University of Tripoli

Faculty of Science

Department of Statistics

Data Science: Statistical Analysis of Big Data

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Statistics

Student Name:

Soha E. Elhaddar

Supervisor:

Dr. Abdullatif S. Tubbal

(Assistant Professor)

Fall 2021/2022

“It’s not that I’m so smart...
It’s just that: I stay with problems longer...!!”

Albert Einstein

Dedication:

I am dedicating this thesis to three beloved people who have meant and continue to mean so much to me. Although they are no longer of this world, their memories regulate my life.

First and foremost, to my parents, **Emhemed & Imbarka,** whose love for me knew no bounds, who taught me the value of education, taught me how to follow my passion and never give up on my dreams.

Next, my inspirational teacher, **prof Mohammed Bani**, the best mathematician I have ever seen, who taught me how the most complicated mathematical subjects could be charmed and enjoyable.

I love you all and miss you all beyond words. May Allah grant you Jannat Al-Firdaws.

Amen.

Acknowledgment:

In completing this thesis, many people have helped me. I would like to thank all those who are related to this project.

I want to express my sincere thanks and gratitude to my motivating mentor **Dr Abdullatif Tubbal.** Besides scientific research and academic work, he taught me so much about life, people, and faith. I want to thank him for his belief in me and his endless support; I am lucky to have a teacher like him.

I would like to thank **Prof Ridha Gajah** for introducing this exciting topic to me and encouraging me to complete this work.

I want to thank **Dr Bahlul Shalabi** for his valuable assistance in obtaining the most recent references.

Many thanks to **Prof Ali Al-Amari** for helpful and valuable consulting.

At last, I would like to extend my heartfelt thanks to my family because this project would not have been successful without their support, and dear friends, "**the black cat gang**", who have been with me all the time.

Finally, special thanks to my amazing **Nadra**.

ملخص البحث:

نحن نعيش في عصر البيانات الضخمة، حيث أن تفاعلاتنا اليومية بدأت تنتقل من العالم المادي إلى العالم الرقمي، فإن كل إجراء نتخذه يولد البيانات، المعلومات تتدفق في كل لحظة من أجهزة هواتفنا الذكية، كل ملف نحفظه ، كل تفاعل نقوم به على وسائل التواصل الاجتماعي، البيانات تصدر عنا حتى عندما نقوم بشيء بسيط كسؤال خدمة الخرائط في Google عن الطريق المختصر لأقرب محطة بنزين..!!

علم البيانات هو المفتاح لجعل هذا التدفق للمعلومات مفيد، ببساطة هو فن استخدام البيانات للتنبؤ بسلوكنا المستقبلي، واكتشاف الأنماط المخفية، واستخدامها للمساعدة في توفير المعلومات أو استخلاص استنتاجات ذات مغزى من هذه الموارد الغير مستغلة من البيانات الضخمة. كل ما سبق من تعريفات تعتبر غامضة وضبابية وتتشابه مع مجالات حديثة أخرى مثل التنقيب عن البيانات والتعلم الآلي والذكاء الاصطناعي. وهذا يضعنا أمام التساؤل، ما هي الاختلافات بين هذه المجالات وعلوم البيانات؟ علاوة على ذلك، ما هو علم البيانات كتطبيق؟

على حد علمي، موضوع علم البيانات غير معروف لدى معظم الإحصائيين الليبيين، ولا يوجد بحث ليبي في هذا المجال. لذلك ، ستوفر هذه الرسالة معرفة إضافية للمهتمين بعلوم البيانات، وخاصة الإحصائيين الليبيين، وتعتبر الخطوة الأولى لتقديم علوم البيانات للباحثين في قسم الإحصاء بجامعة طرابلس.

الغرض الأساسي من هذه الأطروحة هو توضيح التعريفات الغامضة لعلم البيانات وإظهار أن الإحصاء هو الأساس وراء جميع نظرياته، وأن المجالات الأخرى تقدم فقط أدوات متقدمة لتطبيق التحليل الإحصائي على كميات هائلة من البيانات. سنركز على التحليلات الإحصائية ومساهماتها في تطبيق علم البيانات من خلال تحليل ومناقشة الخطوات الأساسية لعملية علم البيانات بالتفصيل، باستخدام قاعدة بيانات كرة القدم الأوروبية (2008-2016) كدراسة حالة، باستخدام قاعدة بيانات SQLite و إصدار لغة البرمجة.R 4.1.1

من أجل تطبيق عملية علم البيانات على دراسة الحالة هذه، تم استخدام العديد من التقنيات الإحصائية في هذه الأطروحة لأغراض مختلفة، مثل الإحصاء الوصفي، وفترات الثقة، ونماذج التصميم مثل تصميم التجارب العاملي مع التداخل، وتصميم تجارب العاملي مع القطاعات والتداخل، بالإضافة إلى بعض تقنيات التنقيب عن البيانات مثل التجميع باستخدام خوارزمية K-mean والتصنيف باستخدام خوارزمية شجرة القرا. Decision Tree.

Abstract:

We live in the **Big Data** era; as our daily interactions move from the physical world to the digital world, every action we take generates data, information pours from our mobile devices, our computers, every file we save, and every social media interaction we make, it is even generated when we do something as simple as asking Google for directions to the nearest gas station...!!

Data science is the key to making this flow of information helpful. Simply, it is the art of employing data to predict our future behavior, discover hidden patterns, and use it to help provide information or draw meaningful conclusions from these vast untapped data resources. These vague and misty definitions are shared with other modern fields such as **Data Mining**, **Machine Learning**, and **Artificial Intelligence**. So, what are the differences between these fields and Data Science? Furthermore, what is Data science in practice?

As far as I know, the subject of data science is not well known for most Libyan statisticians, and there is no Libyan research in this field. Therefore, this thesis will provide additional knowledge to those interested in data science, especially Libyan scientists, and consider the first step to introduce data science for researchers in the Department of Statistics at the University of Tripoli.

The primary purpose of this thesis is to declare the vague definitions of data science and show that statistics is the base behind all of its theories, and other fields are just giving advanced tools to apply statistical analysis on enormous amounts of data. We will focus on statistics and its contributions in applying data science by analyzing and discussing (with some details) the fundamental steps of the data science process by using **the European Soccer Database (2008-2016)** as a case study, using **SQLite** data base and **R programming language version 4.1.1.**

In order to apply the data science process to this case study, many statistical techniques have been used in this thesis for different purposes, such as descriptive statistics, confidence intervals, and design models such as **the design of factorial experiments with interaction**, and **design of factorial experiments with blocks and interaction**, in addition to some **data mining techniques** such as **clustering** with **K-mean** algorithm and **classification** with **Decision Tree** algorithm.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER1:

CHAPTER 2:

CHAPTER 3:

CHAPTER 4:

Chapter 1

Chapter 1:

Introduction to Data Science

1.1. Abstract

Have you ever noticed when you are using your smart phone surfing the internet, browsing your favorite social media application, that suddenly found a sponsored advertising about something you are already thinking of buying it?!

This isn’t magic or kind of hypnotism or reading thoughts, it’s a result of applying the roles of a new field of science called “Data Science”, which is the main topic of this thesis. We will try to introduce the meaning, components and applications of the field, as an initiative to spot light on a subject with applications spreading around us, and being a part of every development in our world.

1.2. Introduction

In 2012, an article published on the Harvard Business Review Magazine, described the data scientist as: "the most attractive job of the 21st century" (Davenport and Patil, 2012), with 80 years left in the century, it’s trivial to say they might yet change their minds. Nevertheless, in the moment, data science is getting a lot of attention. Try to type the term “Data Science” on Google and see how many results do you have, it’s about 3.160.000.000 with an ability to increase by time, which exceeds the terms: "Statistics", "Computer Science" and "Mathematics"! So, what is Data Science?

To different people this means different things, but at its core, Data Science is **using data to answering questions** and Data Scientist is **someone who knows how to ask questions which may answered by data and has curiosity to get the answers.** These are a pretty broad definitions, that is because it is a pretty broad field and still carries the aura of a new field, although, its components descend directly from well-established fields, but data science seems to be a fresh aggregation of these pieces into something that is new (Godsey, 2017).

It is challenging to get one common definition for Data Science, because it is related to many fields, such as: statistics, computer science, software engineering and business management. So, its definitions vary between aspects of these fields, for example, as a statistician I can say: "It is a concept to unify statistics, data analysis and their related methods in order to understand and analyze actual phenomena with data" (Hayashi, 1998), additionally from a programmer’s point of view: "It is using programming languages, queries, algorithms and systems to extract knowledge and insights from many structural and unstructured data" (Rusu, 2013). However, these previous definitions of data science are parsimonious since data science is a field going beyond these limited aspects.

Although the indisputable fact that data science is strongly related to so called "Big Data" and from here it gains most of its importance. We live in exciting, even revolutionary times, as our daily interactions move from the physical world to the digital world, every action we take generates data, information pours from our mobile devices, our computers, every file we save and every social media interaction we make, it's even generated when we do something as simple as asking Google for directions to the closest gas station...!!

Sensors and machines collect, store and process information about the environment around us. This flood of information gives us the power to make more informed decisions, react more quickly to change and better understand the world around us. However, it can be a struggle to know where to start when it comes to making sense of this data deluge, you need to establish what you know, what you have, what you can get, where you are and where you would like to be, this last one is of utmost importance; only when you have well-defined goals you can begin to survey the available resources and all the possibilities for moving toward those goals. How do we get the answers from the data to answer our most pressing questions about our businesses, our lives and our world? Data science is the key to make this flow of information useful. Simply, it is the art of employing data to predict our future behavior, discover Hidden patterns and using it to help in providing information or draw meaning conclusions from these vast untapped data resources (Pierson, 2017).

To practice data science in the true meaning of the term, you need the analytical knowledge of mathematics and statistics, the coding skills to work with data and an area of subject matter expertise. Without this expertise you might call yourself a mathematician or a statistician. Similarly, a software programmer without subject matter expertise and analytical knowledge might better be considered a software engineer or developer, but not a data scientist.

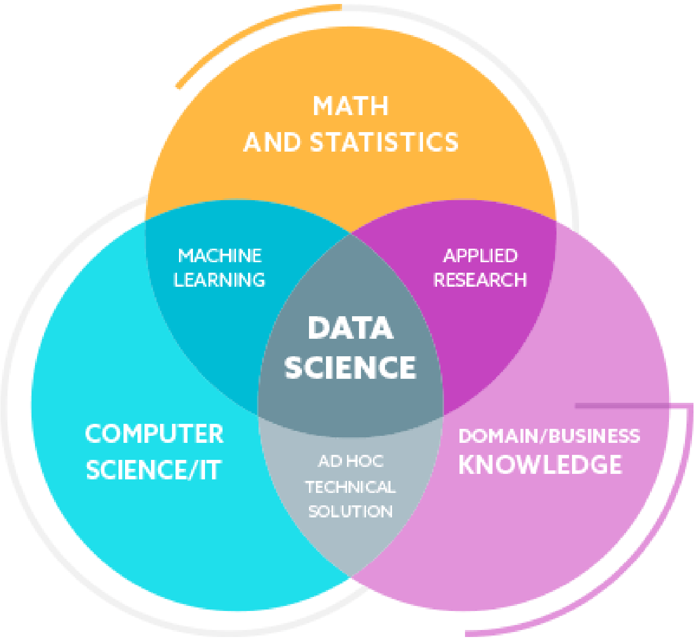


Figure (1-1) Overlapping between Data Science Fields

Data scientists are big data wranglers whose gathering and analyzing large sets of data. Their role combines computer science, statistics and mathematics. They explore, analyze, process and model data, then interpret the results to create actionable plans for companies and other organizations. Many data scientists began their careers as statisticians or data analysts, but as big data began to grow and evolve those roles evolved as well. Data is the key information that requires analysis, creative curiosity and a talent for translating high-tech ideas into new ways to turn it into profits.

We have to confess that traditional statistics degrees are not enough to qualify graduates to deal with the enormous amounts of data, they need some skills and expertise in programming, algorithms design and data structure. Data scientists need some advanced technical tools, for example, they need appropriate programming language such as R, Python, C++, Julia and Java, additionally, they need to query data (write commands to extract relevant datasets from data storage systems), most of the time they use Structured Query Language (SQL) to do that (Peng and Matsui, 2015).

