Data Science from scratch

Contents

[I. Fundamentals 2](#_Toc81476268)

[1. Matrices & Linear Algebra Fundamentals 2](#_Toc81476269)

[2. Hash Functions, Binary Trees 2](#_Toc81476270)

[3. Relational Algebra and DB basics 2](#_Toc81476271)

[4. Inner, Outer, Cross and Theta Join. 2](#_Toc81476272)

[5. CAP Theorem 2](#_Toc81476273)

[6. Sharding 3](#_Toc81476274)

[7. OLAP 3](#_Toc81476275)

[8. Multidimensional Data Model 4](#_Toc81476276)

[9. ETL 5](#_Toc81476277)

[10. JSON & XML 5](#_Toc81476278)

[11. NoSQL 6](#_Toc81476279)

[12. Regex (Regular Expressions) 7](#_Toc81476280)

[13. Virtual Environments Setup (ISN’T COMPLETED) 7](#_Toc81476281)

[II. Programming (Python and R) 8](#_Toc81476282)

[1. Install Packages 8](#_Toc81476283)

[2. Factor Analysis 8](#_Toc81476284)

[3. Functions 9](#_Toc81476285)

[4. Read Data. 10](#_Toc81476286)

[5. Manipulate Data Frames 11](#_Toc81476287)

[6. Subsetting Data Frames 12](#_Toc81476288)

[III. Statistics (and how to calculate them in Python) 12](#_Toc81476289)

[1. Descriptive Statistics 12](#_Toc81476290)

[2. Histograms 14](#_Toc81476291)

[3. Percentiles and Outliers 15](#_Toc81476292)

[4. Probability Theory 15](#_Toc81476293)

[5. Bayes Theorem 17](#_Toc81476294)

# Fundamentals

## Matrices & Linear Algebra Fundamentals

Linear algebra topics:

* Vectors
* Matrices
* Transpose of a matrix
* Inverse of a matrix
* Determinant of a matrix
* Trace of a matrix
* Covariance Matrix
* Dot product
* Eigenvalues
* Eigenvectors

## Hash Functions, Binary Trees

Hashing is a technique that is used to uniquely identify a specific object from a group of similar objects. Some examples of how hashing is used in our lives include:

* In universities, each student is assigned a unique roll number that can be used to retrieve information about them.
* Diagram

  Description automatically generatedIn libraries, each book is assigned a unique number that can be used to determine information about the book, such as its exact position in the library or the users it has been issued to etc.

In both these examples the students and books were hashed to a unique number.

With binary trees you can do a lot of operations like transversals, summations, construction, conversions, etc.



## Relational Algebra and DB basics

 Relational algebra query operations are performed recursively on a relation. The output of these operations is a new relation, which might be formed from one or more input relations.

The relational databases are tabular data that are interrelated.

## What is cross join SQL and When to use cross join SQL?Inner, Outer, Cross and Theta Join.



Table

Description automatically generated

## CAP Theorem

* Consistency
* Availability
* Partition Tolerance

There’s a trade off between consistency and availability when the system has partitions. Partitioning is when the system can’t communicate each part with each other. If the system never has partitions, you can make the system consistent and available.

In real world you have degrees of consistency and degrees of availability and make trade offs between those two.

## Sharding

Sharding is when you have an enormous amount of data and you want to access them in the fastest way. This process is by fragmentating the data into small pieces and storing them in different servers to speed up the performance.

One way to access to every piece is by hierarchical sharding. Which means that you cut your piece into another small pieces and so on.

NoSQL uses sharding internally.

## OLAP

Diagram, engineering drawing

Description automatically generatedOLAP enables fast, flexible multidimensional data analysis for business intelligence (BI) and decision support applications.

OLAP (for online analytical processing) is software for performing multidimensional analysis at high speeds on large volumes of data from a data warehouse, data mart, or some other unified, centralized data store.

