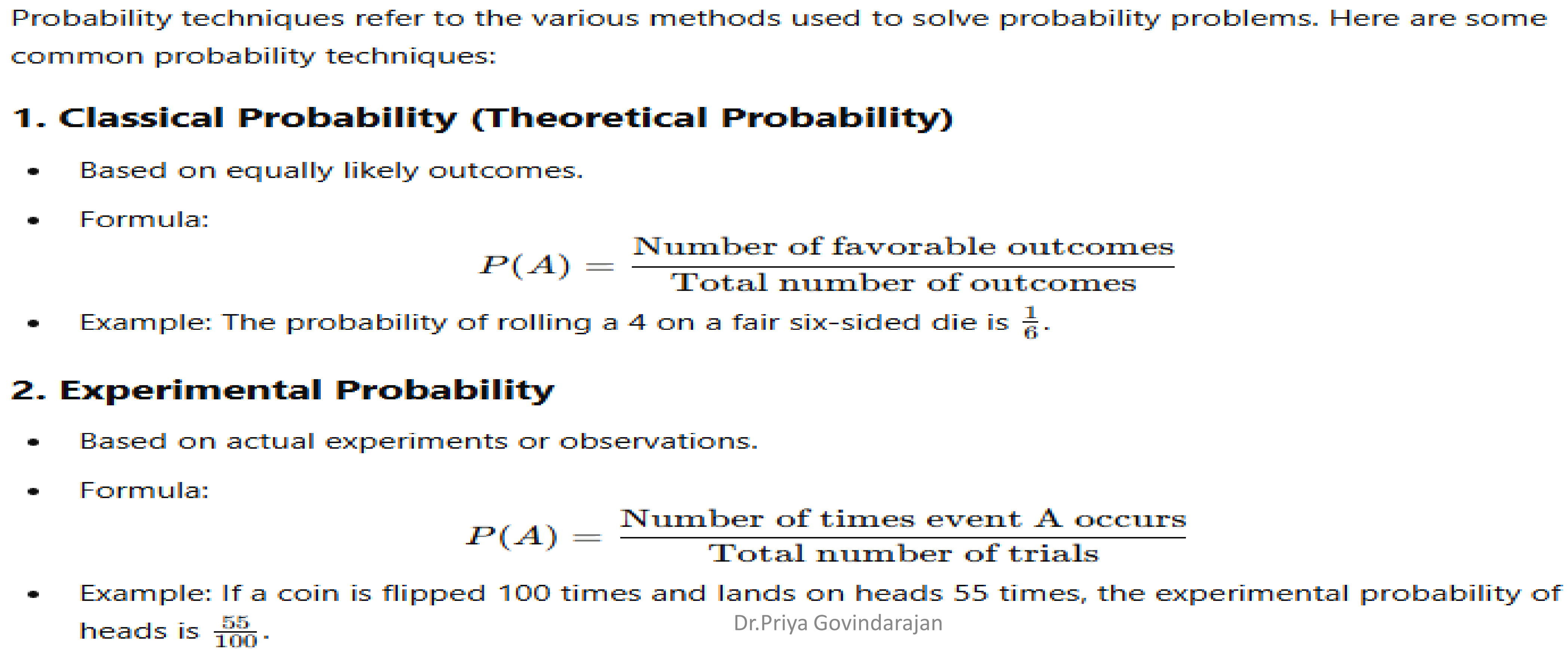
# Probability

Probability is the measure of how likely an event is to occur. It is expressed as a number between 0 and 1, where:

- **0** means the event will not happen.
- **1** means the event will definitely happen.
- A probability of **0.5** means the event has an equal chance of occurring or not.



Probability techniques refer to the various methods used to solve probability problems. Here are some common probability techniques:

### 1. Classical Probability (Theoretical Probability)

- Based on equally likely outcomes.
- Formula:

$$P(A) = \frac{\text{Number of favorable outcomes}}{\text{Total number of outcomes}}$$

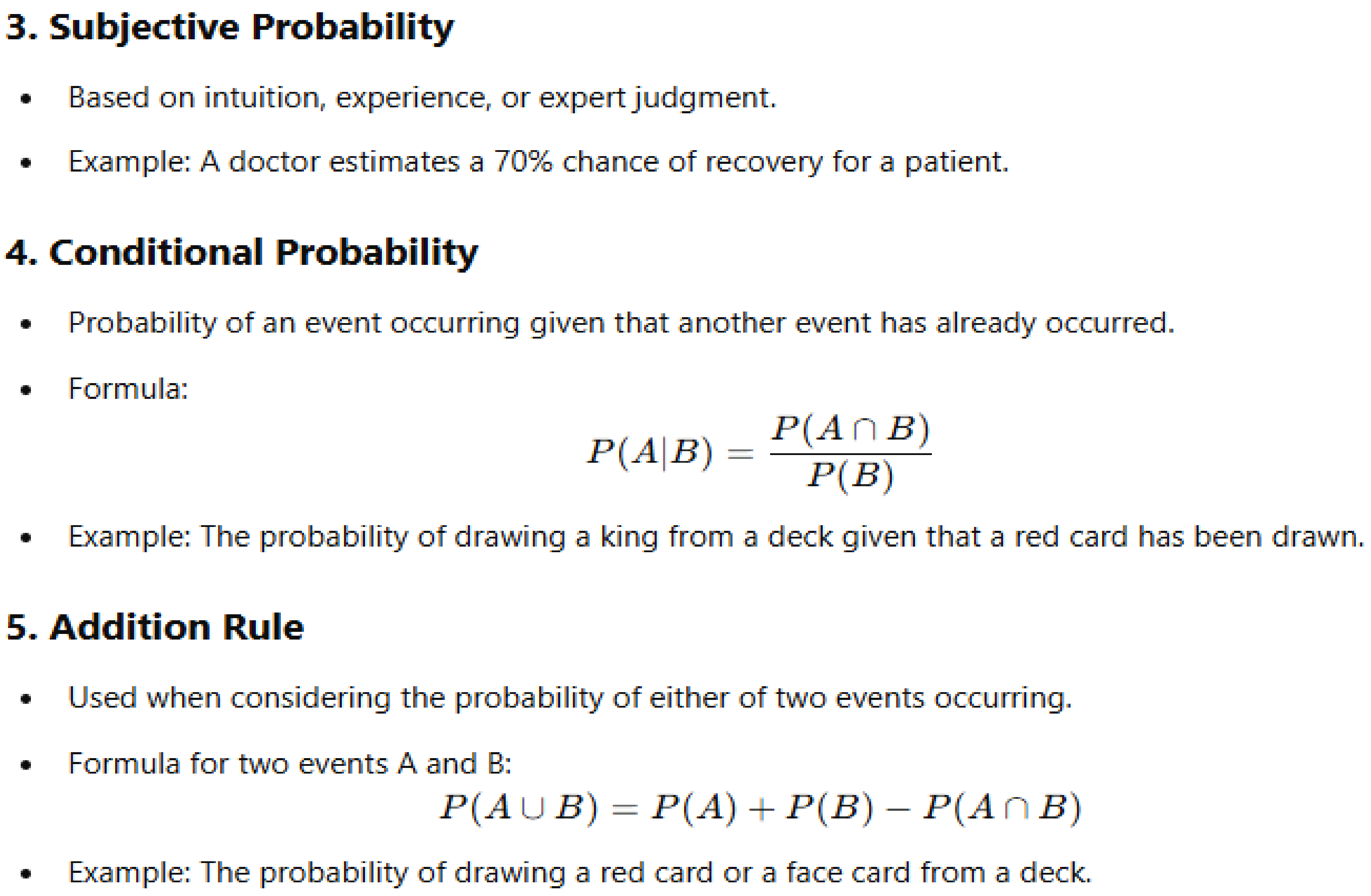
- Example: The probability of rolling a 4 on a fair six-sided die is $\frac{1}{6}$.

### 2. Experimental Probability

- Based on actual experiments or observations.
- Formula:

$$P(A) = \frac{\text{Number of times event A occurs}}{\text{Total number of trials}}$$

- Example: If a coin is flipped 100 times and lands on heads 55 times, the experimental probability of heads is $\frac{55}{100}$.



### 3. Subjective Probability

- Based on intuition, experience, or expert judgment.
- Example: A doctor estimates a 70% chance of recovery for a patient.