In addition, a good knowledge in management and taking decisions is required, since technical skills are not the only thing that matters; however, data scientists often working on high-level plans, giving complex ideas and charged in making fateful decisions. As a result, it is highly important for them to be effective communicators, leaders and team members as well as being analytical thinkers. Experienced data scientists are tasked with developing a company's best performance, from storing to cleaning and processing data. They work cross functionally with other teams throughout their organization such as marketing, customer success and operations. They are highly sought-after in today's large organizations with big project, their salaries and job growth clearly reflect that (Baruah, 2019).

The revolution of Big Data made a kind of "gold rush" situation, especially after the notable successes scored by brand-name global Information technology enterprises, such as **Google** and **Amazon** currently successes recognized by investors and CEOs (chief executive officer). These successes contributed in rising the name of "Data Science" and increasing the demand of employee with data scientist skills (Donoho, 2017).

Because the demand for data insights is increasing exponentially, every area is forced to adopt data science, as such, different flavors of data science have emerged. The following are just a few titles under which experts of every field are using data science: ad tech (short for advertising technology) data scientist, director of banking digital analyst, clinical data scientist, geoengineer data scientist, geospatial analytics data scientist, political analyst, retail personalization data scientist and clinical informatics analyst in pharmacometrics (Pierson, 2017).

The multi-specialty property of data science caused a long history of argument about its root, is it statistics or computer science or even business management?

1.3. Literature Review:

There is a broad argument about the nature of root of data science, whether it is statistics or computer science. The last ten years witnessed a long controversy about this point, started with the **University of Michigan** initiative in 2013, which announced a $100 million ultimately hiring 35 new staff members to prepare and teach an academic post-graduate program for data science. Many academic statisticians had felt that statistics is being neglected here; science the identified leaders of this initiative are faculty from the Electrical Engineering, Computer Science Department and the School of Medicine. The implicit message in these observations is that statistics is a part of what goes on in data science but not a very big part (Donoho, 2017). Various professional statistics organizations are reacting:

"Aren't we Data Science?" (Davidian, 2013). "A grand debate: is data science just a "rebranding" of statistics?" (Goodson, 2014). "Let us own Data Science!" (Bin Yu, 2014). "Why Do We Need Data Science When We've Had Statistics for Centuries?" (Wladawsky-Berger, 2014)

"Data Science is statistics. When physicists do mathematics, they don't say they're doing number science. They're doing math. If you're analyzing data, you're doing statistics. You can call it data science or informatics or analytics or whatever, but it's still statistics. …You may not like what some statisticians do. You may feel they don't share your values. They may embarrass you. But that shouldn't lead us to abandon the term Statistics" (Broman, 2013).

On the other hand, we can find provocation to neglect contributions of statistics for data science:

"Data Science without statistics is possible, even desirable" (Granville, 2014).

"Statistics is the least important part of data science" (Gelman, 2013).

Nevertheless, there are many different dates and timelines that can be used to trace the slow growth of data science and its current impact on the Data Management industry, these dates can illustrate the importance of many contributions, whether it was from statisticians or computer scientists or even business managers. Some of the more significant ones are outlined below:

In 1962, **John Tukey** wrote about a shift in the world of statistics, saying "… as I have watched mathematical statistics evolve, I have had cause to wonder and to doubt…I have come to feel that my central interest is in data analysis…" (Tukey, 1962), Tukey is referring to the merging of statistics and computers, at a time when statistical results were presented in hours, rather than the days or weeks it would take if done by hand. In 1974, **Peter Naur** authored the Concise Survey of Computer Methods, using the term "Data Science" repeatedly. Naur presented his own complicated definition of the new concept, "The science of dealing with data, once they have been established, while the relation of the data to what they represent is delegated to other fields and sciences." (Naur, 1974). In 1977, **The International Association for Statistics, also known as IASC**. The first phrase of their mission statement reads, "It is the mission of the IASC to link traditional statistical methodology, modern computer technology and the knowledge of domain experts in order to convert data into information and knowledge." (Tewari, 2020).

In the late of 1980's, the term **Knowledge Discovery in Databases KDD** (the process of discovering useful knowledge from a collection of data), was certified into the Conference on Knowledge Discovery and Data Mining, which organized its first workshop in 1989. (KDD-Workshop, 1989)

The **Bloomberg Businessweek Magazine** (previously known as BusinessWeek) in 1994 ran the cover story "Database Marketing" revealing that some news companies had started gathering large amounts of personal information with plans to start strange new marketing campaigns. The flood of data was confusing to company managers who were trying to decide what to do with so much disconnected information (Berry, 1994). After five years **Jacob Zahavi** pointed out the need for new tools to handle the massive amounts of information available to businesses. He wrote "Scalability is a huge issue in data mining… Conventional statistical methods work well with small data sets. Today's databases, however, can involve millions of rows and scores of columns of data… Another technical challenge is developing models that can do a better job analyzing data, detecting non-linear relationships and interaction between elements… Special data mining tools may have to be developed to address web-site decisions." (Zahavi, 1999).

In 2001, **William S. Cleveland** laid out plans for training data scientists to meet the needs of the future. He presented an action plan titled "Data Science: An Action Plan for Expanding the Technical Areas of the Field of Statistics" (Cleveland, 2001). It described how to increase the technical experience and range of data analysts. After that, in 2006, **Hadoop 0.1.0** (an open-source non-relational database) was released. Hadoop was based on **Nutch** (another open-source database). This was an important step in developing fields and applications which depend on big data (Seaman, Chaves and Bugbee, 2017).

In 2011, job listings for data scientists increased by 15,000%. There was also an increase in seminars and conferences devoted specifically to data science and big data. Data science had proven itself to be a source of profits and had become a part of corporate culture (Foote, 2016).

All the previous dates represent the most prominent events that contribute in the rising of data scientist as one of the most important careers around the world. The last ten years have shown up valuable Initiatives such as the **University of Michigan** initiative, which we mentioned it before, followed by similar initiatives from major universities including **University of California Berkeley**, **New York University** and **Massachusetts Institute of Technology (MIT)**, (Donoho, 2017). These programs are developing per year, so is the case for literature of data science in general, there is a large variance between the different books that wrote in the last five years, for that, if you want to become a good data scientist you need to follow up most of new publications in the field.

1.4. Data Science Related Fields:

If you have been tried to learn about data science or generally about big data applications and technologies; you must have been crushed with a various confusing terms and labels of fields that you haven’t hear about before, fields that converge in many aspects and diverges in others.

For example, if you want to learn about some technique such as “Classification”, you will find useful information in data science books in addition to data mining and machine learning books! So, what is the relation between these fields? What are the converges and diverges between them?

To answer these questions, we need to preview some brief definitions of some fields:

1.4.1. Data Mining (DM):

Also known as knowledge discovery in databases, can be defined as the process of analyzing large database repositories and of discovering implicit, but potentially useful information (Han, Kamber, and Pei, 2011). The functions or models of DM can be categorized according to the task performed: association, classification, clustering, and outlier analysis (Hui and Jha, 2000; Kao, Chang and Lin, 2003; Nicholson, 2006b).

DM analysis is based normally on two techniques: traditional statistics and machine learning (Girija and Srivatsa, 2006). Traditional statistics is mainly used for exploring data, data relationships, as well as for dealing with data in large databases (Hand, 1998). Examples of traditional statistics include regression analysis, cluster analysis and discriminate analysis. Machine learning (ML) used to develop algorithms that working to extract information from large datasets. DM benefits from these technologies, but differs from the objective pursued: It’s used to discover new, accurate and useful patterns in the data, looking for meaning and relevant information for the organization or individual who needs it. (Arora, 2020). Note that DM do not give any predictions as data science do.

1.4.2. Machine Learning (ML):

The term Machine Learning was coined by Arthur Samuel, an American pioneer in the field of computer gaming and artificial intelligence in 1959 and he stated that “it gives computers the ability to learn without being explicitly programmed” (Foote, 2019).

ML is the process of discovering algorithms that have improved a simulated experience derived from data. It’s the design, study and development of algorithms that allow machines (such as computers) to learn from data, it uses complex programs which can learn through experience and make predictions, by detecting relations, patterns and any information, without human involvement (Hao, 2018). **It’s a tool to make machines smarter, eliminating the human intervention** (Arora, 2020).

Both DM and ML fall under the aegis of Data Science (DS), which makes sense since they both use data. Both processes are used for solving complex problems, so consequently, many people erroneously use the two terms interchangeably. This isn’t so surprising, considering that ML is sometimes used as a means of conducting useful DM. While data gathered from DM can be used to teach machines, so the lines between the two concepts become a bit blurred. Furthermore, both processes employ the same critical algorithms for discovering data patterns, although their desired results ultimately differ, which will become clear as you read on (Arora, 2020).

1.5. Data Science Process Steps:

The main purpose of this thesis is to declare the vague definitions of data science, and show that statistics is the base behind all of its theories, and other fields are just giving advanced tools to apply statistical analysis on big amounts of data. We will focus on statistics and its contributions in applying data science by analyzing and discussing (with some details) the fundamental steps of data science process which are:

1. **Retrieving Data:** This step includes finding suitable data and getting access to it from the data owner. The result is data in its raw form, which probably needs preparing before it becomes usable.
2. **Data Preparation:** This includes transforming the data from a raw form into data directly usable in your models. To achieve this, you'll detect and correct different kinds of errors in

data (data cleaning), combine data from different sources and different forms (could be numeric, text, image, voice, etc.) and transform it (integration and transformation).

3. **Data Exploration:** The goal of this step is to gain a deep understanding of data by looking for patterns, correlations and deviations based on visual and descriptive techniques. The insights gained from this phase will enable you to start modeling.
4. **Setting the Research Goal:** The main purpose here is making sure all the stakeholders understand the what, how and why of the project. In every serious project, this will result in a project charter. It's step of determination the research questions.
5. **Data Modelling:** Now you attempted to gain insights or make predictions stated in the project charter. It is time to bring out the heavy guns, but remember data science work will teach us that often a combination of simple models tends to outperform one complicated model. If you've done this phase right, you're almost done.
6. **Presentation and Automation:** The last step of the data science process is presenting your results and automating the analysis, if needed. The importance of this step is more apparent in projects on a strategic and tactical levels. Certain projects require performing the business process over and over again, so automating the project will save time (Cielen, Meysman and Ali, 2016).

Retrieving and preparing data will be discussed in chapter 2 of this thesis, since exploring data and setting goals will be illustrated in details in chapter 3, while data modelling showed in chapter 4; finally, conclusions and recommendations will be presented on chapter 5.

To illustrate the previous steps in details, we will introduce a case study with accurate big data stored in an **SQL database**, and use the **R programming language** to apply the process, and answer the research questions:

1. What is Data Science?
2. What are the steps of Data Science project?
3. What is the role of Statistics in Data Science?
4. How to become a Data Scientist?

Unfortunately, because of modernity of the field and absence of common academic program for educating data science, it was difficult to collect the material of this thesis, especially with the variety resources and the various backgrounds of authors whose wrote about this subject.

Nevertheless, we tried to make a collection of the most frequent topics using benefits of top courses of the subject and their references in addition to the most ranking books and articles.

Chapter 2

Chapter 2:

Retrieving and Preparing Data

The goal of this chapter is to define the first two steps of the data science project:

1. Collecting, retrieving, or getting data.
2. Preparing or cleaning data.

By introducing the case study of this thesis and its resources, understanding its data and case subject.

2.1 Retrieving Data:

As we mentioned earlier, we live in the data era, where data flows from every device with every step it takes; However, the multiplicity of data sources, obtaining it remains difficult to reach and not as easy as imagined.

The method of collecting data depends on your project's goal; for example, if the goal is to study certain phenomena for scientific, academic, or even personal purposes, you may have to collect data by yourself in a convenient way that helps project goals. This may expose you to deal with different data sources then merging them; in this case, the data will be under your restrictions, even if restricting big data under certain conditions is very difficult.