In theory, a cube can contain an infinite number of layers. (An OLAP cube representing more than three dimensions is sometimes called a hypercube.) And smaller cubes can exist within layers—for example, each store layer could contain cubes arranging sales by salesperson and product. In practice, data analysts will create OLAP cubes containing just the layers they need, for optimal analysis and performance.

**OLAP cubes enable four basic types of multidimensional data analysis:**

**Drill-down**

The drill-down operation converts less-detailed data into more-detailed data through one of two methods—moving down in the concept hierarchy or adding a new dimension to the cube. For example, if you view sales data for an organization’s calendar or fiscal quarter, you can drill-down to see sales for each month, **moving down in the concept hierarchy of the “time” dimension.**

**Roll up**

**Roll up is the opposite of the drill-down function**—it aggregates data on an OLAP cube by moving up in the concept hierarchy or by reducing the number of dimensions. For example, you could move up in the concept hierarchy of the “location” dimension by viewing each country's data, rather than each city.

**Slice and dice**

The slice operation **creates a sub-cube by selecting a single dimension from the main OLAP cube**. For example, you can perform a slice by highlighting all data for the organization's first fiscal or calendar quarter (time dimension).

The dice operation isolates a **sub-cube by selecting several dimensions within the main OLAP cube**. For example, you could perform a dice operation by highlighting all data by an organization’s calendar or fiscal quarters (time dimension) and within the U.S. and Canada (location dimension).

**Pivot**

**The pivot function rotates the current cube view to display a new representation of the data—enabling dynamic multidimensional views of data.** The OLAP pivot function is comparable to the pivot table feature in spreadsheet software, such as Microsoft Excel, but while pivot tables in Excel can be challenging, OLAP pivots are relatively easier to use (less expertise is required) and have a faster response time and query performance.

## Multidimensional Data Model

The multi-Dimensional Data Model is a method which is used for ordering data in the database along with good arrangement and assembling of the contents in the database.

A screenshot of a computer

Description automatically generated with medium confidenceThe Multi-Dimensional Data Model allows customers to interrogate analytical questions associated with market or business trends, unlike relational databases which allow customers to access data in the form of queries. They allow users to rapidly receive answers to the requests which they made by creating and examining the data comparatively fast.

**OLAP (online analytical processing) and data warehousing uses multi-dimensional databases.** It is used to show multiple dimensions of the data to users.

It represents data in the form of data cubes. Data cubes allow to model and view the data from many dimensions and perspectives. It is defined by dimensions and facts and is represented by a fact table. Facts are numerical measures and fact tables contain measures of the related dimensional tables or names of the facts.

The following stages should be followed by every project for building a Multi-Dimensional Data Model:

**Stage 1: Assembling data from the client:** In first stage, a Multi-Dimensional Data Model collects correct data from the client. Mostly, software professionals provide simplicity to the client about the range of data which can be gained with the selected technology and collect the complete data in detail.

**Stage 2: Grouping different segments of the system:**In the second stage, the Multi-Dimensional Data Model recognizes and classifies all the data to the respective section they belong to and builds it problem-free to apply step by step.

**Stage 3: Noticing the different proportions:** In the third stage, it is the basis on which the design of the system is based. In this stage, the main factors are recognized according to the user’s point of view. These factors are also known as “Dimensions”.

**Stage 4: Preparing the actual-time factors and their respective qualities:** In the fourth stage, the factors which are recognized in the previous step are used further for identifying the related qualities. These qualities are also known as **“attributes”** in the database.

**Stage 5: Finding the actuality of factors which are listed previously and their qualities:**In the fifth stage,A Multi-Dimensional Data Model separates and differentiates the actuality from the factors which are collected by it. These play a significant role in the arrangement of a Multi-Dimensional Data Model.

