Math’s and statistics

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

Probability is the backbone of self-driving cars, enabling them to make decisions in unpredictable environments.

Types of probability:

1.Classical (Theoretical) Probability :

The car detects a blurry red sign ahead and needs to know: Is this a **STOP** sign?

**How it works:**  
Based on known shapes/colors:



* Red octagon: 90% chance it's a STOP sign



* Red triangle: 10% chance it's a YIELD sign



2. **Empirical (Experimental) Probability**

**Empirical Probability** is the **probability based on observed data or past experiments**, not theory

**Scenario 2: Pedestrian Behavior**



Your car observes 500 pedestrians near crosswalks:

* 420 cross when signal is green
* 80 cross when signal is red

3) Subjective Probability:

**Subjective probability** is the **personal judgment or belief** about how likely something is to happen — even **without hard data**.

It’s often used when:

* No historical data is available
* The event is uncertain, complex, or one-time
* We rely on **expert experience, intuition, or confidence levels**

**🔸 Scenario 1: Rare Situations (Edge Cases)**

Suppose the car has **never seen a child suddenly run into the road** in that area.

But the system (or human engineers) might **assign a subjective probability**:

“There’s a 10% chance a child might enter the street here due to a nearby playground.”

Even without hard data, this belief affects how cautiously the car behaves:

* Slower approach
* Wider detection zone
* More prediction uncertainty

3) Bayesian Probability:

**Bayesian probability** is a way of **updating our beliefs** when **new evidence** is observed.

It's built on **Bayes' Theorem**, which combines:

* What we **believed before** (prior knowledge)
* What we **just observed** (new data)
* To get what we **should believe now** (posterior probability)

**Example 2: Pedestrian Intent Prediction**

Is the pedestrian going to cross?

* Prior: P(cross)=0.4P(\text{cross}) = 0.4P(cross)=0.4 (from general statistics)
* Evidence: Looking toward the road and stepping forward
* Likelihood: P(evidence∣cross)=0.9P(\text{evidence} \mid \text{cross}) = 0.9P(evidence∣cross)=0.9

Apply Bayes’ theorem → update P(cross evidence)≈0.85P(\text{cross} \mid \text{evidence}) \approx 0.85P(cross evidence)≈0.85

✅ Car plans to **brake** or **adjust speed**.

# Module 2 :- Random Variables and Probability Distributions

Random Variables:

A **random variable** is a **numerical value** that represents the outcome of a **random experiment**.

It is called “random” because its value is **not known until the experiment happens** — but we know the **possible values** it can take and their **probabilities**.

Types of random variable: Discrete, Continuous

Discrete random variable:

* A **random variable** is a function that assigns a numerical value to each outcome in a sample space.
* A **discrete** random variable takes **countable** values, usually integers or categories.
* Examples: Number of pedestrians detected, number of traffic lights turned red, lane change count.

So Random variable it helps 1)Count the number of **pedestrians, vehicles, traffic signs, obstacles**.

2) Used as input to models that decide **when to brake, slow down, or change lanes**.

Continuous random variable :

A **continuous random variable** can take **any real number value** within a range

Majorly it deals with Distance, speed, time , angle, acceleration

It uses LiDAR, Radar, and Cameras measure distances continuously.

Probability distribution:

**All the possible values a variable can take**

**The probability of each value or range happening**

**We have like two types of distribution**

**1)discrete distribution**

**2) continuous distribution**

**Discrete distribution:**

**It deals with countable values like how many pedestrians x={0,1,2,3}are there by that it will calculate whether it should stop or slow or to go fast**

**Continuous probability distribution:**

**Describes probabilities for variables that can take any value within a range (e.g., speed, distance).**

**Example:**

**If a travels at speed of 55 mph and it will calculate a standard deviation of 5 mph so it help to car to to accelerate or stop or to change the lanes by using the various sensors**

**Descriptive Statistics:**

**1. Descriptive Statistics**

* **Definition: Summarizes and describes raw data to understand its main features.**
* **Use: Analyzes sensor data like average speed or distance to obstacles.**
* **Example: Calculating the average speed of nearby vehicles to understand traffic flow.**

**2. Inferential Statistics**

* **Definition: Makes predictions or generalizations about a larger population from sample data.**
* **Use: Estimates the likelihood of events based on partial observations.**
* **Example: Determining the probability that a detected object is a pedestrian, based on sensor samples.**

**3. Predictive Statistics**

* **Definition: Uses historical data to forecast future outcomes or behaviors.**
* **Use: Anticipates movements or actions of other vehicles or pedestrians.**
* **Example: Predicting the future trajectory of a car to avoid collisions.**

**4. Prescriptive Statistics**

* **Definition: Suggests optimal decisions or actions based on data analysis.**
* **Use: Recommends safe maneuvers and adjustments for driving.**
* **Example: Advising the car to slow down or change lanes based on predicted traffic.**