While often, the goal of your project is to meet the requests of a company, organization, or any entity that stores its data to get benefits of any information that can be extracted from it to develop its work and perform its job perfectly, in this case, you will be exposed to work on data which is free from your restrictions as a researcher and restricted by a mechanism that used for storage. It is like recycling; you deal with a huge amount of fuggy, unorganized, unclear, and indefinite garbage. You have to organize, clean, and explore to extract the most information you can get from it. (Godsey, 2017).

Since the several sources of data, there are many different types of it, such as many different ways to store it, and each of them tends to require different tools and techniques to deal with. Here are some examples of types of data:

- ❖ **Structured data:** is data that often easy to store in tables within databases or Excel files.

- **Unstructured data**: that is not easy to fit into a data model and is not easy to store in tables.
- **Natural language:** is a special type of unstructured data, such as text files.
- **Machine-generated data:** is information that is automatically created by a computer, process, application, or other machines without human intervention.
- **Audio, video, and images.** (Cielen, Meysman, and Ali, 2016)

To make big data studies easier, some data websites have been appeared, which provide vast amounts of accredited real data in different fields on open sources, such as **World Bank Data, Google Public Data, World Health Organization, UNICEF Data,** and **Kaggle**.

Let us focus on **Kaggle**, which is the resource of data that we will use as a case study of this thesis.

Kaggle, a subsidiary of **Google LLC**, is an online community of data scientists and machine learning practitioners, allowing users to find and publish data sets, explore and build models in a web-based data science environment.

While surfing Kaggle, you will meet huge data sets from different fields and sources worldwide, such as economics, health, business, entertainment, and education. etc. we have chosen **European Soccer Database** to be the case study of this thesis, which including all information about:

- More than 25,000 matches.
- More than 10,000 players.
- 11 European Countries with their lead championship.
- Seasons from 2008 to 2016.
- Players and Teams' attributes are sourced from EA Sports' FIFA video game series, including the weekly updates.
- Team line up with squad formation (X, Y coordinates).
- Betting odds from up to 10 providers.
- Detailed match events (goal types, possession, corner, cross, fouls, cards, etc.) for more than 10,000 matches.

Anyone may wonder why we have chosen this data set specifically? Firstly, the number of data set observations exceeds a half million, including more than 100 variables, which verifying the huge data specifications. Secondly, it is a semi-structured raw data, with many unorganized subsets, many missing values, and many unreadable values. So, it can be an example to explain how a data scientist works on cleaning and preparing a massive data set for the analysis. Finally, European Soccer matches have wide popularity, so it can be more understandable for most readers, which gives them a chance to focus on comprehending the data scientist's work. In contrast, the main purpose of this thesis is to understand the data science process instead of studying a specific phenomenon.

The data stored in **SQLite** file, distributed between seven tables, represent countries, leagues, matches, teams, team attributes, players, and player attributes.

Table (2-1) Database description

Table	Obs.	Variables
Country	11	2
League	11	3
Match	25979	115
Player	11060	7
Player Attributes	183978	42
Team	299	5
Team Attributes	1458	25

Since the **R Programming Language** will be the main programming tool for this thesis, data has been imported from **SQLite** to **R** and stored in seven data frames in its original form.

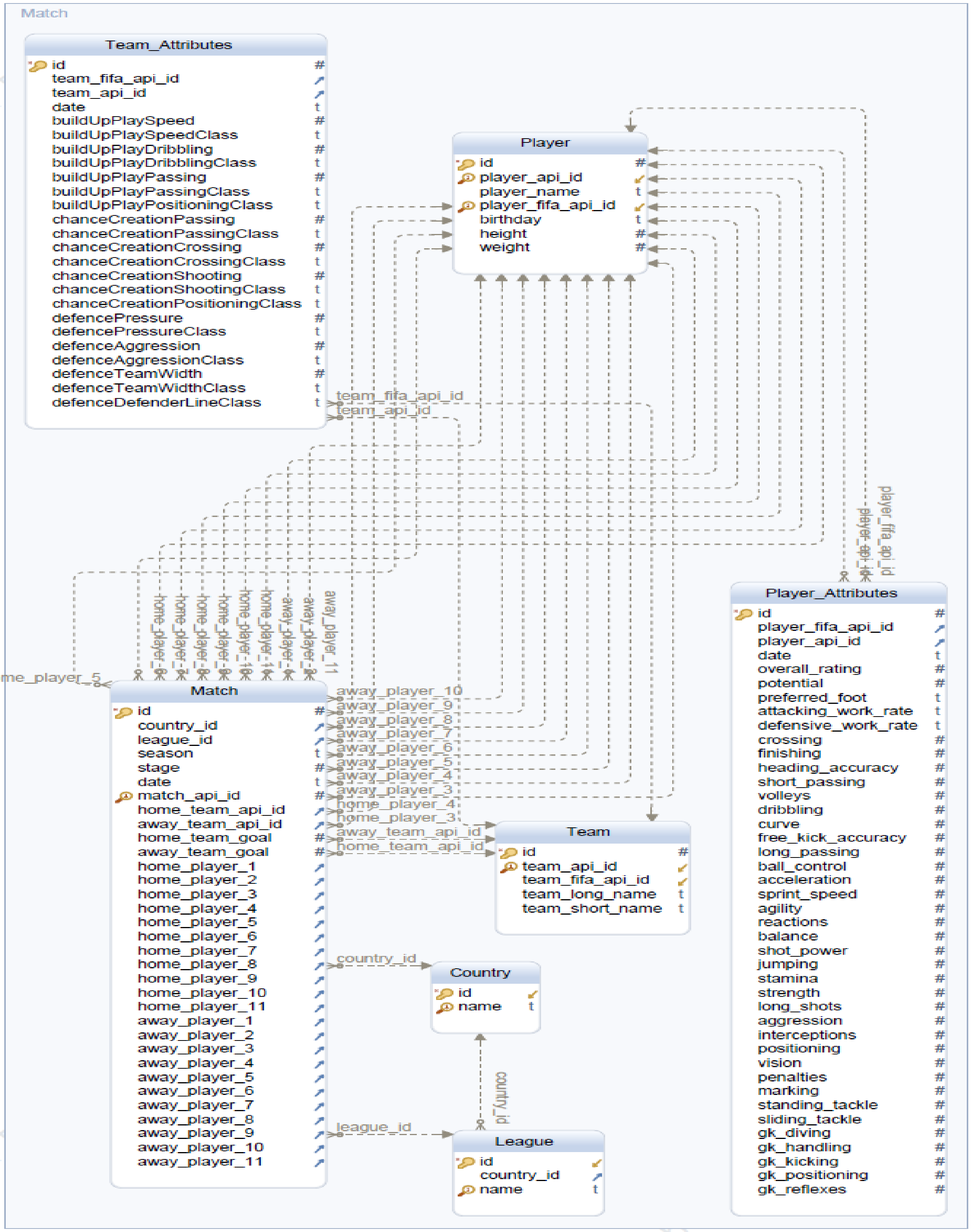


Figure (2-1) SQLite Tables Diagram

2.2 Preparing Data:

The data received from the database is like “a diamond in rough” (Cielen, Meysman, and Ali, 2016); it could not be able to analyze; it needs some adjustments and conformations to get ready for the analysis. This process is called “preparing data” since this call varying from one author to another; some books called it “preprocessing step,” while others called it “data wrangling.”

One definition of wrangling is “having a long and complicated dispute.” That sounds about right. Data wrangling is converting data from complex, unstructured, or otherwise arbitrary formats into something that conventional data analysts can use, which may be called “**Tidy Data**”. (Godsey, 2017).

Before defining tidy data and how to reach it, you need to know that there are four things you should have to accomplish this step of the data science process:

1. Raw data. (Which we already have)
2. Tidy data.
3. A codebook describing each variable and its values in the tidy data set.
4. An explicit and exact recipe that you used to go from 1 to 2 and 3.

So, what is tidy data? It is structured data stored in a table or more, and confirms these conditions:

1. Each measured variable should be in one column.
2. Each observation of that variable should be in a distinct row.
3. There should be one table for each kind of variable.
4. If you have multi-tables, they should include a column in the table that allows them to be linked (i.e., the id variables in database). (Wickham, 2014).

In other words, preparing data is converting raw data to tidy data, also, this process is divided into three procedures:

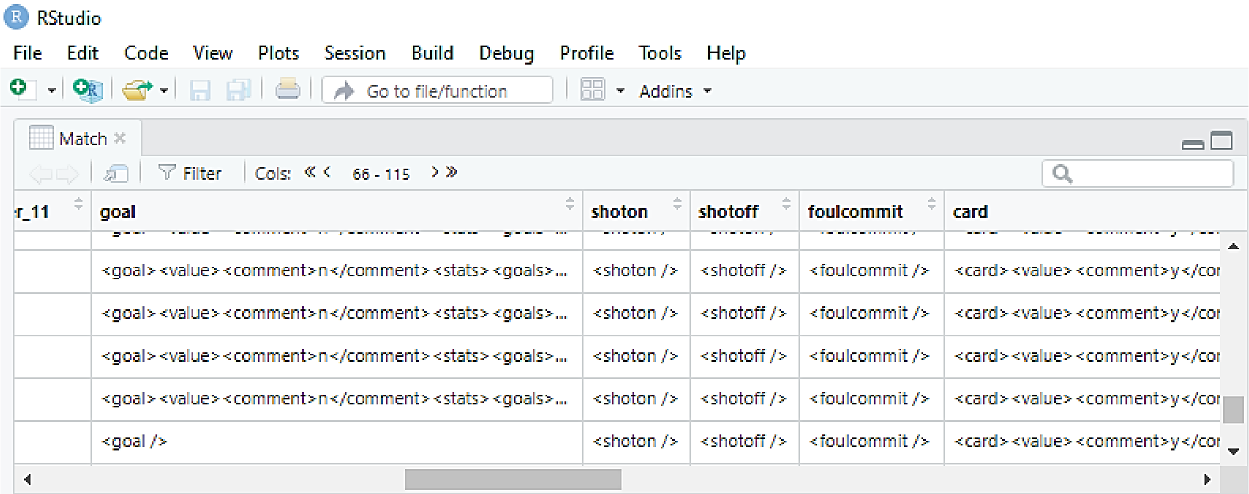
2.2.1 Cleaning:

Data cleaning focuses on removing errors from data, such as missing values, wrong entry values, unreadable values, undesirable outliers, and may be undesirable variables. All these and more are some examples of data errors that must be cleaned.

In the Soccer data, there are too many errors varying between missing values, unreadable values, and undesirable variables; for example, not limit:

In the Match table, there are some variables with unreadable values, their class defined as character, but when we took a look at them, we found unreadable values:

```
> summary (Match [, (78:85)])
     goal               shoton              shotoff             foulcommit       
 Length:25979       Length:25979       Length:25979       Length:25979      
 Class :character   Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character   Mode  :character  
     card               cross               corner             possession       
 Length:25979       Length:25979       Length:25979       Length:25979      
 Class :character   Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character   Mode  :character  
```

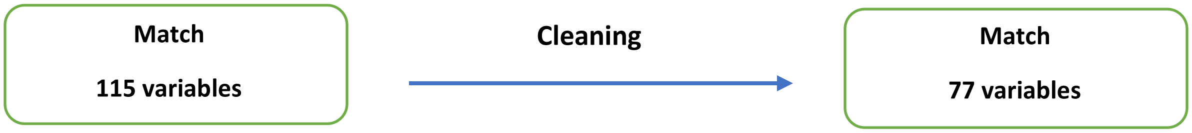

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Match
Filter
Cols: 66 - 115

r_11	goal	shoton	shotoff	foulcommit	card
	<goal><value><comment>n</comment><stats><goals>...	<shoton />	<shotoff />	<foulcommit />	<card><value><comment>y</cor
	<goal><value><comment>n</comment><stats><goals>...	<shoton />	<shotoff />	<foulcommit />	<card><value><comment>y</cor
	<goal><value><comment>n</comment><stats><goals>...	<shoton />	<shotoff />	<foulcommit />	<card><value><comment>y</cor
	<goal><value><comment>n</comment><stats><goals>...	<shoton />	<shotoff />	<foulcommit />	<card><value><comment>y</cor
	<goal />	<shoton />	<shotoff />	<foulcommit />	<card><value><comment>y</cor

Figure (2-2) Unreadable Values in Match Table

These variables and others have been deleted. Another example on the “Match” table is thirty variables represent the odds in soccer betting, which are undesirable variables, so they have been deleted too.