Stage 6: Building the Schema to place the data, with respect to the information collected from the steps above: In the sixth stage, on the basis of the data, which was collected previously, a Schema is built. M

**Advantages of Multi-Dimensional Data Model**

* It is easy to maintain.
* Its performance is better than that of normal databases (e.g. relational databases).
* The representation of data is better than traditional databases. That is because the multi-dimensional databases are multi-viewed and carry different types of factors.
* It is workable on complex systems and applications, contrary to the simple one-dimensional database systems.
* The compatibility in this type of database is an upliftment for projects having lower bandwidth for maintenance staff.

**Disadvantages of Multi-Dimensional Data Model**

* The multi-dimensional Data Model is slightly complicated in nature, and it requires professionals to recognize and examine the data in the database.
* During the work of a Multi-Dimensional Data Model, when the system caches, there is a great effect on the working of the system.
* It is complicated in nature due to which the databases are generally dynamic in design.
* The path to achieving the end product is complicated most of the time.
* As the Multi-Dimensional Data Model has complicated systems, databases have many databases due to which the system is very insecure when there is a security break.

## ETL

ETL is a type of data integration that refers to the three steps (extract, transform, load) used to blend data from multiple sources. **It's often used to build a data warehouse**. During this process, data is taken (extracted) from a source system, converted (transformed) into a format that can be analyzed, and stored (loaded) into a data warehouse or other system. Extract, load, transform (ELT) is an alternate but related approach designed to push processing down to the database for improved performance.

## JSON & XML

JSON:

* Stands for **J**ava**S**cript **O**bject **N**otation
* Is a lightweight format for storing and transporting data
* Is often used when data is sent from a server to a web page
* Is "self-describing" and easy to understand

Graphical user interface, text

Description automatically generatedJSON example:

XML:

* stands for eXtensible Markup Language
* Is a markup language much like HTML
* Was designed to store and transport data
* Graphical user interface, text, application

  Description automatically generatedWas designed to be self-descriptive

Graphical user interface, text

Description automatically generated with medium confidenceXML Example: is displayed as 🡪

## NoSQL

NoSQL databases are purpose built for specific data models and have flexible schemas for building modern applications. NoSQL databases are widely recognized for their ease of development, functionality, and performance at scale. These types of databases are optimized specifically for applications that require large data volume, low latency, and flexible data models, which are achieved by relaxing some of the data consistency restrictions of other databases.

In a NoSQL database, a book record is usually stored as a JSON document. For each book, the item, ISBN, Book Title, Edition Number, Author Name, and AuthorID are stored as attributes in a single document. In this model, data is optimized for intuitive development and horizontal scalability.

NoSQL databases are a great fit for many modern applications such as mobile, web, and gaming that require flexible, scalable, high-performance, and highly functional databases to provide great user experiences.

* Flexibility: NoSQL databases generally provide flexible schemas that enable faster and more iterative development. The flexible data model makes **NoSQL databases ideal for semi-structured and unstructured data.**
* Scalability: NoSQL databases are generally designed to scale out by using distributed clusters of hardware instead of scaling up by adding expensive and robust servers. Some cloud providers handle these operations behind-the-scenes as a fully managed service.
* High-performance: NoSQL database are optimized for specific data models and access patterns that enable higher performance than trying to accomplish similar functionality with relational databases.
* Highly functional: NoSQL databases provide highly functional APIs and data types that are purpose built for each of their respective data models.

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application

Description automatically generated with medium confidence

## Regex (Regular Expressions)

/d: digits /:D non-digits UPPERCASE [A-Z[ lowercase [a-z[ numbers [0-9[

Combinations

* Upper, lower and numbers = [a-zA-Z0-9[
* Upper and lower=[a-zA-Z[

Python example of regular expressions There’s a lot of regular expressions to study!!!!!!!!!!!!!!!!!!!!!!!!

Text

Description automatically generated 🡪 Text

Description automatically generated

## Virtual Environments Setup (ISN’T COMPLETED)

First thing you need to do is execute the [venv](https://docs.python.org/3/library/venv.html) module, which is part of the Python standard library.

