**Week 2**

●Bayesian Thinking:

○ Probability

○ Conditional Probability

○ Programming Probability in Python

○ Bayes’ Rule

○ Programming Probability Distributions

**Topic: Bayesian Thinking**

**Lesson 3: Probability**

**3.1 Uncertainty in Driving**

Due to noise, it is not always easy to figure out the exact output when giving instructions to cars. Keeping in mind that the instruments used in self driving cars are prone to error, it is therefore not possible to be 100% sure of what is happening or what is going to happen. This however does not warrant discarding the concept of SDCs because humans too are prone to errors (possibly more than machines) and yet we are still able to ‘safely’ drive around regardless s our errors.

**3.2 Uncertainty in Robotics**

Using the word **certain** the way we'll use it in this Nanodegree, *nothing* in the previous question is ever certain. Let me explain.

1. **What other traffic will do**: People are impossible to predict with certainty!
2. **Where you are**: It may seem like you know where you are when you drive, but you don't. At least not with *complete* certainty. You may know where you are with *sufficient* certainty, but if I asked you how many millimeters away from the center lane you were, you wouldn't know.
3. **How fast you're going**: The same reasoning applies to knowing your speed. You can get a good idea of how fast you're going by looking at your speedometer (which measures your speed), but these measurements are never perfect.
4. **What will happen when you turn the wheel**: A car is an imperfect mechanical system. If you turned the steering wheel by the same amount 100 times, the car would turn a slightly different amount every time.

As humans we solve these problems in a variety of ways. Number **2** and **3** we solve by saying "ehh, I actually don't need to know exactly where I am or how fast I'm going, I just need to know those quantities with a high degree of certainty". Number **4** we solve by using our brain as a high performance adaptive controller. And number **1**... who knows how we handle number 1...? (just kidding, you'll learn more about this in the machine learning course at the end of this curriculum).

**3.3. Learning objectives explained**

**3.4. Learning objectives: Probability**

The lesson checks background knowledge in probability

**3.5. Learning objectives: Probability**

1,1-1,2-1,3-1,4-1,5-1,6 | 2,1-2,2-2,3-2,4-2,5-2,6 | 3,1-3,2-3,3-3,4-3,5-3,6

4,1-4,2-4,3-4,4-4,5-4,6 | 5,1-5,2-5,3-5,4-5,5,5,6 | 6,1-6,2-6,3-6,4-6,5-6,6

Total = 36

7ups = 6

**3.6. Probability**

In statistics, we are given data and try to infer the possible causes that relate to the data while in probability, we are given causes and try to infer the possible data.

Statistics is the opposite of probability

We study probability because it enables us to establish the relationship between data and underlying causes.

**3.7 Flipping coins quiz**

Probability is the study of anticipated outcomes

… skipped the rest of probability from 3.7 to 3.25

**Lesson 4: Conditional Probability**

**4.1 Conditional Probability**

Conditional Probability is the mechanism that a self driving car uses to take sensor data and infer something else like I am driving at speed X and will arrive at position Y in time Z.

**4.2 Intro to Conditional Probability**

Predict if this time next year the weather will be cloudy

**4.3 Estimating based on conditions**

Another exercise to predict if the weather will be cloudy in the next one minute

Self driving cars take new **sensor measurements** as often as possible to ensure the data they use to make probabilistic predictions is as "fresh" (and therefore as useful) as possible.

**4.4 Dependent events and conditional probability**

Conditional probability involves taking data about what we know and using this data to determine what we don’t know.

**4.5 and 4.6 Recap of previous lessons**

**4.7 Learning objectives – Conditional Probability**

We use the notation where

* P(A)P(A)P(A) means "the probability of A"
* P(¬A)P(\neg A)P(¬A) means "the probability of NOT A"
* P(A,B)P(A,B)P(A,B) means "the probability of A **and** B" and
* P(A∣B)P(A|B)P(A∣B) means "the probability of A **given** B.

The lesson checks if I understand the above notation and their usage in practice

**4.8 Dependent things**

The outcome of the first scenario affects the outcome of the second and subsequent scenarios

**4.9 Notation note**

This lesson will use some notation that you may not be familiar with. The table below explains some of that notation.

| **Symbol** | **Usage** | **Interpretation** |
| --- | --- | --- |
| PPP | P(A)P(A)P(A) | "The **probability** of event AAA occurring" |
| ¬\lnot¬ | P(¬A)P(\lnot A)P(¬A) | "The probability of event AAA **not** occurring" |
| ∣\mid∣ | P(A∣B)P(A \mid B)P(A∣B) | "The probability of event AAA occurring **given that** event BBB occurs" |

**4.10 --- 4.17 Quizes**

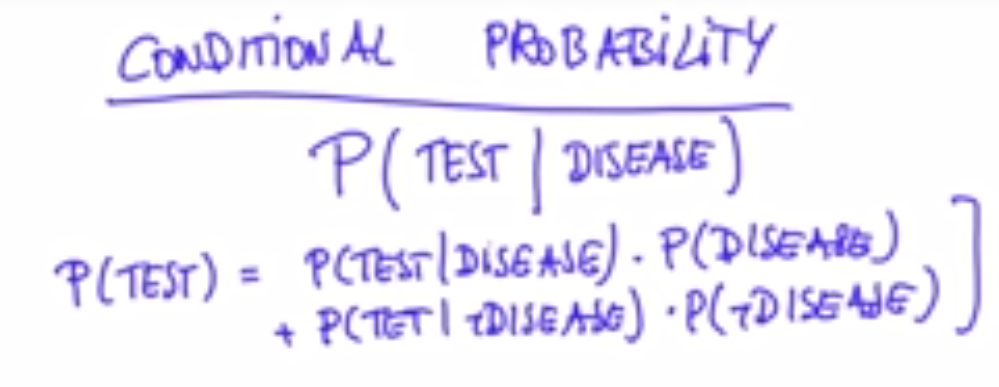
**4.18 Total Probability**

The total probability is always 1. We can therefore use what we know and subtract from one to get what we don’t know.

The ability to use Total Probability in finding out what we don’t know is a key element in Conditional probability

**4.19 --- 4.22 Quizes**

**4.23 Summary**



**Lesson 5: Programming Probability in Python**

**5.1 Learn by Doing**

‘You don’t learn to exercise by watching other people exercise’

**5.2 Your first programming practice**

There will be demonstrations, playgrounds and exercises. Demonstrations are meant to give a glimpse of a concept using code. The playground is an opportunity to test out the concept and the exercises to test out the understanding.

When you get to a demonstration you should:

1. read through the code and comments.
2. come up with a hypothesis for what will happen when you press "Test Run"
3. press Test Run and observe the result. Compare what you see to what you expected. If there is a difference, it means there's an opportunity to learn something! take time to reflect on what was different and why.

**5.3 Python Variable**