After we applied the cleaning procedure on the Match table remained 77 variables that able to be analyzed.

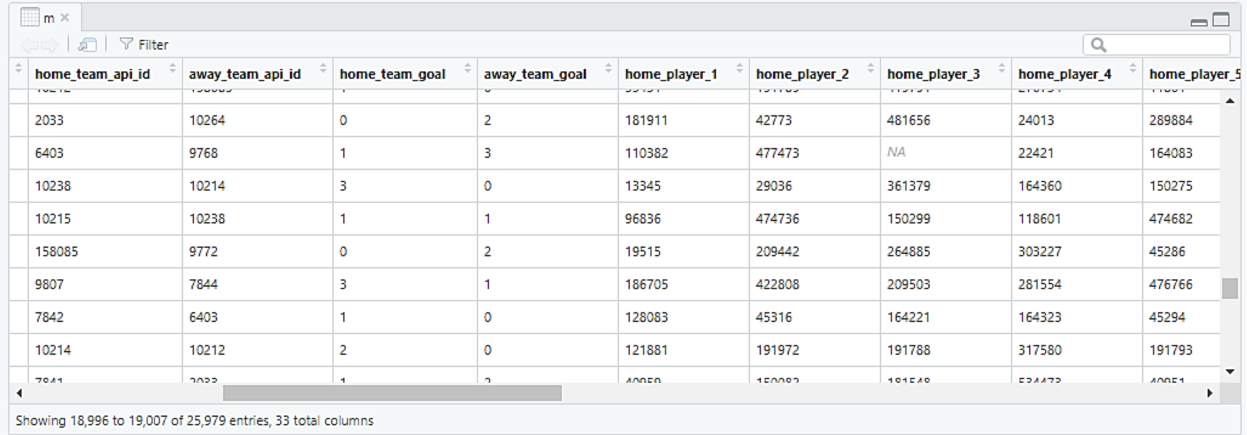


2.2.2 Integration and Reduction:

Integration is merging data from different sources; for example, we can merge two or more tables in our case study such as League and Match by using the (id) variables. Note that the Match has no variable represents the name of league that every match belongs to but has the league's id, and after merging, every single match represented with its league's name:

```
> Match<-merge (League, Match, by.x = "id", by.y = "league_id")
```

Reduction is reducing data dimensionality by aggregate some variables or observations. For example, in the Match table, there are variables that represent home team goals, away team goals, home team id, away team id, and about 66 variables for players ids distributed as their position on the yard. These variables may be aggregated in three variables: team goals, team id, and player id, and distinctive by a new variable for the state (home or away).

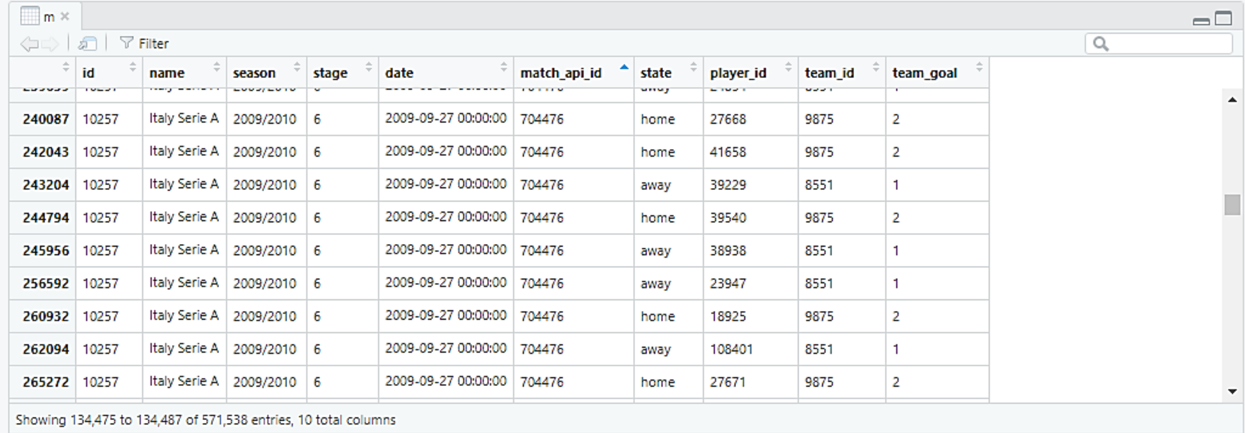


home_team_api_id	away_team_api_id	home_team_goal	away_team_goal	home_player_1	home_player_2	home_player_3	home_player_4	home_player_5
2033	10264	0	2	181911	42773	481656	24013	289884
6403	9768	1	3	110382	477473	NA	22421	164083
10238	10214	3	0	13345	29036	361379	164360	150275
10215	10238	1	1	96836	474736	150299	118601	474682
158085	9772	0	2	19515	209442	264885	303227	45286
9807	7844	3	1	186705	422808	209503	281554	476766
7842	6403	1	0	128083	45316	164221	164323	45294
10214	10212	2	0	121881	191972	191788	317580	191793

Showing 18,996 to 19,007 of 25,979 entries, 33 total columns

Figure (2-3) Match Table before Reduction

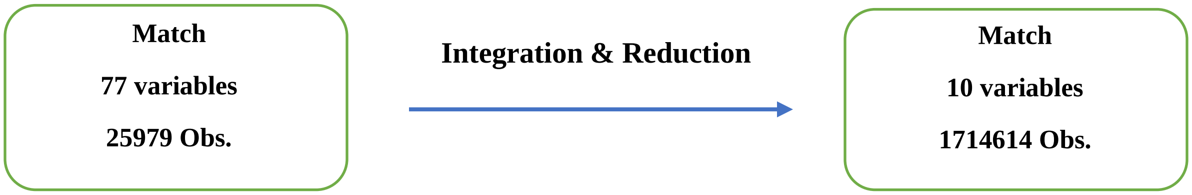
```
> Match <- Match %>% gather (player_state, player_id, -(1:11)) %>% separate (
player_state, c (“state”, “player”, “num”))
> Match <- Match [, -(13:14)]
> Match <- mutate (Match, team_id=1: 1714614, team_goal=1: 1714614)
> for (j in 1: 1714614) {
+   if (Match[j,12] == "home”) {
+     Match$team_goal[j]<-Match[j,10]
+     Match$team_id[j]<- Match [j,8]
+   }

+   if (Match[j,12] == "away") {
+     Match$team_goal[j]<- Match [j,11]
+     Match$team_id[j]<- Match [j,9]
+   }
+ }
> Match <- Match [, -(8:11)]
```



	id	name	season	stage	date	match_api_id	state	player_id	team_id	team_goal
240087	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	home	27668	9875	2
242043	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	home	41658	9875	2
243204	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	away	39229	8551	1
244794	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	home	39540	9875	2
245956	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	away	38938	8551	1
256592	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	away	23947	8551	1
260932	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	home	18925	9875	2
262094	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	away	108401	8551	1
265272	10257	Italy Serie A	2009/2010	6	2009-09-27 00:00:00	704476	home	27671	9875	2

Showing 134,475 to 134,487 of 571,538 entries, 10 total columns

Figure (2-4) Match Table after Reduction



Match

77 variables

25979 Obs.

Integration & Reduction

Match

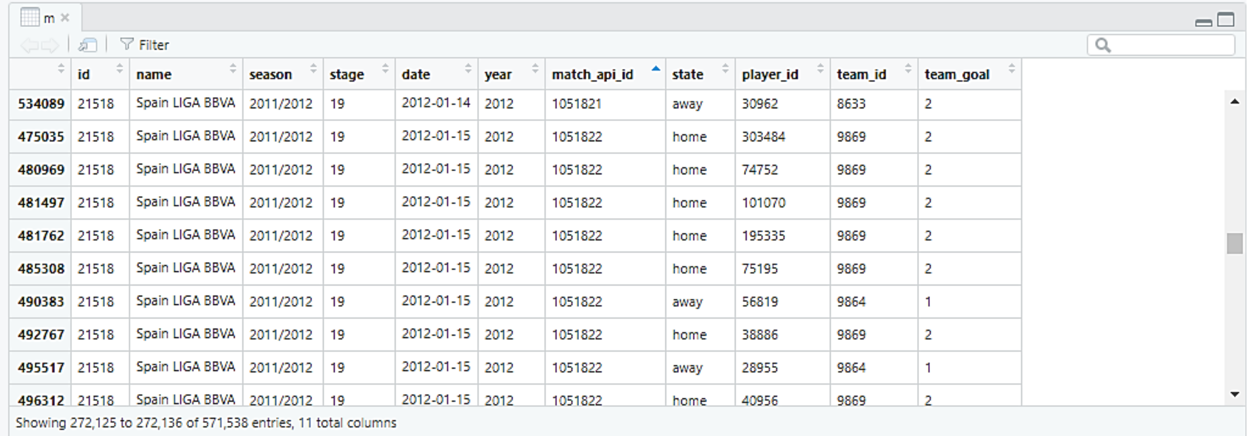
10 variables

1714614 Obs.

2.2.3 Transformation:

Some models and functions required data in a specific shape or specific class. Wherefore, sometimes we need to modify the data to be in a suitable form for modeling. For example, in some situations, we use the logarithmic function to transform nonlinear data to fits linear models. Also, we use standardization to fits models that require normality. In other situations, we may change the class of some variables as needed. For example, the date variables in our data are stored as characters; so, to get benefits from the advantage of date information; we have to change the class from "character" to "date" so we can determine the year, month, or even day from the date.

```
> Match$date<-as.Date (Match$date)
> Match<-mutate (Match, year= year (Match$date))
```



	id	name	season	stage	date	year	match_api_id	state	player_id	team_id	team_goal
534089	21518	Spain LIGA BBVA	2011/2012	19	2012-01-14	2012	1051821	away	30962	8633	2
475035	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	303484	9869	2
480969	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	74752	9869	2
481497	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	101070	9869	2
481762	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	195335	9869	2
485308	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	75195	9869	2
490383	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	away	56819	9864	1
492767	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	38886	9869	2
495517	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	away	28955	9864	1
496312	21518	Spain LIGA BBVA	2011/2012	19	2012-01-15	2012	1051822	home	40956	9869	2

Showing 272,125 to 272,136 of 571,538 entries, 11 total columns

Figure (2-5) Prepared Match Table

Before moving to the next step of the data science process, there are some points we need to mention about data preparation:

1. Data preparing is not a task with specific steps. Every case is different and takes its own procedures to get tidy data.
2. Some functions may help in starting with preparing, such as **str ()**, **summary ()**, **head ()**, **tail ()**, and **quantile ()**. All these functions may represent some general information about data, so you can decide where to start.
3. The data preparation does not finish at a specific stage; It continues in all subsequent steps. For example, to explore data, you may want to plot a graph that views the top five teams with the highest scores of goals and the number of their wins. In this case, the required information is divided between two tables (Match and Team), so you need to merge them and create new variables to calculate the total goals and number of wins for each team. Although the idea of plotting a single graph is simple, it can take many steps in preparation to do it.
4. Many may think that this step does not depend on statistical methods as much as it relies on some skills of programming. But in reality, it needs a statistician with high experience in dealing with data to make it happen perfectly.
5. Finally, note that several packages contain functions used in the data preparation. We used, for example, **plyr**, **dplyr**, **tidyr**, and **Lubridate** packages.

Chapter 3

Chapter 3:

Exploring Data and Setting Goals

This chapter is about defining the third and fourth steps of the data science process:

1. Exploring data or exploratory analysis.
2. Asking the main questions and setting goals for the project.

First of all, we need to mention that the order of data science process steps is not fixed; it depends on the data situation; for example, some companies may want to use their massive data to answer specific questions. In this case, “setting goals” will be the first step in the data science process. However, in other situations, they do not know which questions to ask and what are benefits they may gain from this massive amount of data. In this case, some exploration is needed to recognize the power points in data and to determine which useful questions can be answered by this data (Cielen, Meysman, and Ali, 2016).