Graphical user interface, text

Description automatically generated with medium confidence

The only thing left to do is to “activate” our environment by running the following scripts



# Programming (Python and R)

## Install Packages

To Install packages in Python you need to open de Windows Terminal or Windows PowerShell and write the followings command: pip install *package name or* py -m pip install *package name*



To use packages in Rstudio you must activate them in the Packages window

Graphical user interface, application, Word

Description automatically generated

And if the packages aren’t in that window, you have to open Tools 🡪 Install Packages and write the package name

## Factor Analysis

Factor analysis is a technique that is used to reduce a large number of variables into fewer numbers of factors. This technique extracts maximum common variance from all variables and puts them into a common score. As an index of all variables, we can use this score for further analysis. Factor analysis is part of general linear model (GLM) and this method also assumes several assumptions**: there is linear relationship, there is no multicollinearity, it includes relevant variables into analysis, and there is true correlation between variables and factors.** Several methods are available, but principal component analysis (PCA) is used the most.

Assumptions:

**No outlier:** Assume that there are no outliers in data.

**Adequate sample size**: The case must be greater than the factor.

**No perfect multicollinearity:** Factor analysis is an interdependency technique. There should not be perfect multicollinearity between the variables.

**Homoscedasticity:** Since factor analysis is a linear function of measured variables, it does not require homoscedasticity between the variables.

**Linearity:** Factor analysis is also based on linearity assumption. Non-linear variables can also be used. After transfer, however, it changes into linear variable.

**Interval Data**: Interval data are assumed.

There are different types of methods used to extract the factor from the data set:

1. **Principal component analysis**: This is the most common method used by researchers. PCA starts extracting the maximum variance and puts them into the first factor. After that, it removes that variance explained by the first factors and then starts extracting maximum variance for the second factor. This process goes to the last factor.

2. **Common factor analysis**: The second most preferred method by researchers, it extracts the common variance and puts them into factors. This method does not include the unique variance of all variables. This method is used in SEM.

3. **Image factoring**: This method is based on correlation matrix. OLS Regression method is used to predict the factor in image factoring.

4. **Maximum likelihood method**: This method also works on correlation metric, but it uses maximum likelihood method to factor.

5. **Other methods of factor analysis**: Alfa factoring outweighs least squares. Weight square is another regression-based method which is used for factoring.

## Functions

An R function is created by using the keyword **function**. The basic syntax of an R function definition is as follows:

A picture containing text

Description automatically generated

Text

Description automatically generatedWe can create user-defined functions in R. They are specific to what a user wants and once created they can be used like the built-in functions. Below is an example of how a function is created and used.

A picture containing icon

Description automatically generated

🡪

Text

Description automatically generated with medium confidenceIn Python functions work the same as in R, but have a different code

 🡪

If you do not know how many arguments that will be passed into your function, add a \* before the parameter name in the function definition. In the following example the functions will print out the length of the list that will be created by iterating each value passed to the function

Text

Description automatically generated🡪 2

If you do not know how many keyword arguments that will be passed into your function, add two asterisk: \*\* before the parameter name in the function definition.

A screenshot of a computer

Description automatically generated with medium confidence🡪 His last name is Refsnes

Also, If we call the function without argument, it uses the default value:

Text

Description automatically generated🡪 Text

Description automatically generated

## Read Data.

In Python to read data for data science you’ll use the pandas package as pd and call each function for each data type.

Text

Description automatically generated

Each function has its own arguments, but I’ll show you only the most used ones.