# Module 4: System of Linear Equations

**A system of linear equations is a set of two or more linear equations involving the same variables.**

**Our goal is to all the equation at the same time.**

**Purpose of linear equations:**

**In real-world scenarios (like self-driving cars), multiple sensors give partial views (each modeled as a linear equation).  
By solving them together as a system, the car can make a more accurate decision — like:**

* **Detect object position**
* **Align with lane center**
* **Estimate motion direction**

**Because in self driving cars we have different types of sensor to get information of different object it helps us get the actual position of the object**

# Module 5: Matrices

**Matrix : A matrix is a rectangular arrangement of numbers in rows and columns.**

**It’s a compact way to represent data, equations, or transformations.**

**Why is Matrix Important in Self-Driving Cars?**

**Sensor Fusion ------** **Combine multiple inputs**

**Object Detection ---------- Represent image pixels as matrices**

**Path Planning -------------- Solve systems of equations using matrices**

**Camera Calibration ---------- Use transformation matrices**

| **Matrix Type** | **Math Example** | **Self-Driving Use Case** |
| --- | --- | --- |
| **Square** | **[2 4;3 1][2\ 4; 3\ 1][2 4;3 1]** | **Sensor fusion** |
| **Zero** | **[0 0;0 0][0\ 0; 0\ 0][0 0;0 0]** | **No input (sensor failure)** |
| **Identity** | **[1 0;0 1][1\ 0; 0\ 1][1 0;0 1]** | **Keep frame unchanged** |
| **Diagonal** | **[0.8 0;0 0.3][0.8\ 0; 0\ 0.3][0.8 0;0 0.3]** | **Confidence weighting** |
| **Symmetric** | **[0 5;5 0][0\ 5; 5\ 0][0 5;5 0]** | **Distance between objects** |
| **Row** | **[30 31 33][30\ 31\ 33][30 31 33]** | **Speed over time** |
| **Column** | **[60;62;65][60; 62; 65][60;62;65]** | **Speed from 3 sensors** |
| **Sparse** | **[0 0 1;0 0 0;2 0 0] [0\ 0\ 1; 0\ 0\ 0; 2\ 0\ 0] [0 0 1;0 0 0;2 0 0]** | **LiDAR data** |
| **Transformation** | **[0 −1;1 0] [0\ -1; 1\ 0] [0 −1;1 0]** | **Rotate camera image** |

# Module 6: Vector Spaces

A **vector** is a way to represent any quantity that has both **magnitude** and **direction** related to the car or its environment.

It helps the car understand **where it is**, **where things are**, and **how things move**.

The car’s GPS position is a vector [x,y,z] that tells where it is on the map.

Other moving objects’ velocities (like pedestrians or other cars) are vectors too.

Vectors help track where objects are and where they’re moving.

* 1. **Distance of a Vector in Self-Driving Cars**
* a **vector** might represent a set of sensor readings or the position of an object relative to the car.
* The **distance of the vector** can represent **how far away an object is** from your car.
* For example, if a sensor gives the position of an obstacle as:
* v⃗=[x,y]

where:

x = distance to the right or left (lateral) y = distance forward or backward (longitudinal)

**2. Angle of a Vector in Self-Driving Cars**

The **angle of the vector** shows **where the obstacle or object is located relative to the car’s forward direction**.

* If your car is pointing forward along the yyy-axis, the angle tells you if the object is to the left, right, or straight ahead.

Graph :  A **graph** is a way to represent **objects and the connections between them**.

 It consists of:

* **Nodes (or vertices):** The objects (like people, places, sensors, or cars).
* **Edges:** The connections or relationships between nodes (like roads, communication links, or sensor detections).
* What is **Centrality**?
* **Centrality** measures **how important a node is** in the graph.

### Common types of centrality:

* **Degree Centrality:** Number of edges connected to a node (how many direct connections it has).
* **Betweenness Centrality:** How often a node appears on shortest paths between other nodes (how important it is as a bridge).
* **Closeness Centrality:** How close a node is to all other nodes (measures efficiency in spreading information).
* **Eigenvector Centrality:** How important a node is based on the importance of its neighbors.

Imagine you have a **graph representing the environment around the car**:

* **Nodes:** Objects detected by sensors (other cars, pedestrians, traffic lights).
* **Edges:** Relationships or interactions (distance, line of sight, or communication links).
* **Degree Centrality:** A pedestrian with many cars close by might be more “important” to watch carefully.
* **Betweenness Centrality:** A car at a busy intersection might be critical for traffic flow.
* **Closeness Centrality:** A sensor node that can quickly communicate with many other sensors is important for fast data sharing.
* **Eigenvector Centrality:** An obstacle that is connected to many important objects (like other cars) may affect the car’s decisions more.