The last situation is conforming to our case study, so we will do some exploratory analysis to know which questions to ask and set some valuable goals.

3.1. Exploratory Analysis:

The data scientist **Roger D. Peng** used fantastic imagery to explain the meaning of exploratory analysis: “Have you ever gotten a present before the time when you were allowed to open it? Sure, we all have. The problem is that the present is wrapped, but you desperately want to know what is inside. What is a person to do in those circumstances? Well, you can shake the box a bit, maybe knock it with your knuckle to see if it makes a hollow sound, or even weigh it to see how heavy it is. this is how you should think about your dataset before you start analyzing it for real.” (Peng and Matsui, 2016)

So, exploring data is like knocking on data to hint about the nature of patterns hiding inside. It helps to look at data before making any assumptions by using graphical techniques as the primary tool; information becomes much easier to understand when pictures are used.

The visualization techniques used in this step ranges from simple line graphs or histograms to more complicated graphs, while sometimes it is helpful to draw a composite graph from simple graphs to get a close look at variables that depend on some factors (Cielen, Meysman, and Ali, 2016).

Although graphing is the more common in the exploration step, but it is not the only tool that could be used. Exploration analysis helps for collect information about outliers, variables distributions, categorical variables behavior, interesting relations among variables, and valuable comparisons about some critical parameters. So, it is time to call descriptive statistics and statistical inference.

3.1.1 General View on the Data:

After we defined the meaning of exploratory analysis, now it is time to have a look on **European Soccer Data**, which represent eleven leagues with different number of teams, players and matches for every league:

Table (3-1) Number of teams, players, and matches of each league

	League's Name	Teams	Players	Matches
1	Belgium Jupiler League	24	939	1728
2	England Premier League	34	1229	3040
3	France Ligue 1	35	1281	3040
4	Germany 1. Bundesliga	30	1041	2448
5	Italy Serie A	32	1263	3017
6	Netherlands Eredivisie	25	1054	2448
7	Poland Ekstraklasa	22	618	1920
8	Portugal Liga ZON Sagres	29	1132	2052
9	Scotland Premier League	17	853	1824
10	Spain LIGA BBVA	33	1306	3040
11	Switzerland Super League	15	622	1422

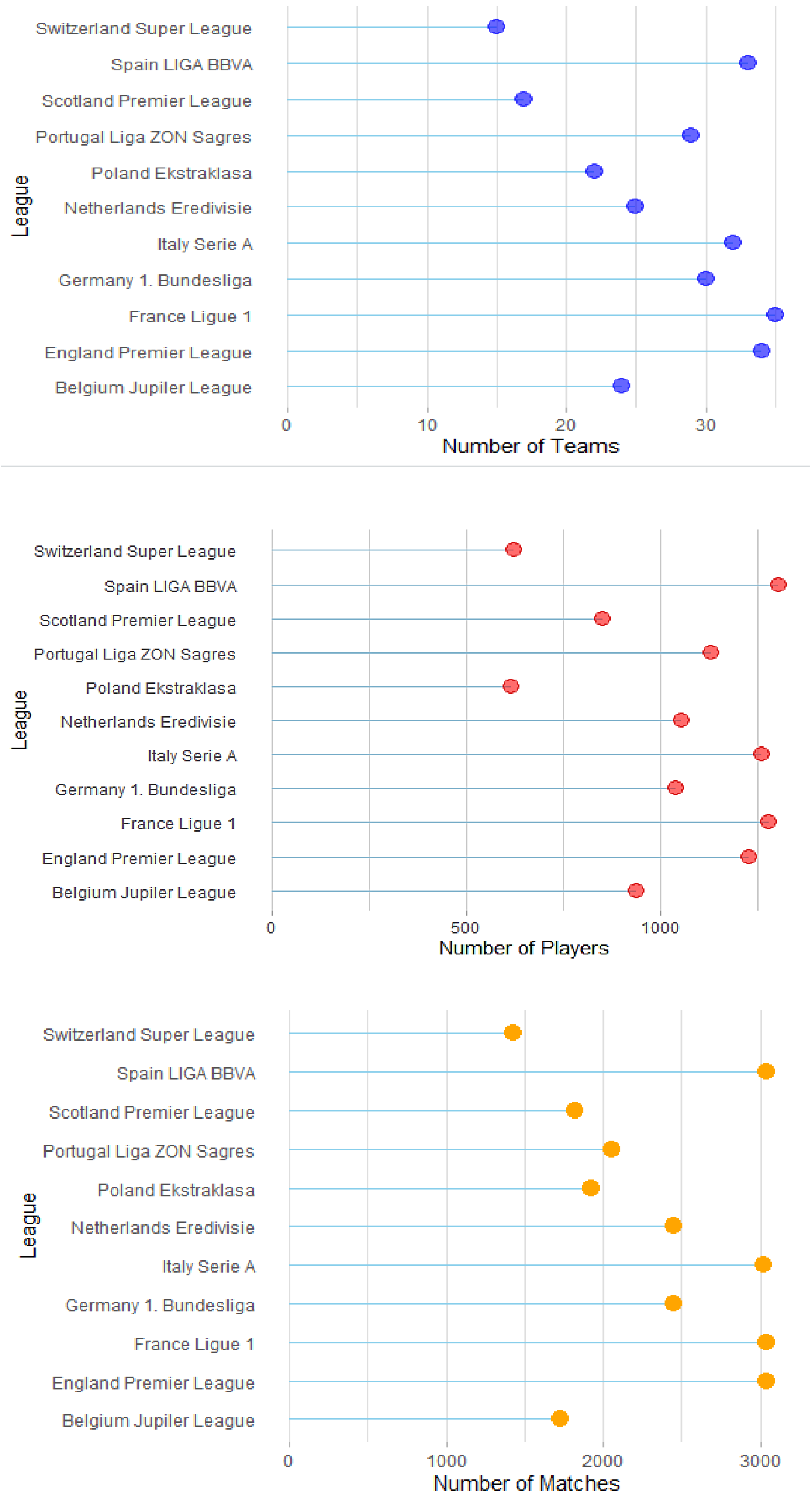


Figure (3-1) Number of teams, players, and matches of each league

Note that, number of players and matches depends on the number of teams for every league; so, four leagues have the highest number of teams, players, and matches: France Ligue 1, England Premier League, Spain LIGA BBVA, and Italy Serie A.

3.1.1.1. General View on Teams’ Data:

As we mentioned before, the database contains two tables for teams. One has general information about every team. The second has information about team attributes that comprise evaluations of the team’s features, such as building up the play, chance creation, and defense. Twenty-one variables represent all these attributes, some of them qualitative and others are quantitative, as shown in **appendix [1]**; so, we are going through study all of them with some details.

Table (3-2) Percentages of team’s qualitative attributes

	Team’s Qualitative Attributes	Classes			Median
1	Build up play speed	Slow 6.996%	Balanced 81.21%	Fast 11.797%	Balanced
2	Build up play dribbling	Little 68.86%	Normal 29.698%	Lots 1.44%	Little
3	Build up play passing	Long 6.45%	Mixed 84.77%	Short 8.78%	Mixed
4	Build up play positioning	Free form 4.94%	Organized 95.06%		Organized
5	Chance creation passing	Risky 11.73%	Normal 84.43%	Safe 3.48%	Normal
6	Chance creation crossing	Little 3.57%	Normal 81.96%	Lots 14.47%	Normal
7	Chance creation shooting	Little 2.54%	Normal 83.95%	Lots 13.51%	Normal
8	Chance creation positioning	Free form 10.23%	Organized 89.78%		Organized
9	Defense pressure	Deep 10.56%	Medium 85.25%	High 4.18%	Medium
10	Defense aggression	Double 6.79%	Press 87.38%	Contain 5.83%	Press
11	Defense team width	Narrow 4.18%	Normal 88.2%	Wide 7.6%	Normal
12	Defense defender line	Cover 93.42%	Offside trap 6.58%		Cover

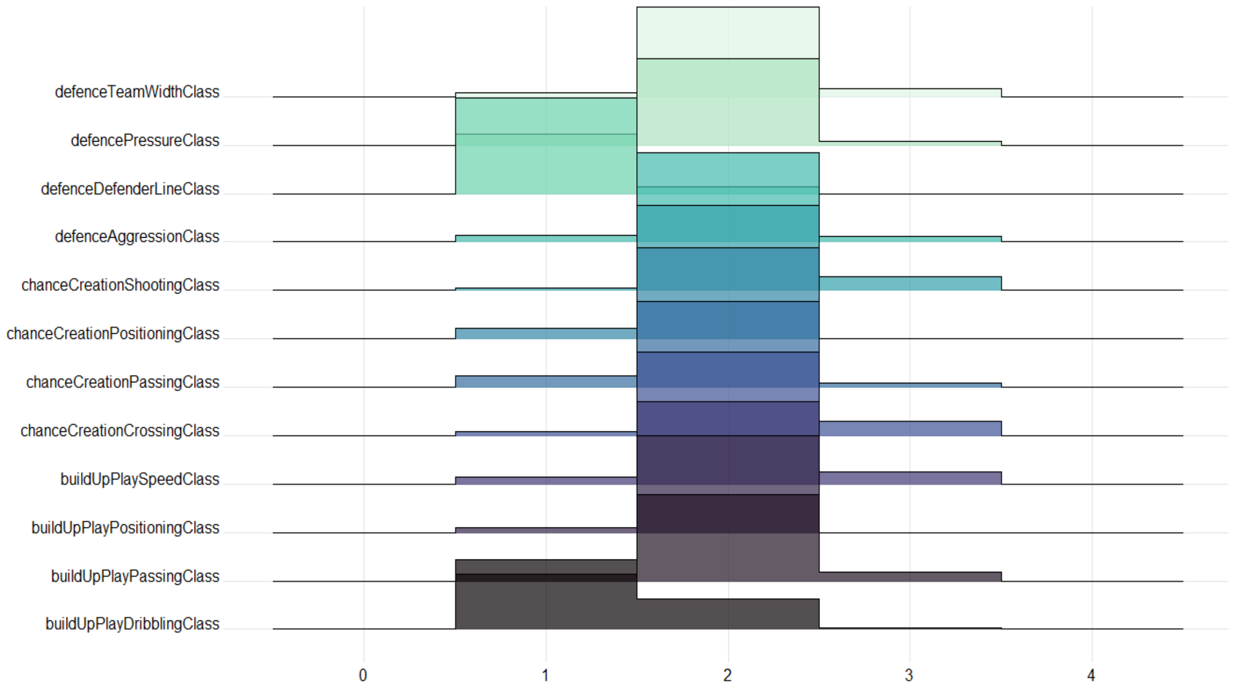


Figure (3-2) Distribution of team's qualitative attributes

Table (3-3) Descriptive statistics of team's quantitative attributes

	Team's Quantitative Attributes	**Min.**	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max.**	**SD**	**95% C.I for Mean**
1	Build up play speed	20	45	52	52.46	62	80	11.546	51.869, 53.055
2	Build up play dribbling	24	42	49	48.61	55	77	9.678	48.19, 49.027
3	Build up play passing	20	40	50	48.49	55	80	10.678	47.93, 49.05
4	Chance creation passing	21	46	52	52.17	59	80	10.36	51.633, 52.697
5	Chance creation crossing	20	47	53	53.73	62	80	11.087	53.16, 54.301
6	Chance creation shooting	22	48	53	53.97	61	80	10.328	53.438, 54.4997
7	Defense pressure	23	39	45	46.02	51	72	10.227	45.49, 46.54
8	Defense aggression	24	44	48	49.25	55	72	9.738	48.751, 49.75
9	Defense team width	29	47	52	52.19	58	73	9.57	51.694, 52.678