### 4. Conditional Probability

- Probability of an event occurring given that another event has already occurred.
- Formula:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

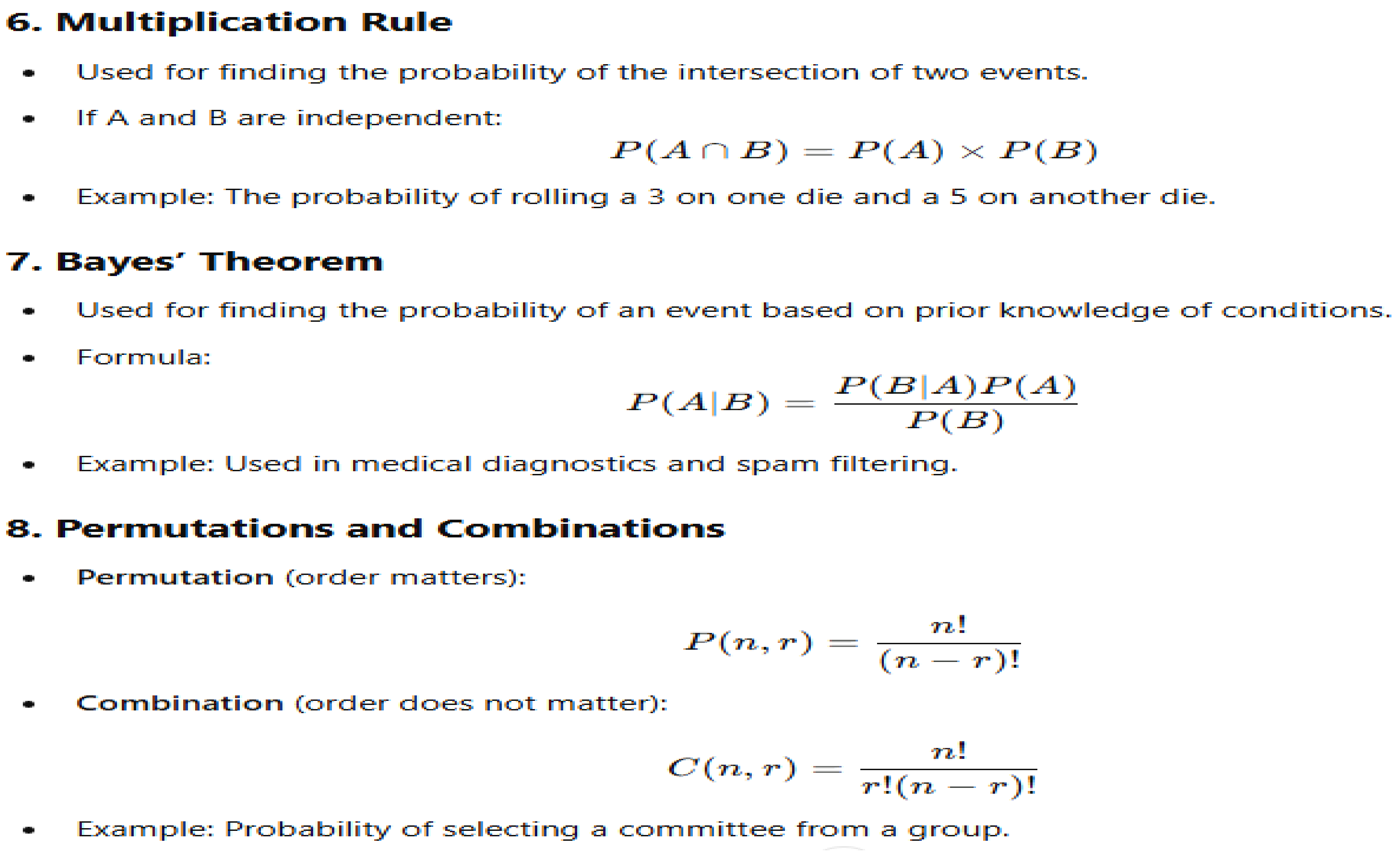
- Example: The probability of drawing a king from a deck given that a red card has been drawn.

### 5. Addition Rule

- Used when considering the probability of either of two events occurring.
- Formula for two events A and B:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

- Example: The probability of drawing a red card or a face card from a deck.



## 6. Multiplication Rule

- Used for finding the probability of the intersection of two events.
- If A and B are independent:

$$P(A \cap B) = P(A) \times P(B)$$

- Example: The probability of rolling a 3 on one die and a 5 on another die.

## 7. Bayes' Theorem

- Used for finding the probability of an event based on prior knowledge of conditions.
- Formula:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- Example: Used in medical diagnostics and spam filtering.

## 8. Permutations and Combinations

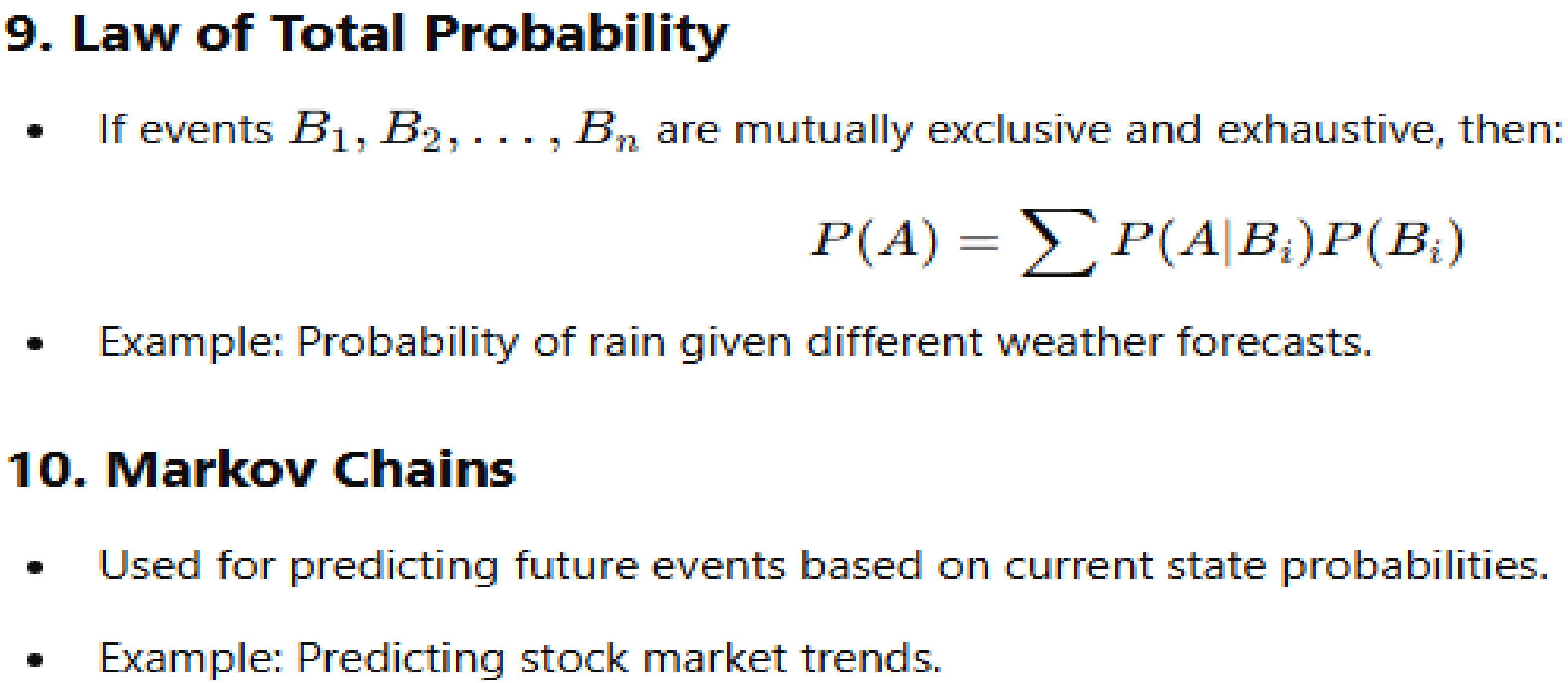
- **Permutation** (order matters):

$$P(n,r) = \frac{n!}{(n-r)!}$$

- **Combination** (order does not matter):

$$C(n,r) = \frac{n!}{r!(n-r)!}$$

- Example: Probability of selecting a committee from a group.



## 9. Law of Total Probability

- If events $B_1, B_2, \ldots, B_n$ are mutually exclusive and exhaustive, then:

$$P(A) = \sum P(A|B_i)P(B_i)$$

- Example: Probability of rain given different weather forecasts.

## 10. Markov Chains

- Used for predicting future events based on current state probabilities.
- Example: Predicting stock market trends.