A picture containing graphical user interface

Description automatically generated

In Rstudio its much easier to read data. Just click the import dataset option and select your data type

Graphical user interface, application

Description automatically generated

And if you’re working with CSV files you need to call them following code: *read.csv(“data.csv”)*

## Manipulate Data Frames

In this chapter I’m going to introduce you to pandas functions to manipulate data frames in python

Graphical user interface

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with low confidence

Text

Description automatically generated with medium confidence

🡪

Text

Description automatically generated

A picture containing text

Description automatically generatedText

Description automatically generated 🡪

Table

Description automatically generated

And now I’m going to show you some R code to manipulate data frames

Graphical user interface, text, application, table

Description automatically generatedThis code shows us the data type for each column 🡪

Table

Description automatically generated

A picture containing table

Description automatically generatedData types 🡪



Graphical user interface

Description automatically generated with medium confidence

## Subsetting Data Frames

In this chapter I will show you only how to slice your data in R

Text

Description automatically generatedA picture containing text, person, orange

Description automatically generated

Text, letter

Description automatically generated

Text

Description automatically generated with medium confidenceText

Description automatically generated with medium confidence

Text

Description automatically generatedNow the code to subset data in Python

Text

Description automatically generated

# Statistics (and how to calculate them in Python)

For this chapter we are going to work with the same dataset to perform all the statistics in python.

## Descriptive Statistics

The descriptive statistics are a group of different metrics that we can extract from a data set.

* Mean: Average value
* Median: Midpoint between the highest and the lowest values
* Rango: Span of values over which your data occurs
* Standard Deviation of the Mean (σ): On average, how much each measurement deviates from the mean
* Variance (): Is a measure of how spread out a data set is. It is calculated as the average squared deviation of each number from the mean of a data set

In just one line of code, we have the number of values, mean, standard deviation, minimum value, maximum value, the values positioned in 25%, 50% (median) and 75%. For each column.

Graphical user interface, text, application

Description automatically generated

A small standard deviation can be a goal in certain situations where the results are restricted, for example, in product manufacturing and quality control. A particular type of car part that has to be 2 centimeters in diameter to fit properly had better not have a very big standard deviation during the manufacturing process! A big standard deviation in this case would mean that lots of parts end up in the trash because they don’t fit right; either that, or the cars will have major problems down the road. So the more spread out the group of numbers are, the higher the standard deviation.

But in situations where you just observe and record data, **a large standard deviation isn’t necessarily a bad thing**; it just reflects a large amount of variation in the group that is being studied.

For example, if you look at salaries for everyone in a certain company, including everyone from the student intern to the CEO, the standard deviation may be very large. On the other hand, if you narrow the group down by looking only at the student interns, the standard deviation is smaller, because the individuals within this group have salaries that are similar and less variable. The second data set isn’t better, it’s just less variable.

Similar to the mean, **outliers affect the standard deviation** (after all, the formula for standard deviation includes the mean). Here’s an example: the salaries of the L.A. Lakers in the 2009–2010 season range from the highest, $23,034,375 (Kobe Bryant) down to $959,111 (Didier Ilunga-Mbenga and Josh Powell). Lots of variation, to be sure!

The standard deviation of the salaries for this team turns out to be $6,567,405; it’s almost as large as the average. However, as you may guess, if you remove Kobe Bryant’s salary from the data set, the standard deviation decreases because the remaining salaries are more concentrated around the mean. The standard deviation becomes $4,671,508

*The standard deviation has the same units of measure as the original data. If you’re talking about inches, the standard deviation will be in inches*.

Variance is the square of the standard deviation and is the last metric:

Text

Description automatically generated

The more spread out the values are in a dataset, the higher the variance. Therefore, the variance gives more importance to outliers than the standard deviation.

## Histograms

All histograms have the same components but because they have different values they differ from each other’s shape.

* Range
  + How widely dispersed are the frequencies of each bin? Extremely large frequency ranges (particularly as a percentage) may indicate data that is fundamentally unreliable.
  + How wide are the bins themselves? Specifically, how broad are the intervals or how descriptive are the classes? Unusually large or small intervals, or unusually broad or narrow categories may indicate important observations about the data as a whole.
* Frequency Density
  + Frequency is measured by the area of the bar. What that means it that you can use a histogram with different interval or class widths to represent data with varying densities.
* Shape
  + The shape of a histogram can lead to valuable conclusions about the trend(s) of the data