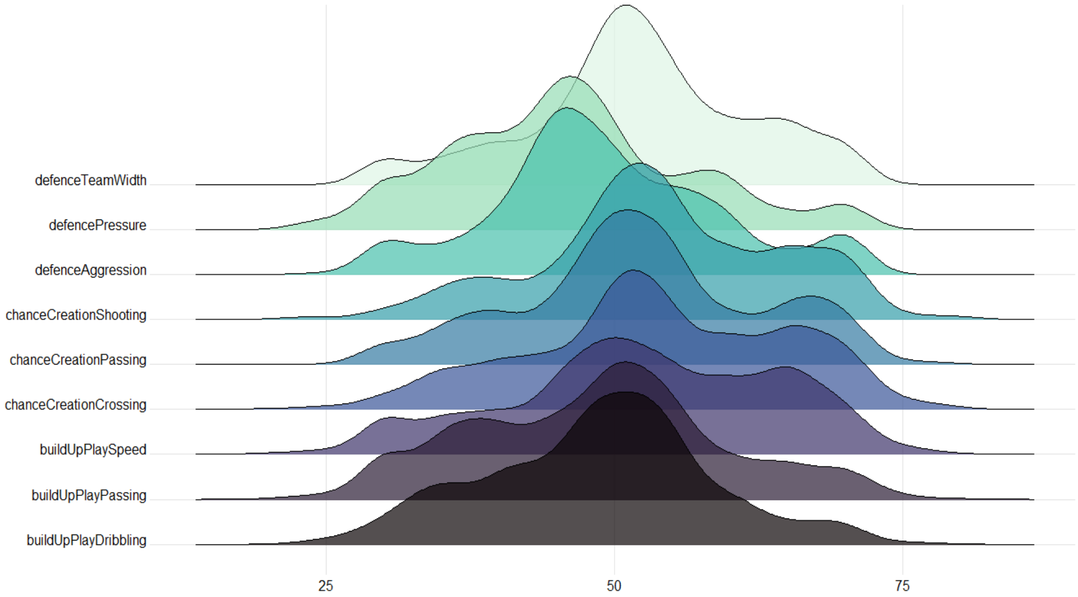


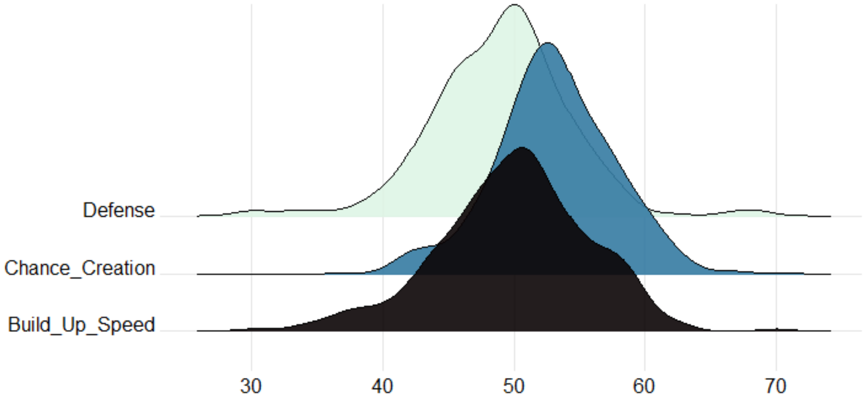
Figure (3-3) Distribution of team's quantitative attributes

It is clear that teams' variables are homogeneous and tend to normality since all means range between 48 and 54, while standard deviations between 9 and 12.

So, Teams' attributes could be classified into three groups, attributes of building up the play, attributes of chance creation, and attributes of defense; and averages of groups have been taken:

Table (3-4) Descriptive statistics of aggregated team's quantitative attributes

	Teams Average Attributes	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD	95% C.I for Mean
1	Build Up Play	30	46.36	50.22	49.97	53.64	70	5.78	49.29, 50.65
2	Chance Creation	41.28	50.39	52.94	53.02	56	66.67	4.39	52.51, 53.54
3	Defense	30	45.78	49.33	49	51.94	68.33	5.19	48.39, 49.60



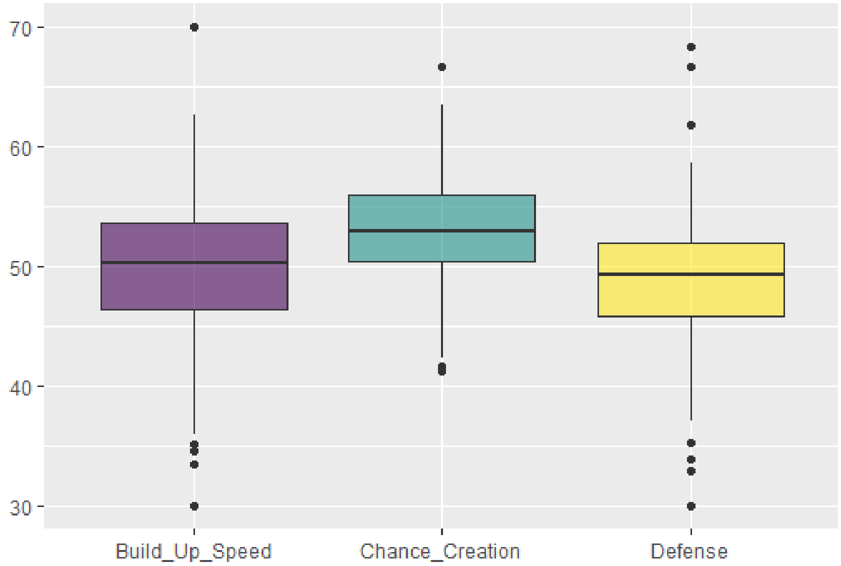


Figure (3-4) Distribution of aggregated team's quantitative attributes

From **Figure (3-4)**, the normality is apparent, and there is not much difference between groups, but still, chance creation attributes have the highest score on average and the minimal standard deviation since the defense attributes group has the most significant number of outliers.

3.1.1.2. General View on Players' Data:

As for teams, the database has two tables for players, one for general information about players, and the other about players' attributes in which three of them are qualitative and thirty-five are quantitative, and all of them are categorized according to the FIFA website to skills, mental, physical, and goal-keeper attributes. As shown in **appendix [1]**.

Table (3-5) Percentages of player's qualitative attributes

	Player's Qualitative Attributes	**Classes**				**Median**
1	Preferred foot	Right 75.23%	Left 24.31%		Missing 0.45%	Right
2	Attacking work rate	Low 4.658%	Medium 67.98%	High 23.28%	Missing 4.085%	Medium
3	Defensive work rate	Low 10.02%	Medium 71.12%	High 14.698%	Missing 4.16%	Medium

Table (3-6) Descriptive statistics of player's quantitative attributes

	Player's Quantitative Attributes	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD	95% C.I for Mean
1	Overall rating	33	64	69	68.6	73	94	7.041	68.573, 68.627
2	Potential	39	69	74	73.46	78	97	6.59	73.435, 73.486
3	Crossing	1	45	59	55.09	68	95	17.24	55.021, 55.153
4	Finishing	1	34	53	49.92	65	97	19.039	49.848, 49.994
5	Heading accuracy	1	49	60	57.27	68	98	16.489	57.203, 57.329
6	Short passing	3	57	65	62.43	72	97	14.19	62.375, 62.484
7	Volleys	1	35	52	49.47	64	93	18.257	49.398, 49.539
8	Dribbling	1	52	64	59.18	72	97	17.745	59.107, 59.243
9	Curve	2	41	56	52.97	67	94	18.256	52.895, 53.036
10	Free kick accuracy	1	36	50	49.38	63	97	17.83	49.312, 49.449
11	Long passing	3	49	59	57.07	67	97	14.39	57.015, 57.125
12	Ball control	5	58	67	63.39	73	97	15.197	63.330, 63.447
13	acceleration	10	61	69	67.66	77	97	12.98	67.609, 67.709
14	Sprint speed	12	62	69	68.05	77	97	12.57	68.003, 68.100
15	Agility	11	58	68	65.97	75	96	12.95	65.921, 66.021
16	Reactions	17	61	67	66.1	72	96	9.16	66.069, 66.139
17	Balance	12	58	67	65.19	74	96	13.06	65.139, 65.240
18	Shot power	2	54	65	61.81	73	97	16.14	61.746, 61.870
19	Jumping	14	60	68	66.97	74	96	11.007	66.927, 67.012
20	Stamina	10	61	69	67.04	76	96	13.17	66.988, 67.089
21	Strength	10	60	69	67.42	76	96	12.07	67.378, 67.471
22	Long shots	1	41	58	53.34	67	96	18.37	53.269, 53.410
23	Aggression	6	51	64	60.95	73	97	16.09	60.886, 61.010
24	Interceptions	1	34	57	52.01	68	96	19.45	51.935, 52.084
25	Positioning	2	45	60	55.79	69	96	18.45	55.716, 55.857
26	Vision	1	49	60	57.87	69	97	15.14	57.815, 57.932
27	Penalties	2	45	57	55	67	96	15.55	54.944, 55.064
28	Marking	1	25	50	46.77	66	96	21.23	46.691, 46.854
29	Standing tackle	1	29	56	50.35	69	95	21.487	50.269, 50.434
30	Sliding tackle	2	25	53	48	67	95	21.599	47.918, 48.085
31	Gk diving	1	7	10	14.7	13	94	16.865	14.640, 14.769
32	Gk handling	1	8	11	16.06	15	93	15.867	16.003, 16.125
33	Gk kicking	1	8	12	21	15	97	21.45	20.916, 21.081
34	Gk positioning	1	8	11	16.13	15	96	16.099	16.070, 16.194
35	Gk reflexes	1	8	11	16.44	15	96	17.198	16.375, 16.508

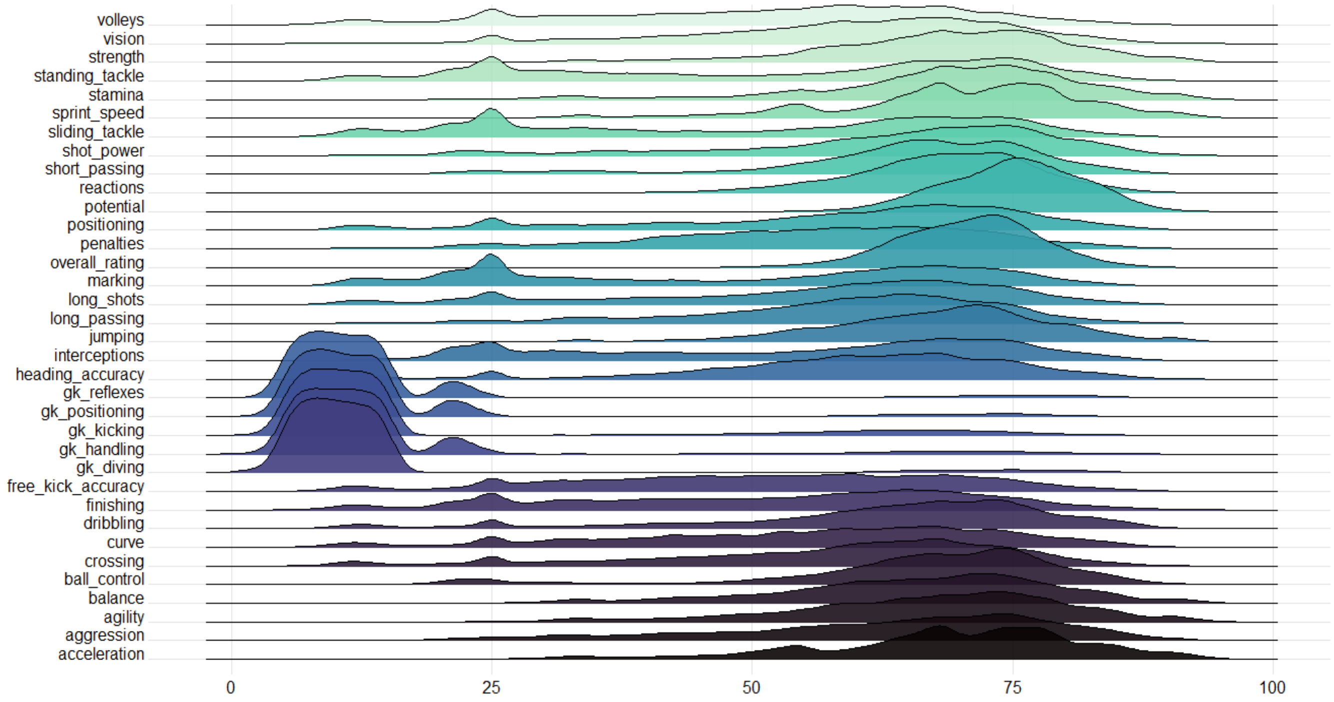


Figure (3-5) Distribution of player's quantitative attributes

Contrary to what was presented in teams' data, players' attributes variables are not homogeneous; there is a notable disparity between distributions of variables, some of them skewed to high scores with another smaller heap skewed to low scores, this forms a bio-mode shape. In addition, a tiny number of variables skewed to the left around low scores.