# Bayesian Theory

- Bayes' theorem is also known as Bayes' rule, Bayes' law, or Bayesian reasoning, **which determines the probability of an event with uncertain knowledge.**

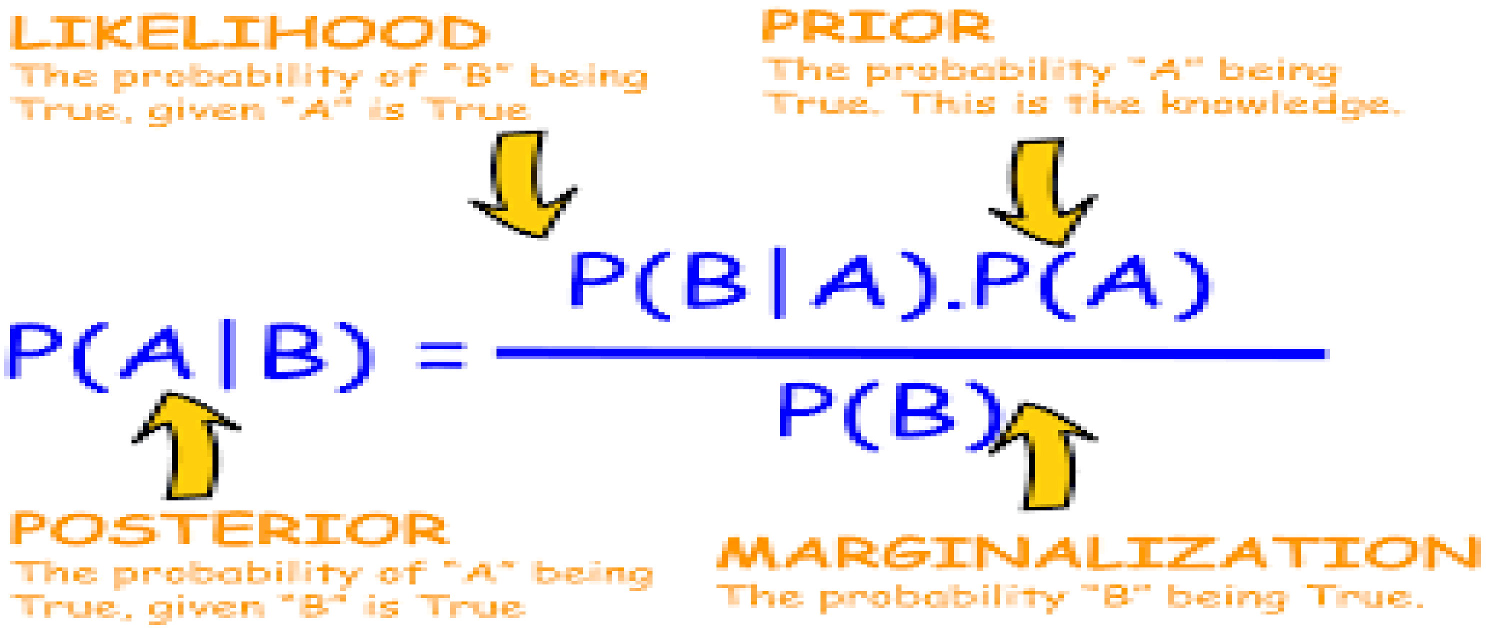
- In probability theory, it relates the conditional probability and marginal probabilities of two random events.

- Bayes' theorem was **named after the British mathematician Thomas Bayes**. The **Bayesian inference is an application of Bayes' theorem, which is fundamental to Bayesian statistics.**

- **It is a way to calculate the value of P(B|A) with the knowledge of P(A|B).**

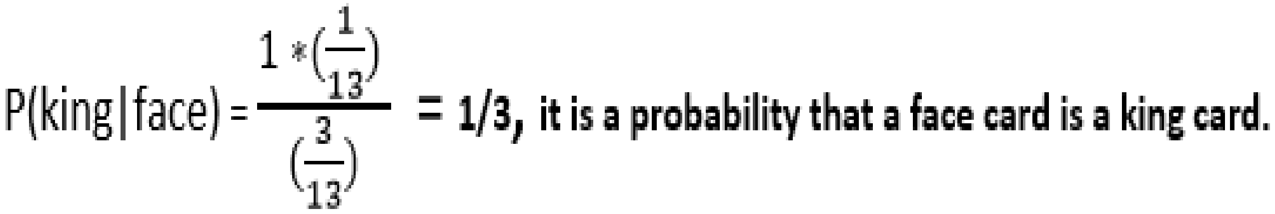
- Bayes' theorem **allows updating the probability - prediction of an event by observing new information** of the real world.

# **Bayes' rule** or **Bayes' theorem**.


LIKELIHOOD
The probability of "B" being True, given "A" is True
PRIOR
The probability "A" being True. This is the knowledge.
$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$
POSTERIOR
The probability of "A" being True, given "B" is True
MARGINALIZATION
The probability "B" being True.

Question: From a standard deck of playing cards, a single card is drawn. The probability that the card is king is 4/52, then calculate posterior probability P(King|Face), which means the drawn face card is a king card.

- P(king): probability that the card is King= 4/52= 1/13
- P(face): probability that a card is a face card= 12/52 = 3/13
- P(Face|King): probability of face card when we assume it is a king = 1
- Implementation:



$$P(king|face) = \frac{1 * (\frac{1}{13})}{(\frac{3}{13})}$$ **= 1/3, it is a probability that a face card is a king card.**

**Applications of Bayes' theorem:**

•It is used to **calculate the next step of the robot when the already executed step is given**.

•Bayes' theorem is helpful in **weather forecasting.**

## Naïve Bayes Classifier Algorithm

- Naïve Bayes algorithm is a **supervised learning algorithm**, which is based on **Bayes theorem** and used for solving classification problems.
- It is mainly **used in *text classification* that includes a high-dimensional training dataset.**
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the **fast machine learning models that can make quick predictions.**
- **It is a probabilistic classifier, which means it predicts on the basis of the probability of an object**.
- Some popular examples of Naïve Bayes Algorithm are **spam filtration, Sentimental analysis, and classifying articles**.

**The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:**

- **Naïve**: **It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features.** Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- **Bayes**: It is called Bayes **because it depends on the principle of Bayes' Theorem**.

**Advantages of Naïve Bayes Classifier:**

- Naïve Bayes is one of the **fast and easy** ML algorithms to predict a class of datasets.
- It can be used **for Binary as well as Multi-class Classifications**.
- It performs well in Multi-class predictions as compared to the other Algorithms.
- It is the most popular choice for **text classification problems**.

**Disadvantages of Naïve Bayes Classifier:**

- Naive Bayes assumes that **all features are independent or unrelated, so it cannot learn the relationship between features.**

**Applications of Naïve Bayes Classifier:**

- It is used for **Credit Scoring**.
- It is used in **medical data classification**.
- It can be used in **real-time predictions** because Naïve Bayes Classifier is an eager learner.
- It is used in Text classification such as **Spam filtering** and **Sentiment analysis**.

# Example 1: Using Naïve Bayes Classifier, find a fruit – which is Yellow, Sweet, and Long.

| Fruit | Yellow | Non-yellow | Sweet | Non-Sweet | Long | Not long | Total |
|---|---|---|---|---|---|---|---|
| Orange | 350 | 300 | 450 | 200 | 0 | 650 | 650 |
| Banana | 400 | 0 | 300 | 100 | 350 | 50 | 400 |
| Others | 50 | 100 | 100 | 50 | 50 | 100 | 150 |
| Total | 800 | 400 | 850 | 350 | 400 | 800 | 1200 |