Chart, histogram

Description automatically generated

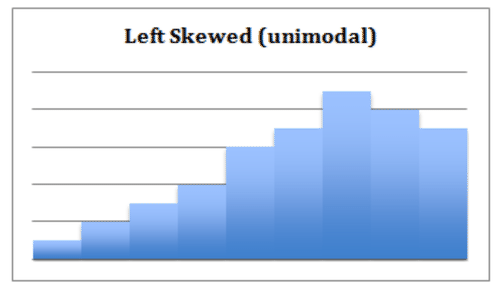
As you can see in the code (🡪), we obtained the histogram for the CrimeRate and Education columns. The shape of the data isn’t a bell because they have some outliers.

We can say that the crime rates between 80 and 120 have more frequency than the others. And Education has more values between 12 and 13 .

Other types of histograms:

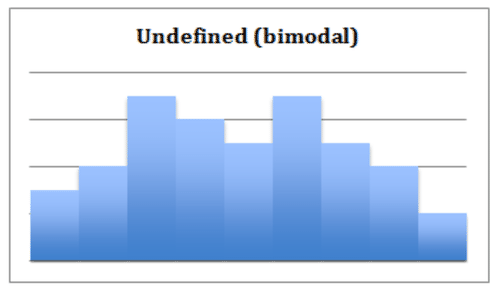
* Chart, histogram

  Description automatically generatedBell-shaped: A histogram with a prominent ‘mound’ in the center and similar tapering to the left and right. One indication of this shape is that the data is unimodal – meaning that the data has a single mode, identified by the ‘peak’ of the curve. **If the shape is symmetrical, then the mean, median, and mode are all the same value**. Note that a normally distributed data set creates a symmetric histogram that looks like a bell, leading to the common term for a normal distribution: a bell curve
* Right or Left Skewed: A skewed histogram has a peak that is left (or right) of center and a more gradual tapering to the other side of the graph. This is a unimodal data set, with the mode closer to the left (or right) of the graph and smaller than either the mean or the median. The mean of right-skewed data will be located to the right side of the graph and will be a greater value than either the median or the mode. This shape indicates that there are a number of data points, perhaps outliers, that are greater than the mode for the left skewed and lesser than the mode for the right

Chart, histogram

Description automatically generated

* Undefined Bimodal: This shape is not specifically defined, but we can note regardless that it is bimodal, having two separated classes or intervals equally representing the maximum frequency of the distribution.



## Percentiles and Outliers

Percentiles is in everyday use, but there is no universal definition for it. The most common definition of a percentile is a number where a certain percentage of scores fall below that a number.

An outlier is an observation that lies an abnormal distance from other values in a random sample.

Chart

Description automatically generatedOne way to visualize the percentiles rank and the outliers is by plotting a boxplot.

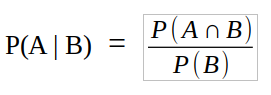
The box limits the 25th and 75th percentiles. The two lines outside the box limit the 10th and 90th percentile and the circles in red show the outliers.

## Probability Theory

The probability theory is a branch of mathematics concerned with the analysis of random phenomena. The outcome of a random event cannot be determined before it occurs, but it may be any one of several possible outcomes. The actual outcome is considered to be determined by chance. This theory has great importance for data science. One needs to possess a comprehensive understanding of the probability theory to be a well-performing data scientist. For instance, probability distributions play a key role in predictive analytics.

**Conditional Probability**

We defined the conditional probability as the probability of event A given that event B has occurred, is denoted as p(A|B).

*“Conditional probability is the likelihood of an event A to occur given that another event that has a relation with event A has already occurred.”*

The formula of the conditional probability:

P(A ∩ B) is the probability that both events A and B occur. P(B) is the probability that event B occurs. The conditional probability is typically not commutative which means that P(A | B) is not equal to P(B | A).

In the following example we see a probability space (Ω) which indicates all the probabilities add up to 1.

Diagram, venn diagram

Description automatically generated

Conditional probability is a fundamental concept in probability theory and statistics. For instance, Bayesian statistics arises from an interpretation of the conditional probability.