To reduce the disparity between players' attributes, averages have been calculated, since attributes grouped in respect to FIFA's classification as fifteen attributes under skills, five under mental, eight under physical, and five under goal-keeper attributes, baring in mind that two variables (**overall rating** and **potential**) are excluded from the averaging below:

Table (3-7) Descriptive statistics of aggregated player's quantitative attributes

	Players Average Attributes	**Min.**	**1st Qu.**	**Median**	**Mean**	**3rd Qu.**	**Max.**	**SD**	**95% C.I for Mean**
1	Skills	9.467	52.47	59.33	57.18	65.27	86.33	12.62	57.108, 57.25
2	Mental	7.6	50	58.2	56.66	65.6	90.25	12.58	56.59, 56.73
3	Physical	28.5	64.38	69.25	68.21	73.38	88.75	7.4	68.168, 68.25
4	GK_attributes	1	9.4	10.40	15.77	12	89.8	15.89	15.68, 15.86

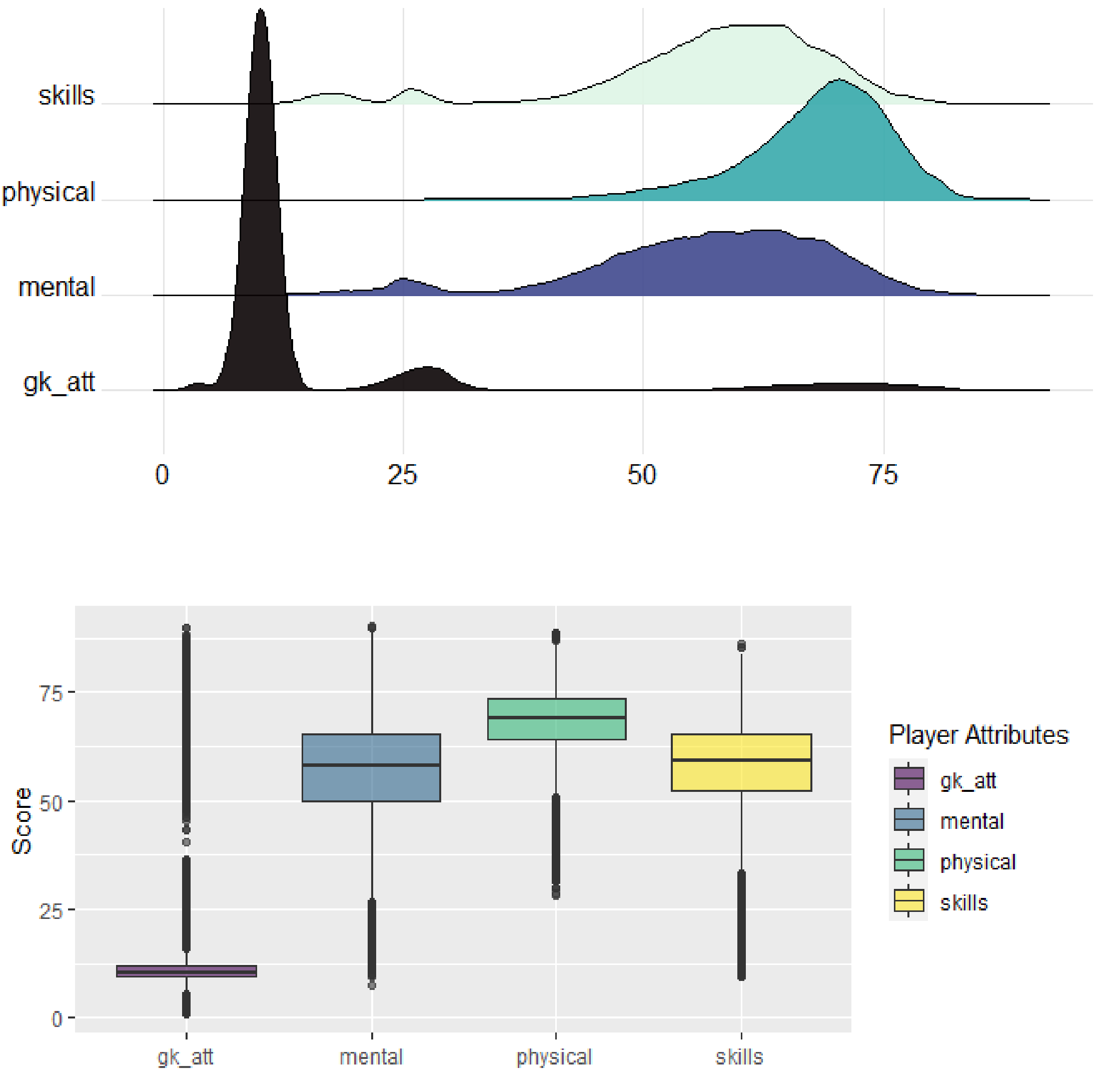


Figure (3-6) Distribution of aggregated player's quantitative attributes

From **Figure (3-6)**, it is clear that physical attributes have the highest scores with the slightest standard deviations, while goal-keeper attributes have the minor scores with the highest standard deviations; The reason for this is due to the tiny number of goal-keepers compared to other players, so there is a large number of players with a low score of goal-keeper attributes. In general, the discordant between players' positions caused the scuttering that appeared in the boxplot above.

3.1.1.3. General View on Matches' Data:

The table of matches contains information about home and away teams in every match, and how much goals they scored, in eleven leagues. Extra knowledge could be extracted from this table, for example, how many matches ended with draw:

Table (3-8) Matches results of each league

	League	Matches	Home Team Won		Draw		Away Team Won	
1	England Premier League	3040	1390	45.7%	783	25.76%	867	28.5%
2	France Ligue 1	3040	1359	44.7%	859	28.26%	822	27.04%
3	Spain LIGA BBVA	3040	1485	48.85%	704	23.16%	851	27.99%
4	Italy Serie A	3017	1407	46.64%	796	26.38%	814	26.98%
5	Germany 1. Bundesliga	2448	1107	45.22%	597	24.39%	744	30.39%
6	Netherlands Eredivisie	2448	1171	47.84%	581	23.7%	696	28.4%
7	Portugal Liga ZON Sagres	2052	908	44.25%	533	25.97%	611	29.78%
8	Poland Ekstraklasa	1920	870	45.3%	525	27.3%	525	27.3%
9	Scotland Premier League	1824	760	41.67%	447	24.5%	617	33.8%
10	Belgium Jupiler League	1728	810	46.88%	425	24.6%	493	28.5%
11	Switzerland Super League	1422	650	45.7%	346	24.33%	426	29.96%

The previous table shows that most of matches ended in favor of the home team, while the rest of the matches' results balanced between the draw and won for the away team.

Also, we could get some knowledge about goals and goals ratio:

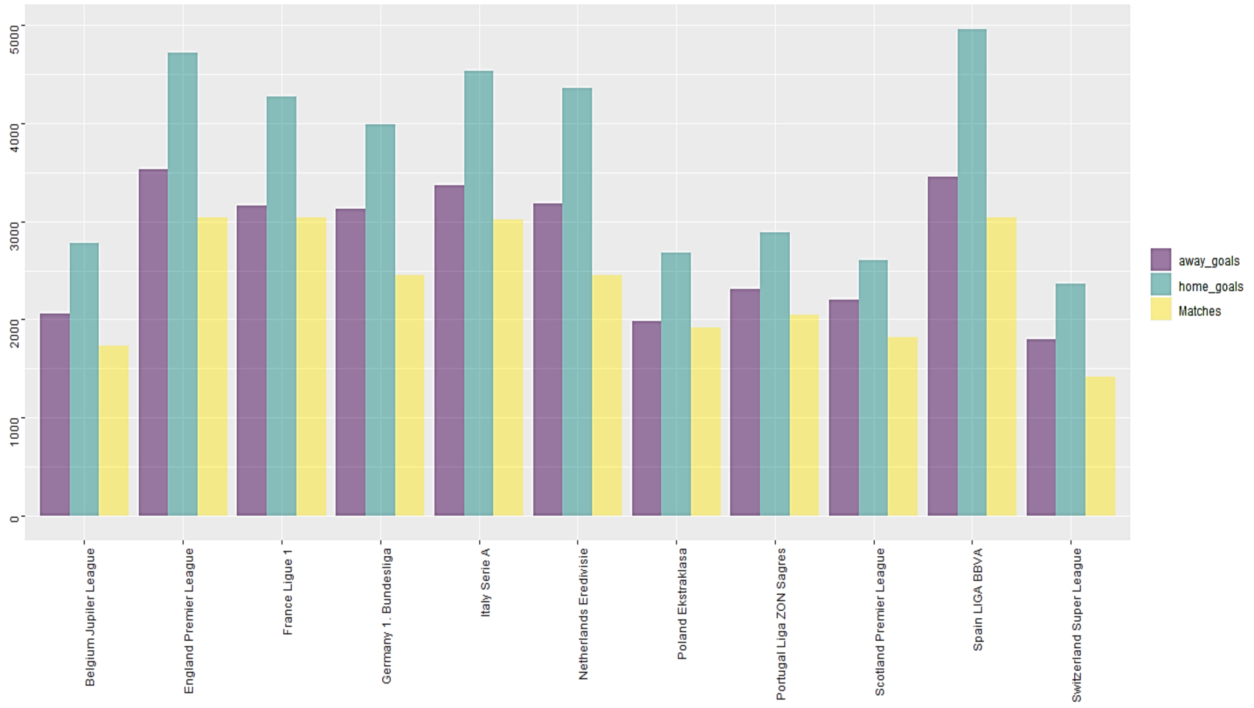


Figure (3-7) Goals results of each league

Figure (3-7) shows that, in general, teams scores on their ground more than the opponent team's ground. Also, it shows that the **Spain LIGA BBVA** has the highest number of goals, then the **England Premier League,** while the **Switzerland Super League** has the lowest goals number. However, because of inequality in the number of matches for every league, the goals ratio has been calculated:

$$Goals\ Ratio = \frac{Total\ Goals}{Number\ of\ Matches}$$

Table (3-9) Goals results of each league

	League	Home Goals	Away Goals	Total Goals	Goals Ratio
1	Netherlands Eredivisie	4357	3185	7542	3.08
2	Switzerland Super League	2365	1801	4166	2.93
3	Germany 1. Bundesliga	3982	3121	7103	2.9
4	Belgium Jupiler League	2781	2060	4841	2.8
5	Spain LIGA BBVA	4959	3453	8412	2.77
6	England Premier League	4715	3525	8240	2.71
7	Scotland Premier League	2607	2197	4804	2.63
8	Italy Serie A	4528	3367	7895	2.62
9	Portugal Liga ZON Sagres	2890	2311	5201	2.53
10	France Ligue 1	4265	3162	7427	2.44
11	Poland Ekstraklasa	2678	1978	4656	2.43