1. P(Yellow|Orange) = P(Orange|Yellow).P(Yellow) / P(Orange) = (350|800).(800|1200) / (650|1200)

2. P(Sweet|Orange) =

3. P(Long|Orange) =

P(Fruit|Orange) = 0.5X0.69X0 = 0

1. P(Yellow|Banana) =

2. P(Sweet|Banana) =

3. P(Long|Banana) =

P(Fruit|Banana) =  ? = 0.64

1. P(Yellow|Others) =

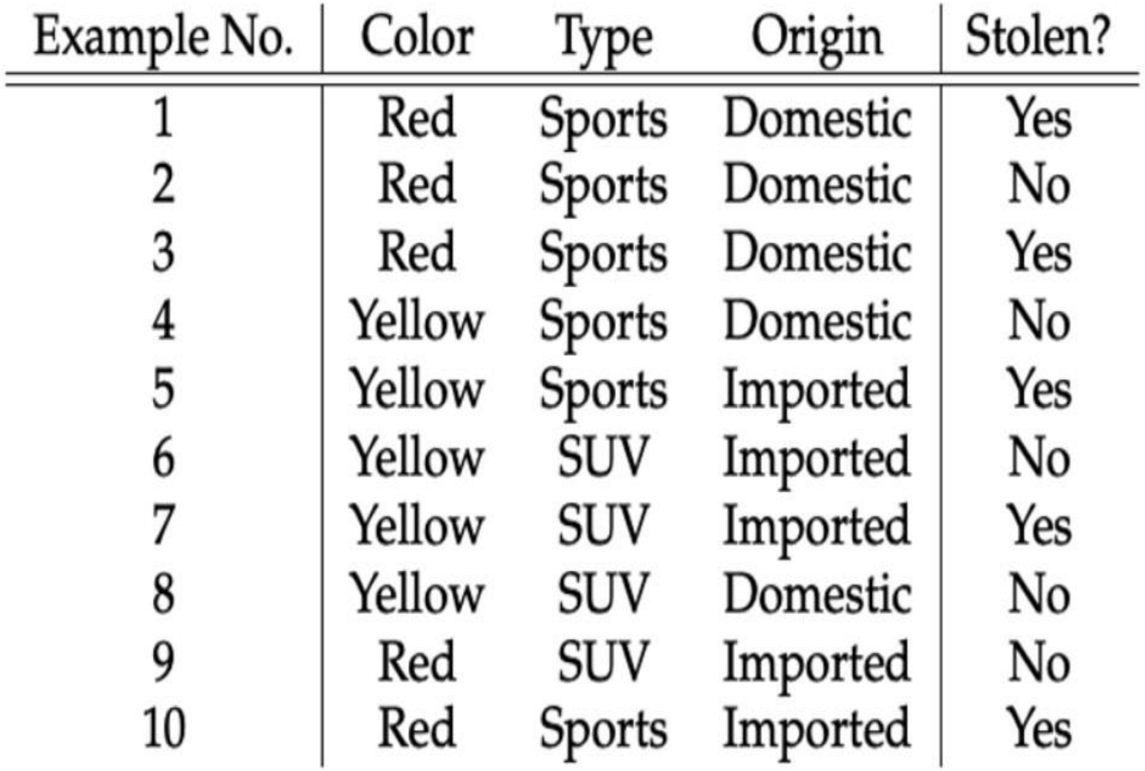
2. P(Sweet|Others) =

3. P(Long|Others) =

P(Fruit|Others) = ? = 0.072

**Find the highest probability, among the fetched probabilities – to find the fruit.**

## Example 2: For the given dataset, apply the Naive-Bayes algorithm and predict the outcome for Car = {Red, Domestic, SUV}



| Example No. | Color | Type | Origin | Stolen? |
|---|---|---|---|---|
| 1 | Red | Sports | Domestic | Yes |
| 2 | Red | Sports | Domestic | No |
| 3 | Red | Sports | Domestic | Yes |
| 4 | Yellow | Sports | Domestic | No |
| 5 | Yellow | Sports | Imported | Yes |
| 6 | Yellow | SUV | Imported | No |
| 7 | Yellow | SUV | Imported | Yes |
| 8 | Yellow | SUV | Domestic | No |
| 9 | Red | SUV | Imported | No |
| 10 | Red | Sports | Imported | Yes |

1. P(Red|Yes) = P(Yes|Red). P(Red) / P(Yes) = (3/5).(5/10) / (5/10) = 3/5

2. P(Domestic|Yes) = ?

3. P(SUV|Yes) = ?

P(X|Yes) = (3/5)(2/5)(1/5) = 0.048

1. P(Red|No) = ?

2. P(Domestic|No) = ?

3. P(SUV|No) = ?

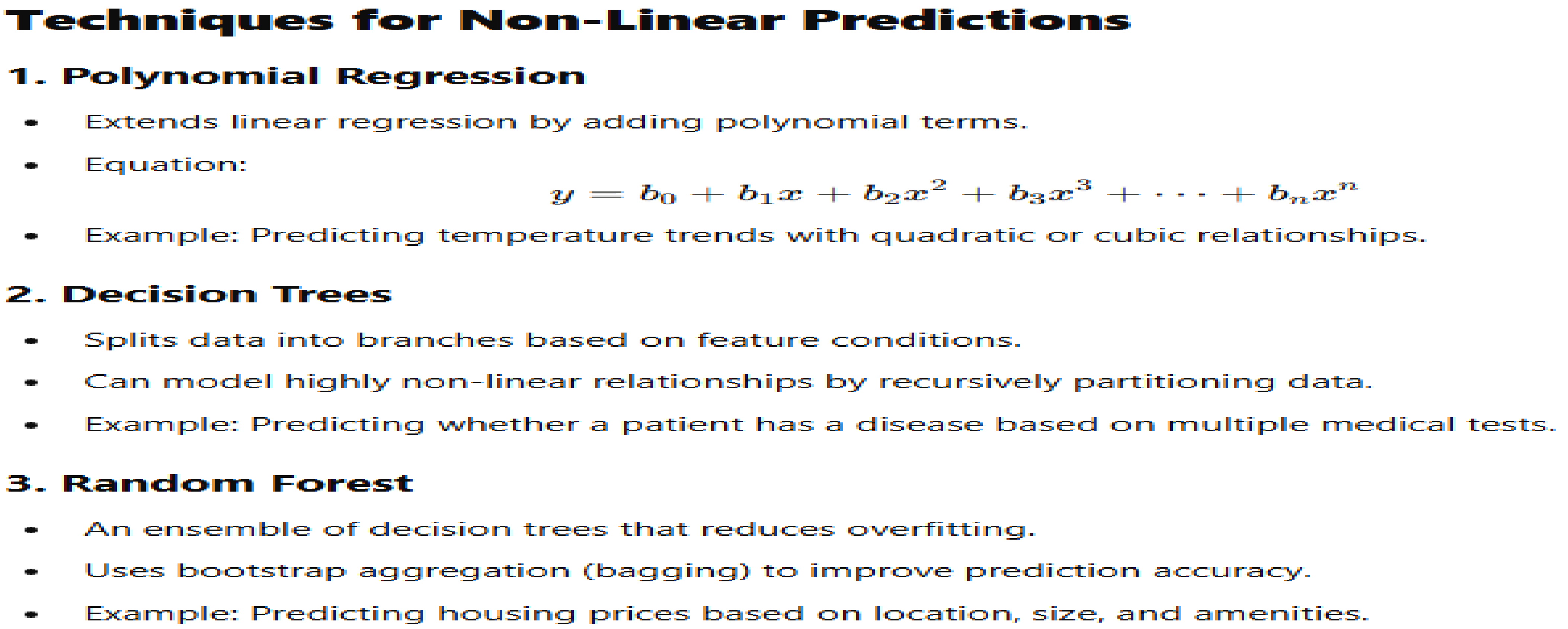
P(X|No) = (2/5)(3/5)(2/5) = 0.096

The biggest probabilities, among the fetched ones – is the outcome.

# Non-linear predictions

**Non-Linear Prediction**

- Non-linear prediction refers to forecasting or estimating future outcomes where the relationship between input and output variables does not follow a straight line. Unlike linear models, non-linear models capture complex dependencies, interactions, and patterns in data.



## Techniques for Non-Linear Predictions

### 1. Polynomial Regression

- Extends linear regression by adding polynomial terms.
- Equation:

$$y = b_0 + b_1x + b_2x^2 + b_3x^3 + \cdots + b_nx^n$$

- Example: Predicting temperature trends with quadratic or cubic relationships.

### 2. Decision Trees

- Splits data into branches based on feature conditions.
- Can model highly non-linear relationships by recursively partitioning data.
- Example: Predicting whether a patient has a disease based on multiple medical tests.

### 3. Random Forest

- An ensemble of decision trees that reduces overfitting.
- Uses bootstrap aggregation (bagging) to improve prediction accuracy.
- Example: Predicting housing prices based on location, size, and amenities.

## 4. Support Vector Machines (SVM) with Non-Linear Kernels

- Uses kernel functions to transform data into a higher-dimensional space where it becomes linearly separable.
- Common Kernels:
  - **Polynomial Kernel:** Captures polynomial relationships.
  - **Radial Basis Function (RBF) Kernel:** Captures complex patterns.
- Example: Classifying images of handwritten digits.

A kernel function is a mathematical function that compares data points in a higher-dimensional space. It's used in machine learning to detect non-linear relationships between data.

It calculates the inner product between pairs of data points in the higher-dimensional space.
It uses the inner product to determine how similar the data points are.