A given condition might not have any effect on an event so the conditional and unconditional probabilities are equal (i.e. P( A | B) = P(A)). In such cases, the events **A and B are said to be independent.**

**Joint Probability**

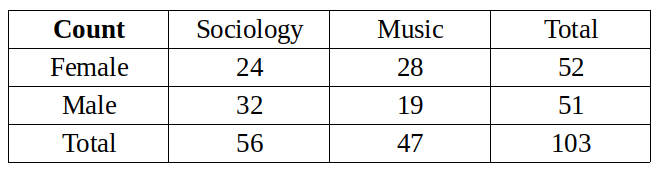
Graphical user interface, application, Word

Description automatically generatedJoint probability is the probability of two events occurring together. If two events are independent, the joint probability is calculated by multiplying the probabilities of each event.

**P(A ∩ B) = P(A) \* P(B)** If we put that in the equation of the conditional probability:

Joint and conditional probability example:

The following table shows the number of female and male students enrolled in the sociology and music classes.



There are 103 students. We will first calculate the unconditional probabilities.

P(Female) = 52 / 103 = 0.505

P(Male) = 51 / 103 = 0.495

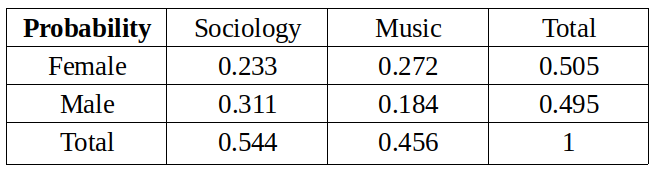
P(Music) = 47 / 103 = 0.456

P(Sociology) = 56 / 103 = 0.544

The joint probabilities can be calculated by dividing the number in a cell by the total number of students. For instance, the probability that a student is female and enrolled in the sociology class:

***~~P(Female ∩ Sociology) = (P(Female) \* P(Sociology)) =24 / 103 = 0.233~~***

We can calculate the other joint probabilities similarly. The following table contains all the probabilities for these events.



We will calculate the conditional probabilities now.

What is the probability that a student is female given that the student is enrolled in the music class?

P(Female | Music) = P(Female ∩ Music) / P(Music) = 0.272 / 0.456 = 0.596

What is the probability that a student is male given that the student is enrolled in the sociology class?

P(Male | Sociology) = P(Male ∩ Sociology) / P(Sociology) = 0.311 / 0.544 = 0.572

## Bayes Theorem

Principled way of calculating a conditional probability without the joint probability.

Specifically, one conditional probability can be calculated using the other conditional probability; for example:

*Firstly, in general, the result P(A|B) is referred to as the posterior probability and P(A) is referred to as the prior probability and sometimes P(B|A) is referred to as the likelihood and P(B) is referred to as the evidence.*

P(A|B) = P(B|A) \* P(A) / P(B)

The reverse is also true; for example:

P(B|A) = P(A|B) \* P(B) / P(A)

It is often the case that we do not have access to the denominator directly, e.g. P(B).

We can calculate it an alternative way; for example:

P(B) = P(B|A) \* P(A) + P(B|not A) \* P(not A)

This gives a formulation of Bayes Theorem that we can use that uses the alternate calculation of P(B), described below::

P(A|B) = P(B|A) \* P(A) / (P(B|A) \* P(A) + P(B|not A) \* P(not A))

**Diagnostic Test Scenario**

Consider a human population that may or may not have cancer (Cancer is True or False) and a medical test that returns positive or negative for detecting cancer (Test is Positive or Negative), e.g. like a mammogram for detecting breast cancer.

Problem: If a randomly selected patient has the test and it comes back positive, what is the probability that the patient has cancer?

**Manual Calculation**

Medical diagnostic tests are not perfect; they have error.

Sometimes a patient will have cancer, but the test will not detect it. This capability of the test to detect cancer is referred to as the sensitivity, or the true positive rate.

