risks, improvement technical and organizational measures
- to maintain a record of data processing activities
- to notify certain types of data breaches to the supervisory authority and to data subjects
- to appoint a data protection officer
- to restrict transfers of personal data outside the **European Economic Area (EEA)**
- to include mandatory terms in data processor contracts
- to document legitimate interest
- to capture and document GDPR-level consent
- to manage third-country transfers through encrypted transfers
- to assess specific areas of the enterprise compliance (e.g., marketing, CCTV, employee contracts, and children’s data)
- to manage contracts with processors and agreements together with other joint controllers
- to manage data retention policies and procedures

As part of compliance requirements, staff needs to receive training on data protection, including the company’s policies and procedures, as well as processing, technical, and organizational controls. These policies and procedures cover areas such as cybersecurity, information security, and physical security risks.

European Economic Area

This is a trade agreement between the EU and three European Free Trade Association (EFTA) member states: Norway, Iceland, and Liechtenstein. The EEA agreement allows these EFTA countries to participate in the EU’s single market without being full members of the EU.



Copyright compliance has been a topic of concern for many years, as it aims to protect the rights of individuals and organizations who create original works. Every business should have a comprehensive policy in place for copyright compliance, including specific procedures for obtaining permission to use copyrighted material that aligns with the needs of the business.

Reducing the risk of copyright infringement involves fostering a culture of copyright compliance within the business. It is important to monitor all business products to determine if permission is needed to use material that is not owned by the company.

To enhance copyright protection, businesses should register their original works with the national copyright office. Employing copyright notices and digital rights management tools can also provide additional protection for digital works (World Intellectual Property Organization, 2006).

What should you do if someone violates your copyright? It is the responsibility of the copyright owner to take action to address any infringement of their rights. The owner should first identify the violation and determine the appropriate steps to enforce their rights.

One option is for the copyright owner to send a letter to the alleged infringer, informing them of the potential conflict. It is recommended to seek legal assistance in crafting this letter. In certain countries, if copyright infringement occurs online, the owner can send a special cease and desist letter to the internet service provider (ISP) requesting the removal or blocking of the infringing content. Alternatively, the owner can directly notify the ISP, prompting them to inform their clients about the alleged infringement and work towards a resolution.

A copyright lawyer or law firm can offer guidance on available options and assist as needed. They can help determine if, when, how, and what legal actions may be appropriate against the infringers.

However, in certain situations, informing an infringer of a claim can allow them to hide or destroy evidence. If this is a possibility, the copyright owner who is aware of where the infringement is taking place may choose to seek court intervention without providing any prior notice to the infringer. They can then request an order that permits a surprise inspection of the infringer's premises and the seizure of pertinent evidence (World Intellectual Property Organization, 2006).

Some content or material is allowed to be used without permission. This is the case when the work is in the public domain. This is also the case when the use falls under the concepts of "fair use" or "fair dealing," or any other exception or limitation specifically outlined in the national copyright law. Fair use and fair dealing are principles followed in common law countries, such as the U.S., Canada, India, the U.K., and Australia. These principles acknowledge that certain uses of copyrighted works do not require the permission of the copyright holders, as the usage is minimal and won't unreasonably infringe on their exclusive rights. Individuals have more freedom to copy works for personal use under "fair use" compared to commercial use. Finally, content or material may be used without per-

mission if there is a part of the work that is protected by copyright law, such as facts or ideas that are used in one's own way without copying the author's expression, one is allowed to use them.



SUMMARY

The General Data Protection Regulation (GDPR) is an important regulation in the EU that protects the rights of individuals regarding their personal data. It consists of principles, individual rights, responsibilities, and enforcement. Personal data refers to any information related to an identified person, also known as the data subject. Data subjects have the right to own their data and request its deletion from accessible and third-party storage, under certain conditions. These conditions include when a data subject has given consent for data processing.

There are different legal grounds for processing personal data, such as having consent, being in a contractual relationship with the data subject, fulfilling a legal obligation, or protecting vital interests. GDPR also establishes a model with various roles and responsibilities for maintaining and processing personal data. This model includes the Information Commissioner's Office as the supervisory authority, as well as the data processor, data controller, data subject, third party, and connections to other countries. All these entities comply with the European Data Protection Board. Organizations and businesses need to have a comprehensive policy and well-informed staff to comply with copyright laws.