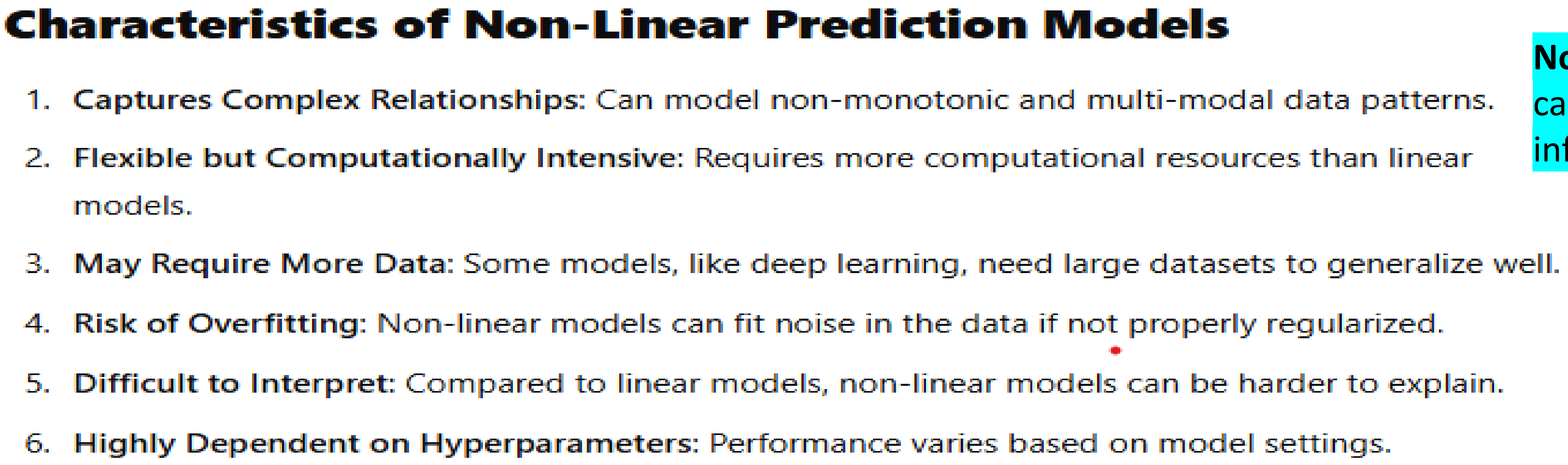
## 5. Neural Networks (Deep Learning)

- Uses multiple layers of neurons with non-linear activation functions (e.g., ReLU, Sigmoid, Tanh).
- Can model highly complex relationships between inputs and outputs.
- Example: Speech recognition and natural language processing.

The Rectified Linear Unit (ReLU) is an activation function used in neural networks to introduce non-linearity.
Hyperbolic tangent - It helps neural networks make complex decisions by determining which neurons to activate.

## 6. k-Nearest Neighbors (k-NN)

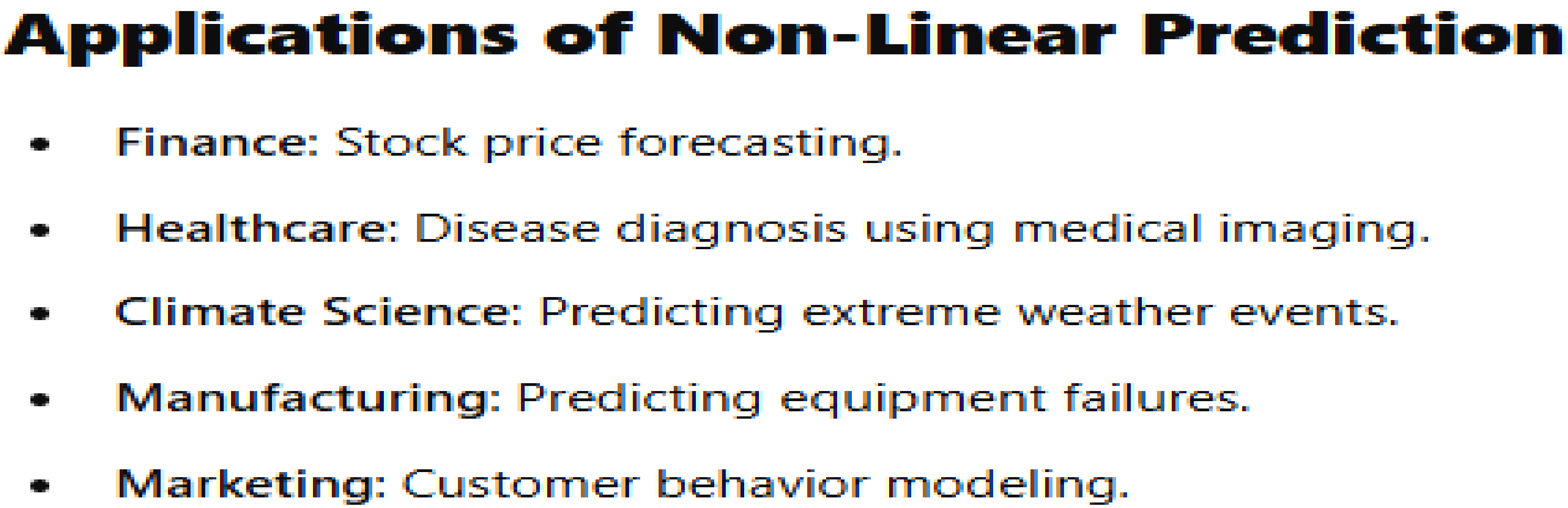
- A non-parametric method that predicts based on the closest k neighbors in the training set.
- Example: Recommender systems (e.g., suggesting movies based on user preferences).



## Characteristics of Non-Linear Prediction Models

1. **Captures Complex Relationships:** Can model non-monotonic and multi-modal data patterns.
2. **Flexible but Computationally Intensive:** Requires more computational resources than linear models.
3. **May Require More Data:** Some models, like deep learning, need large datasets to generalize well.
4. **Risk of Overfitting:** Non-linear models can fit noise in the data if not properly regularized.
5. **Difficult to Interpret:** Compared to linear models, non-linear models can be harder to explain.
6. **Highly Dependent on Hyperparameters:** Performance varies based on model settings.

**Non-monotonic** - Inferences can change based on new information.



# Applications of Non-Linear Prediction

- Finance: Stock price forecasting.
- Healthcare: Disease diagnosis using medical imaging.
- Climate Science: Predicting extreme weather events.
- Manufacturing: Predicting equipment failures.
- Marketing: Customer behavior modeling.

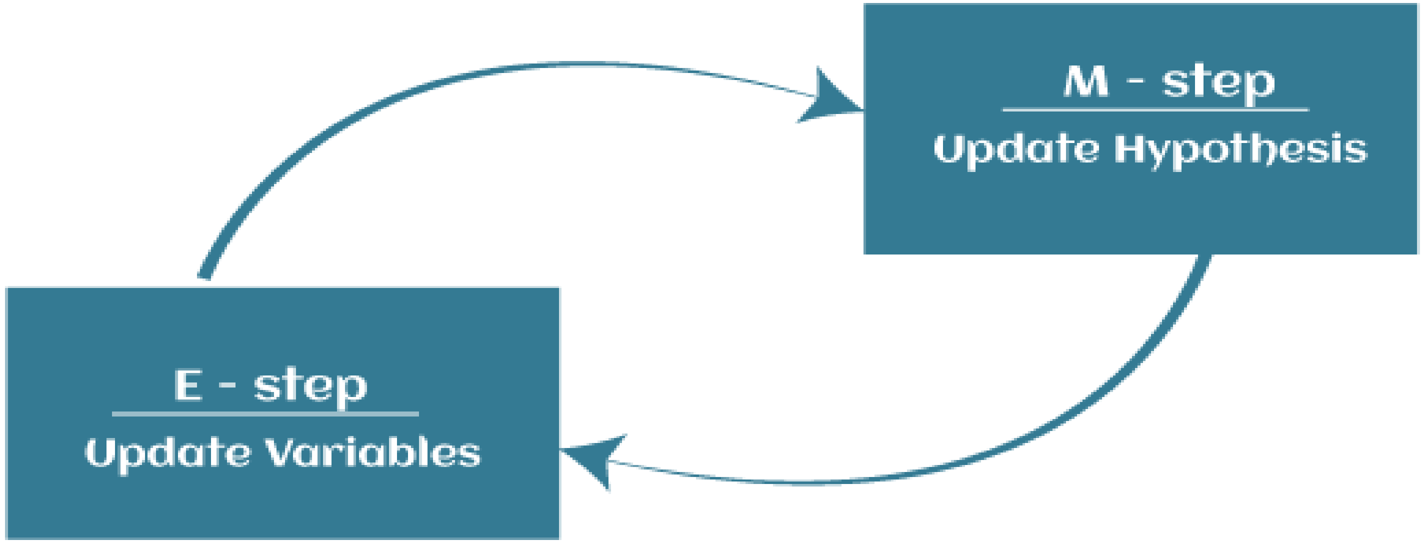
# Expectation-Maximization Algorithm

***The EM algorithm is considered to find the local maximum likelihood parameters of a statistical model, proposed by Arthur Dempster, Nan Laird, and Donald Rubin in 1977***. The EM (Expectation-Maximization) algorithm is one of the most commonly **used terms in machine learning to obtain maximum likelihood estimates of variables that are sometimes observable and sometimes not.**

If the variables are observable, then it can predict the value using instances. **On the other hand, the variables which are latent or directly not observable, for such variables Expectation-Maximization (EM) algorithm plays a vital role to predict the value with the condition.**

## EM Algorithm

The EM algorithm is the **combination of various unsupervised ML algorithms**, such as the **k-means clustering algorithm**. Being an iterative approach, **it consists of two modes**. In the first mode, we **estimate the missing or latent variables**. Hence it is referred to as the **Expectation/estimation step (E-step**). Further, the other mode is used **to optimize the parameters of the models so that it can explain the data more clearly**. The second mode is known as the **maximization-step or M-step.**