In this case, we will contrive a sensitivity value for the test. The test is good, but not great, with a true positive rate or sensitivity of 85%. That is, of all the people who have cancer and are tested, 85% of them will get a positive result from the test.

P(Test=Positive | Cancer=True) = 0.85

Given this information, our intuition would suggest that there is an 85% probability that the patient has cancer.

Our intuitions of probability are wrong.

This type of error in interpreting probabilities is so common that it has its own name; it is referred to as the **base rate fallacy**.

It has this name because the error in estimating the probability of an event is caused by ignoring the base rate. That is, it **ignores the probability of a randomly selected person having cancer,** regardless of the results of a diagnostic test.

In this case, we can assume the probability of breast cancer is low and use a contrived base rate value of one person in 5,000, or (0.0002) 0.02%.

P(Cancer=True) = 0.02%.

We can correctly calculate the probability of a patient having cancer given a positive test result using Bayes Theorem.

Let’s map our scenario onto the equation:

**P(A|B) = P(B|A) \* P(A) / P(B)**

**P(Cancer=True | Test=Positive) = P(Test=Positive|Cancer=True) \* P(Cancer=True) / P(Test=Positive)**

We know the probability of the test being positive given that the patient has cancer is 85%, and we know the base rate or the prior probability of a given patient having cancer is 0.02%; we can plug these values in:

P(Cancer=True | Test=Positive) = 0.85 \* 0.0002 / P(Test=Positive)

We don’t know P(Test=Positive), it’s not given directly.

Instead, we can estimate it using:

P(B) = P(B|A) \* P(A) + P(B|not A) \* P(not A)

P(Test=Positive) = P(Test=Positive|Cancer=True) \* P(Cancer=True) + P(Test=Positive|Cancer=False) \* P(Cancer=False)

Firstly, we can calculate P(Cancer=False) as the complement of P(Cancer=True), which we already know

P(Cancer=False) = 1 – P(Cancer=True)

= 1 – 0.0002

= 0.9998

We can plug in our known values as follows:

P(Test=Positive) = 0.85 \* 0.0002 + P(Test=Positive|Cancer=False) \* 0.9998

We still do not know the probability of a positive test result given no cancer.

This requires additional information.

Specifically, we need to know how good the test is at correctly identifying people that do not have cancer. That is, testing negative result (Test=Negative) when the patient does not have cancer (Cancer=False), called the true negative rate or the specificity.

We will use a contrived specificity value of 95%.

P(Test=Negative | Cancer=False) = 0.95

With this final piece of information, we can calculate the false positive or false alarm rate as the complement of the true negative rate.

P(Test=Positive|Cancer=False) = 1 – P(Test=Negative | Cancer=False)

= 1 – 0.95

= 0.05

We can plug this false alarm rate into our calculation of P(Test=Positive) as follows:

P(Test=Positive) = 0.85 \* 0.0002 + 0.05 \* 0.9998

P(Test=Positive) = 0.00017 + 0.04999

P(Test=Positive) = 0.05016

Excellent, so the probability of the test returning a positive result, regardless of whether the person has cancer or not is about 5%.

We now have enough information to calculate Bayes Theorem and estimate the probability of a randomly selected person having cancer if they get a positive test result.

P(Cancer=True | Test=Positive) = P(Test=Positive|Cancer=True) \* P(Cancer=True) / P(Test=Positive)

P(Cancer=True | Test=Positive) = 0.85 \* 0.0002 / 0.05016

P(Cancer=True | Test=Positive) = 0.00017 / 0.05016

**P(Cancer=True | Test=Positive) = 0.003389154704944**

The calculation suggests that if the patient is informed they have cancer with this test, then there is only 0.33% chance that they have cancer.

And we code all of this in a few lines of code, as it follows:

Text

Description automatically generated

## Random Variables

STATISTIC

There is a dataset where I analyze all the statistics below ()

A fondo:

Eigen values for data science