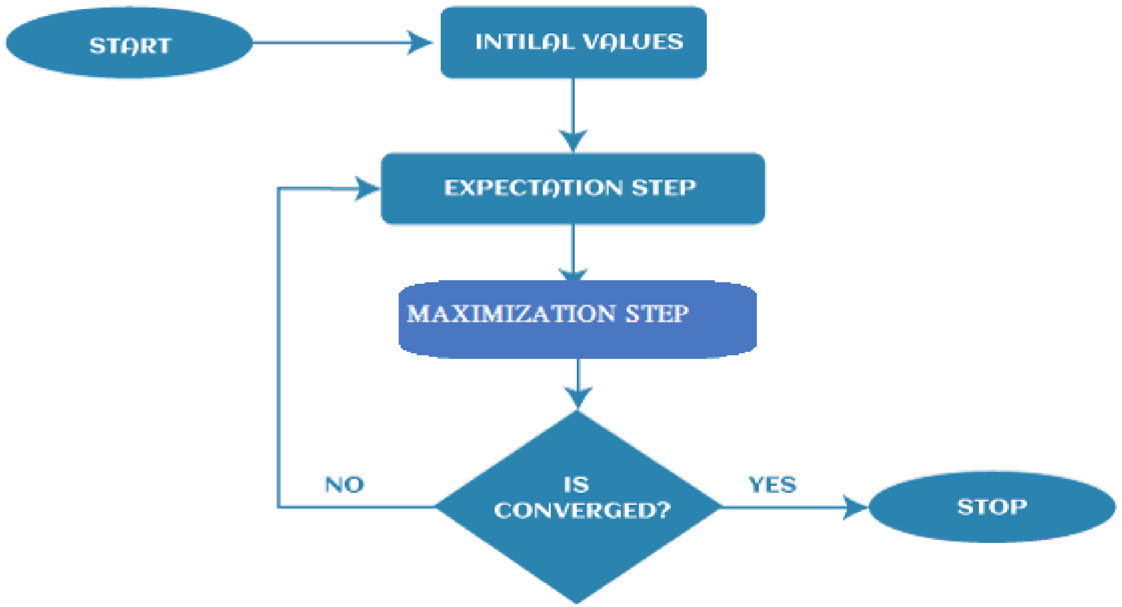
- **Expectation step (E - step):** It involves the **estimation (guess) of all missing values in the dataset** so that after completing this step, there should **not be any missing values**.
- **Maximization step (M - step):** This step involves the use of **estimated data in the E-step and updating the parameters.**
- **Repeat E-step and M-step until the convergence of the values occurs**.
- The primary goal of the EM algorithm is to use the available observed data of the dataset to estimate the missing data of the latent variables and then use that data to update the values of the parameters in the M-step.

## Convergence in the EM algorithm

- ***Convergence is defined as the specific situation in probability based on intuition***, e.g., **if there are two random variables that have very less difference in their probability, then they are known as converged. In other words, whenever the values of given variables are matched with each other, it is called convergence.**

## Steps in EM Algorithm

- The EM algorithm is completed mainly in 4 steps, which include I***nitialization Step, Expectation Step, Maximization Step, and convergence Step***. These steps are explained as follows:



- **1st Step:** The very first step is to initialize the parameter values. Further, the system is provided with incomplete observed data with the assumption that data is obtained from a specific model.
- **2nd Step:** This step is known as Expectation or E-Step, which is used to estimate or guess the values of the missing or incomplete data using the observed data. Further, E-step primarily updates the variables.
- **3rd Step:** This step is known as Maximization or M-step, where we use complete data obtained from the 2nd step to update the parameter values. Further, M-step primarily updates the hypothesis.
- **4th step:** The last step is to check if the values of latent variables are converging or not. If it gets "yes", then stop the process; else, repeat the process from step 2 until the convergence occurs.

**Usage of EM algorithm –**

- It can be used to **fill the missing data** in a sample.
- It can be used as **the basis of unsupervised learning of clusters.**
- It can be used for the purpose of **estimating the parameters of Hidden Markov Model (HMM) -** Describes the **probabilistic relationship between a sequence of observations and a sequence of hidden states**.
- It can be used for **discovering the values of latent variables**.

**Advantages of EM algorithm –**

- It is always **guaranteed that likelihood will increase with each iteration**.
- The **E-step and M-step** are often pretty **easy for many problems in terms of implementation**.
- Solutions to the M-steps often exist in the closed form.

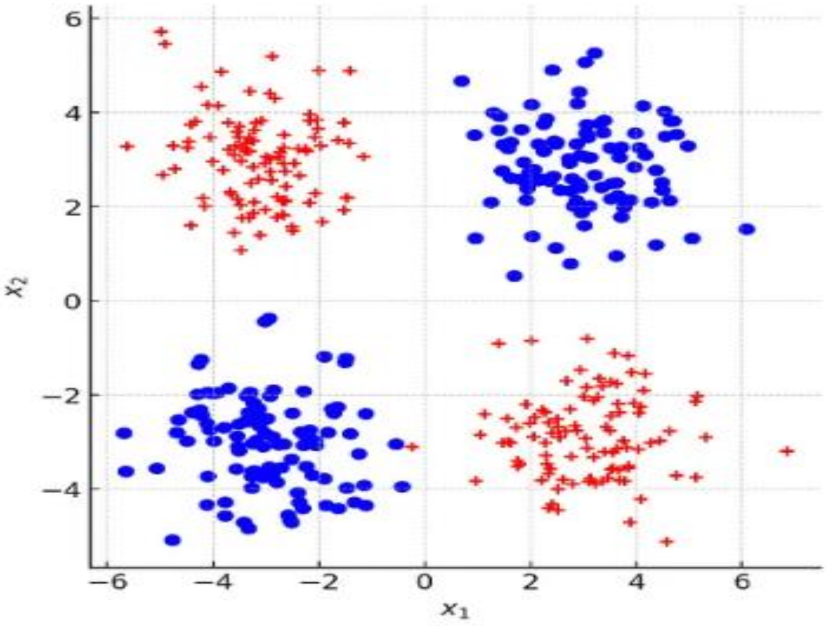
**Disadvantages of EM algorithm –**

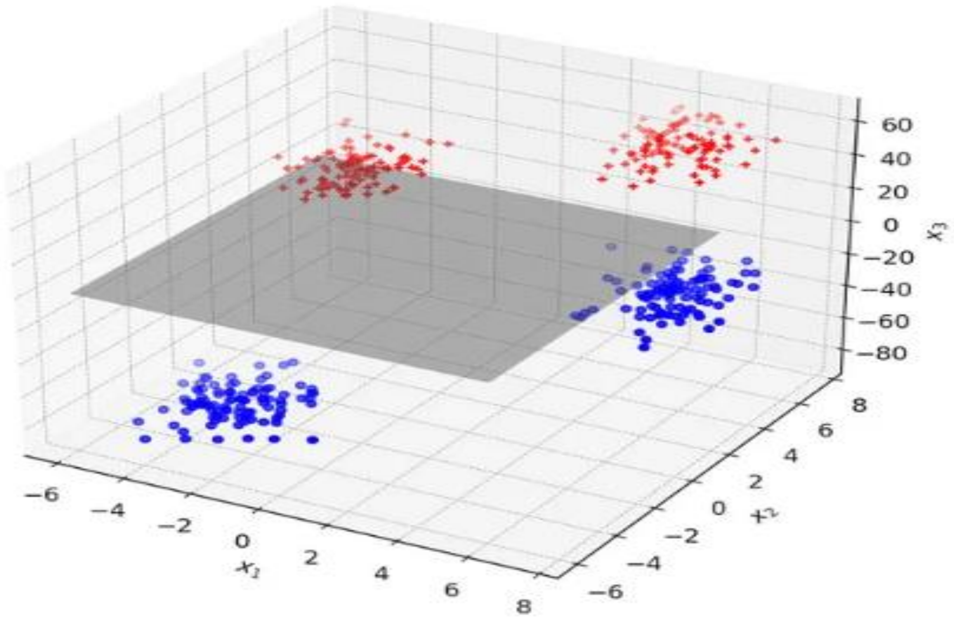
- It **has slow convergence**.
- It requires both the **probabilities, forward and backward** (numerical optimization requires only forward probability).

# Kernels

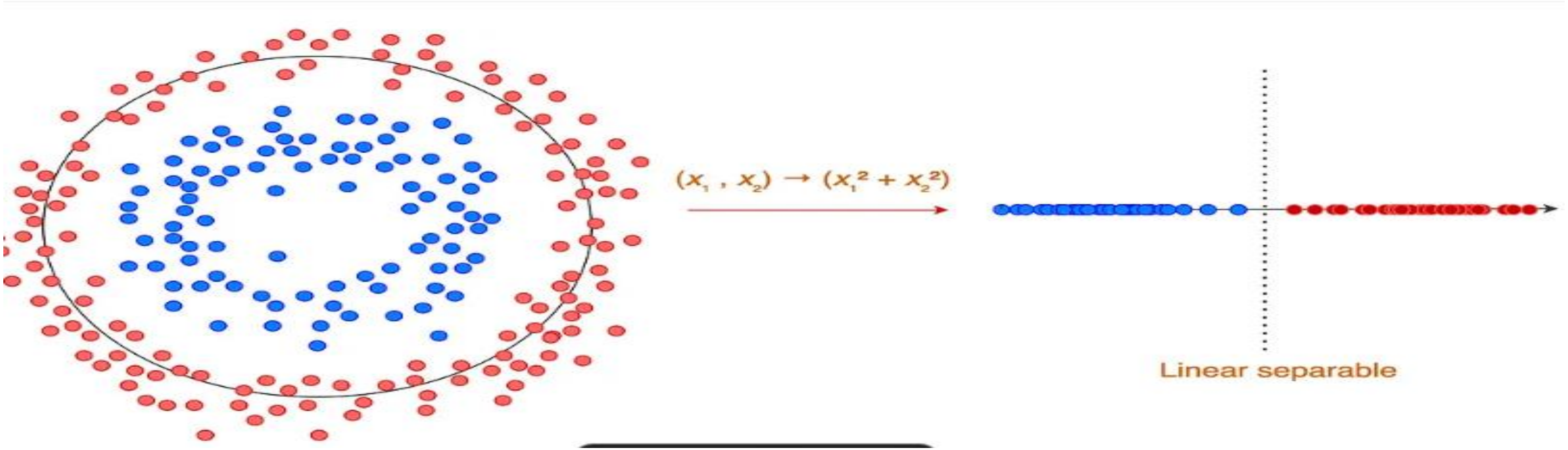
**In machine learning, a kernel is a function that transforms data into a higher-dimensional space. This allows linear methods to be applied to non-linear problems. Kernel methods are used to solve complex decision-making problems.**

In many machine learning problems, input data is transformed into a higher-dimensional feature space using a non-linear mapping to make it easier to find patterns or separate classes. Subsequently, **linear models can project this feature space into a one-dimensional space for classification. For example, in the left diagram below, the red and blue dots cannot be separated by a straight line in their original two-dimensional space. However, when transformed into a three-dimensional space using a non-linear function, a plane can successfully separate the two classes.**





- **Kernel functions offer a principled approach to detecting nonlinear relationships using linear algorithms. This is achieved by implicitly mapping the original input data into a feature space where the data might become linearly separable.** For example, with the mapping Fn. $\phi(x) = x_1^2 + x_2^2$, the red and blue dots in the diagram above are now linearly separable in this newly transformed feature space, as opposed to their original input space.



- However, **in real-world problems, the feature space is often high-dimensional to untangle complex data, and the mapping $\phi$ is usually unknown**. However, **one can use machine learning or deep learning techniques to model $\phi$. Once modeled, a linear algorithm such as Support Vector Machines (SVM) can be applied to the feature space.** In the example, a 2D input can be transformed into a pre-assumed 9D feature space using a third-degree polynomial. Then, inferences are made with the linear model

## Types of kernels:

**Linear kernel**: Used for linear problems

**Polynomial kernel**: Used for **non-linear** problems

**Gaussian kernel**: Used for **non-linear** classification problems

**Sigmoid kernel**: Used for **binary classification** problems

**Laplacian kernel**: Used for **non-linear** classification problems

**Radial basis function (RBF) kernel**: Used to **model complex decision boundaries**

## How to choose a kernel

The **best kernel depends on the problem, data type, and available computational resources**.

One can **experiment with different kernels to find the best one for a given problem.**

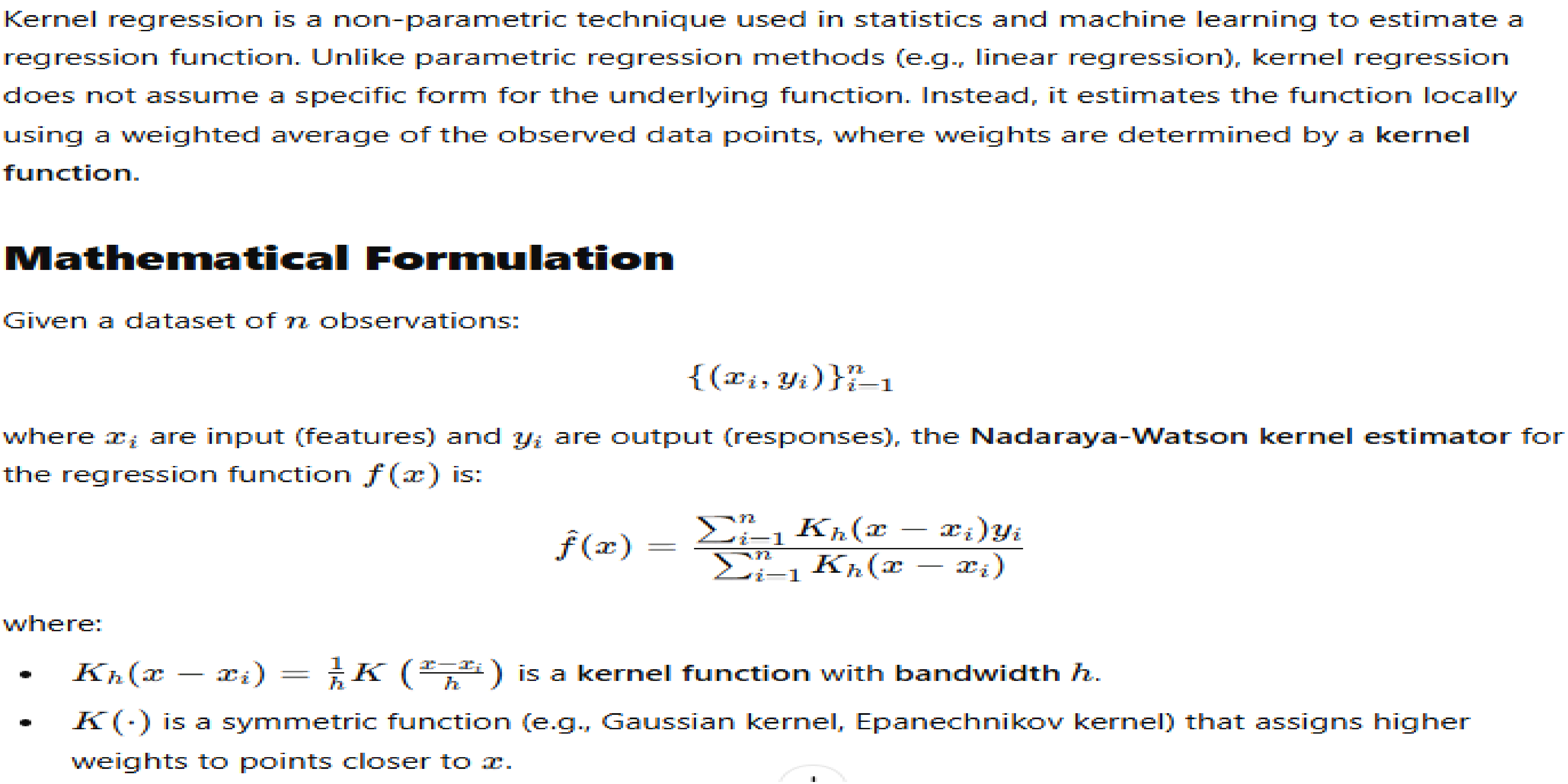
**Custom kernels can be designed based on domain knowledge or problem-specific requirements.**

## Kernel methods in machine learning

Kernel methods are particularly **useful when dealing with high-dimensional data.**

Kernel methods are a **staple machine learning approach in Natural Language Processing (NLP).**

# Kernel regression



Kernel regression is a non-parametric technique used in statistics and machine learning to estimate a regression function. Unlike parametric regression methods (e.g., linear regression), kernel regression does not assume a specific form for the underlying function. Instead, it estimates the function locally using a weighted average of the observed data points, where weights are determined by a **kernel function.**

## Mathematical Formulation

Given a dataset of $n$ observations:

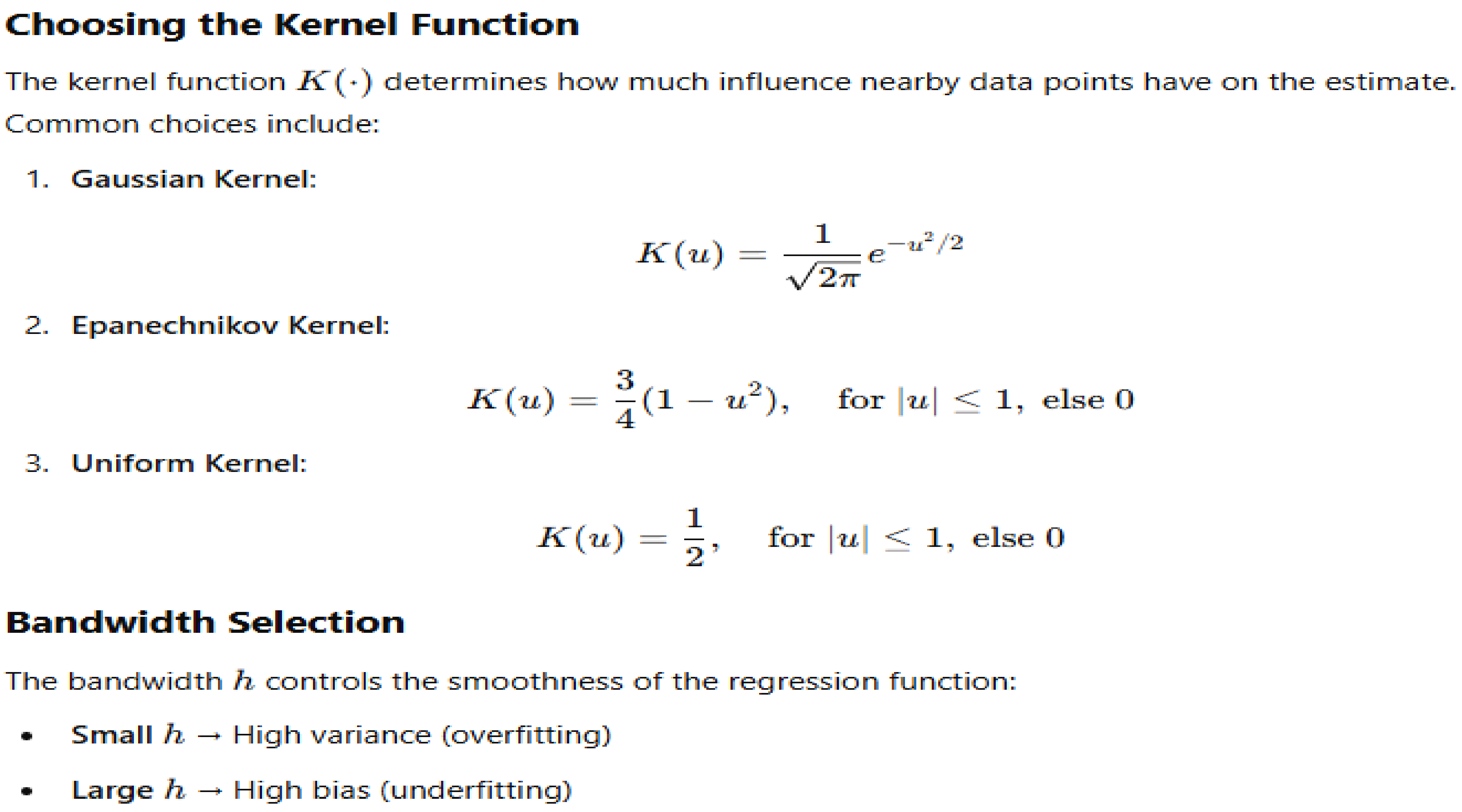
$$\{(x_i, y_i)\}_{i=1}^{n}$$

where $x_i$ are input (features) and $y_i$ are output (responses), the **Nadaraya-Watson kernel estimator** for the regression function $f(x)$ is:

$$\hat{f}(x) = \frac{\sum_{i=1}^{n} K_h(x - x_i) y_i}{\sum_{i=1}^{n} K_h(x - x_i)}$$

where:

- $K_h(x - x_i) = \frac{1}{h} K\left(\frac{x - x_i}{h}\right)$ is a **kernel function with bandwidth** $h$.
- $K(\cdot)$ is a symmetric function (e.g., Gaussian kernel, Epanechnikov kernel) that assigns higher weights to points closer to $x$.



## Choosing the Kernel Function

The kernel function $K(\cdot)$ determines how much influence nearby data points have on the estimate. Common choices include:

1. **Gaussian Kernel:**

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-u^2/2}$$

2. **Epanechnikov Kernel:**

$$K(u) = \frac{3}{4}(1 - u^2), \quad \text{for } |u| \leq 1, \text{ else } 0$$

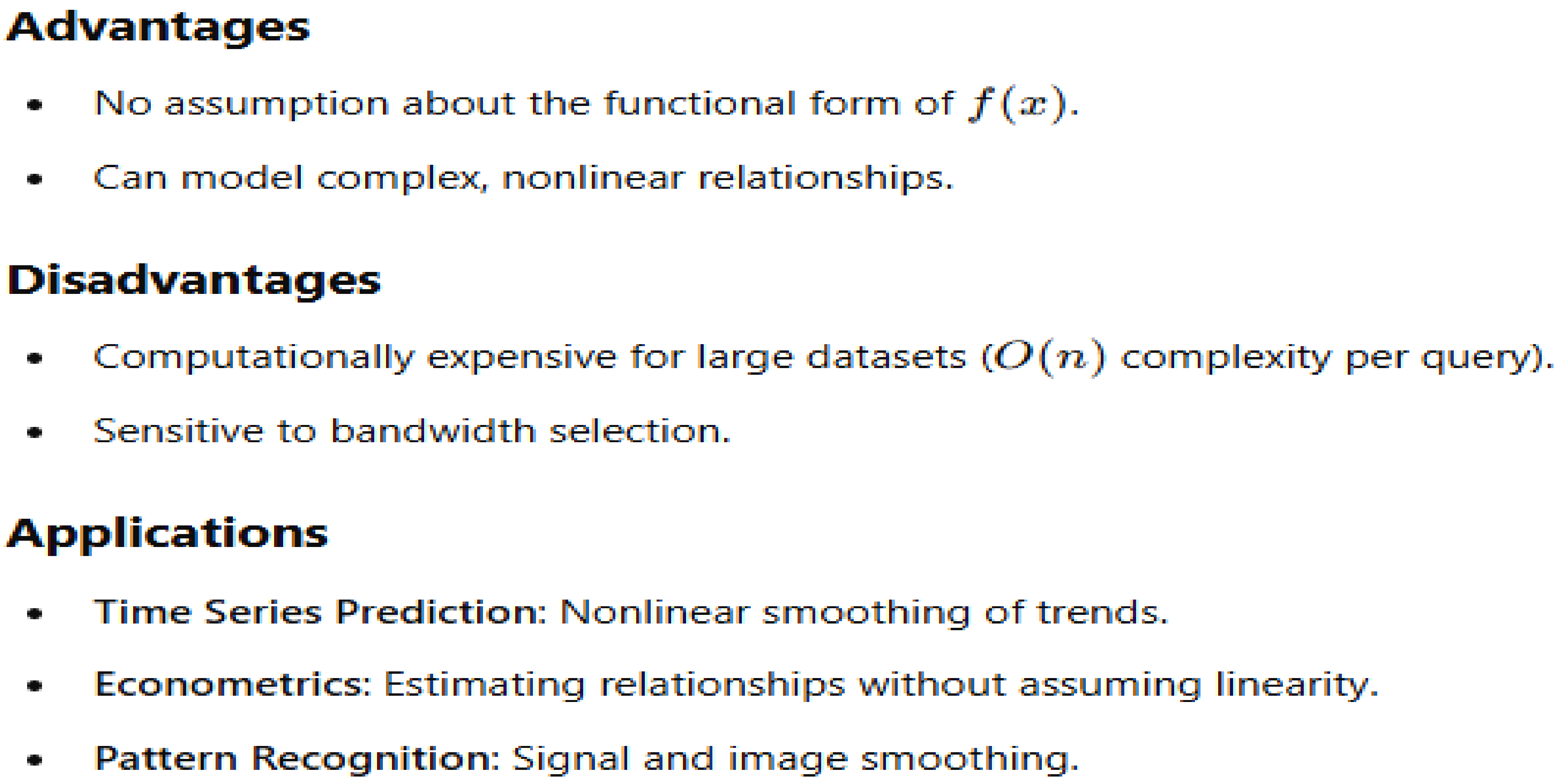
3. **Uniform Kernel:**

$$K(u) = \frac{1}{2}, \quad \text{for } |u| \leq 1, \text{ else } 0$$

## Bandwidth Selection

The bandwidth $h$ controls the smoothness of the regression function:

- **Small** $h$ → High variance (overfitting)
- **Large** $h$ → High bias (underfitting)



## Advantages

- No assumption about the functional form of $f(x)$.
- Can model complex, nonlinear relationships.

## Disadvantages

- Computationally expensive for large datasets ($O(n)$ complexity per query).
- Sensitive to bandwidth selection.

## Applications

- **Time Series Prediction:** Nonlinear smoothing of trends.
- **Econometrics:** Estimating relationships without assuming linearity.
- **Pattern Recognition:** Signal and image smoothing.