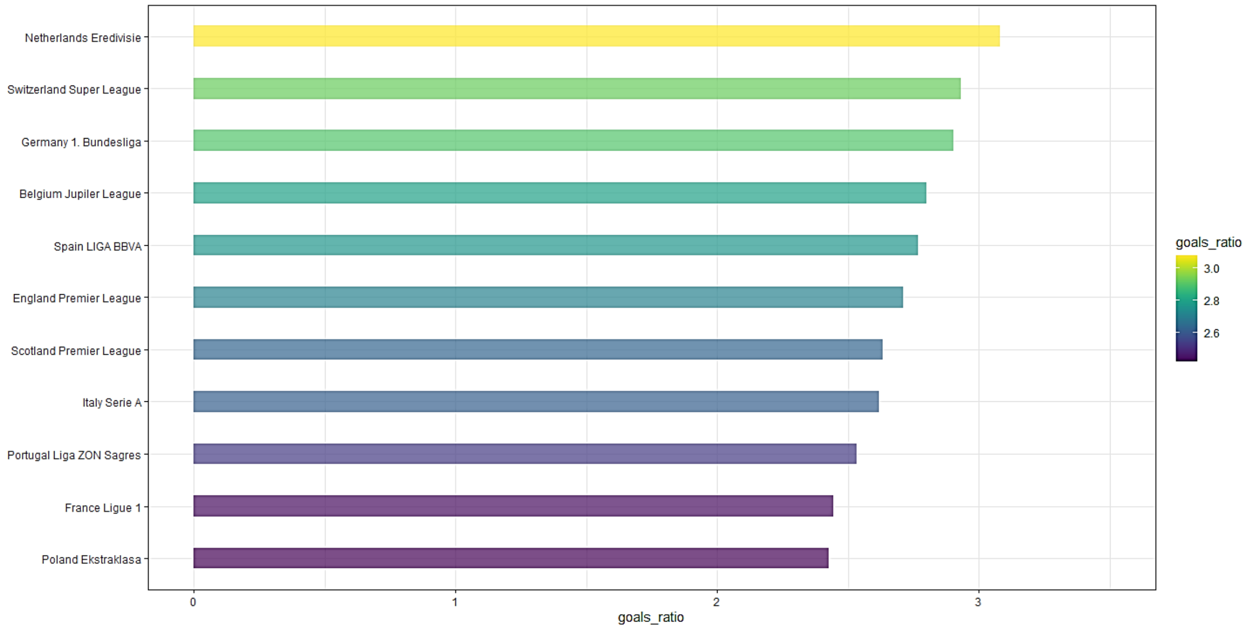


Figure (3-8) Goals ratios of each league

Unexpectedly, **Netherlands Eredivisie** and **Switzerland Super League** have the highest goals ratio among leagues, followed by **Germany 1. Bundesliga** as the third, with nearly three goals in every match.

To have a close look at matches results; text-mining tools have been used on winners' name variable to figure the team with the highest number of wins:

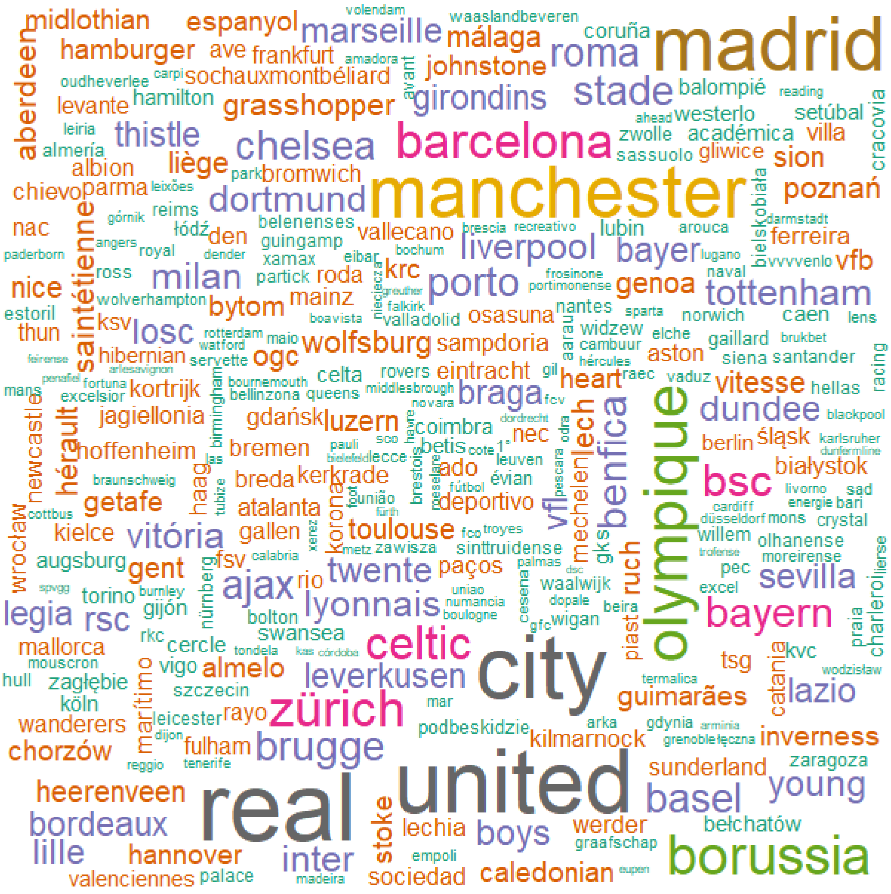


Figure (3-9) Top winner teams

From the **Figure (3-9)**, the words: **Manchester**, **real**, **united**, **city**, and **Madrid**, are the most common words in the winners' data but some words repeated with different teams' names (i.e., **Manchester** for **Manchester United** and **Manchester City**, **Madrid** for **Real Madrid** and **Atlético Madrid**, etc.); bearing in mind there is only one **Barcelona**! So, for more specification, some calculations are essential.

Table (3-10) Results of top winner teams

	Team	Wins	Total Goals	Goals Diff-Means	Home Wins	Home Goals	Away Wins	Away Goals
1	FC Barcelona	234	779	2.77	131	473	103	306
2	Real Madrid CF	228	776	2.64	129	479	99	297
3	Celtic	218	613	2.39	120	365	98	248
4	FC Bayern Munich	193	591	2.57	109	359	84	232

5	Manchester United	192	490	2.02	116	310	76	180
6	Juventus	189	443	1.91	105	253	84	190
7	SL Benfica	185	509	2.26	102	296	83	213
8	FC Porto	183	494	2.25	102	276	81	218
9	Ajax	181	550	2.58	103	333	78	217
10	FC Basel	180	506	2.12	103	305	77	201
11	PSV	178	559	2.44	105	335	73	224
12	Chelsea	176	475	2.18	101	276	75	199
13	Manchester City	175	498	2.27	113	328	62	170
14	Paris Saint-Germain	175	467	2.15	102	291	73	176
15	Arsenal	170	443	2.08	97	262	73	181
16	Atlético Madrid	167	423	2.05	103	282	64	141
17	Roma	162	384	1.77	97	247	65	137
18	Borussia Dortmund	157	447	2.25	88	255	69	192
19	Inter	154	371	1.80	90	225	64	146
20	Milan	154	368	1.86	92	221	62	147

The previous table (3-10), shows information about the winning matches of every team, since Barcelona has the highest number of wins, highest number of total goals. Also, it wins with the highest rate of goals difference, in addition, **Barcelona** has the highest goals ratio in general as shown in the following table which gives information about total matches:

Table (3-11) Goals ratio of top winner teams

	Team	Away Goals	Away Goals Ratio	Home Goals	Home Goals Ratio	Total Goals	Total Matches	Total Goals Ratio
1	FC Barcelona	354	2.33	495	3.26	849	304	2.79
2	Real Madrid CF	338	2.22	505	3.32	843	304	2.77
3	FC Bayern Munich	271	1.99	382	2.81	653	272	2.40
4	PSV	282	2.07	370	2.72	652	272	2.40
5	Ajax	287	2.11	360	2.65	647	272	2.38
6	SL Benfica	247	1.99	321	2.59	568	248	2.29
7	Celtic	306	2.01	389	2.56	695	304	2.29
8	FC Porto	246	1.98	295	2.38	541	248	2.18
9	FC Basel	275	1.92	344	2.41	619	286	2.16
10	Rangers	147	1.93	177	2.33	324	152	2.13
11	Borussia Dortmund	253	1.86	298	2.19	551	272	2.03
12	RSC Anderlecht	180	1.70	247	2.33	427	212	2.01
13	Manchester City	241	1.59	365	2.40	606	304	1.99
14	Club Brugge KV	186	1.75	235	2.22	421	212	1.985

15	Chelsea	250	1.64	333	2.19	583	304	1.92
16	Manchester United	244	1.61	338	2.22	582	304	1.91
17	Arsenal	267	1.76	306	2.01	573	304	1.88
18	FC Twente	220	1.62	289	2.12	509	272	1.87
19	Paris Saint-Germain	236	1.55	332	2.18	568	304	1.87
20	BSC Young Boys	210	1.47	319	2.23	529	286	1.85

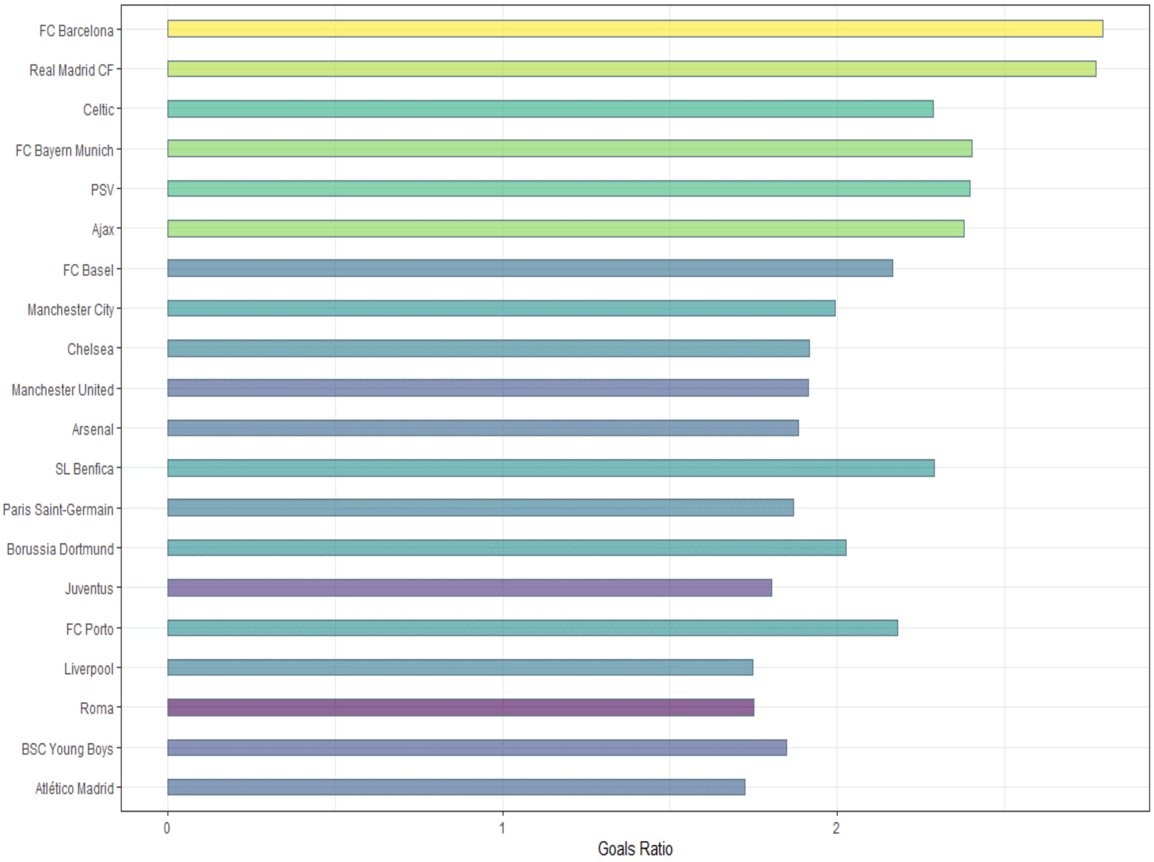


Figure (3-10) Goals ratio of top winner teams

The results show a convergence between **Barcelona** and **Real Madrid**, but **Barcelona** excels in away matches.

The following **Figure (3-11)** shows the top thirty teams in goals ratio and wins' goals difference since the length of the bar presents the goals ratio while the color represents goals difference:

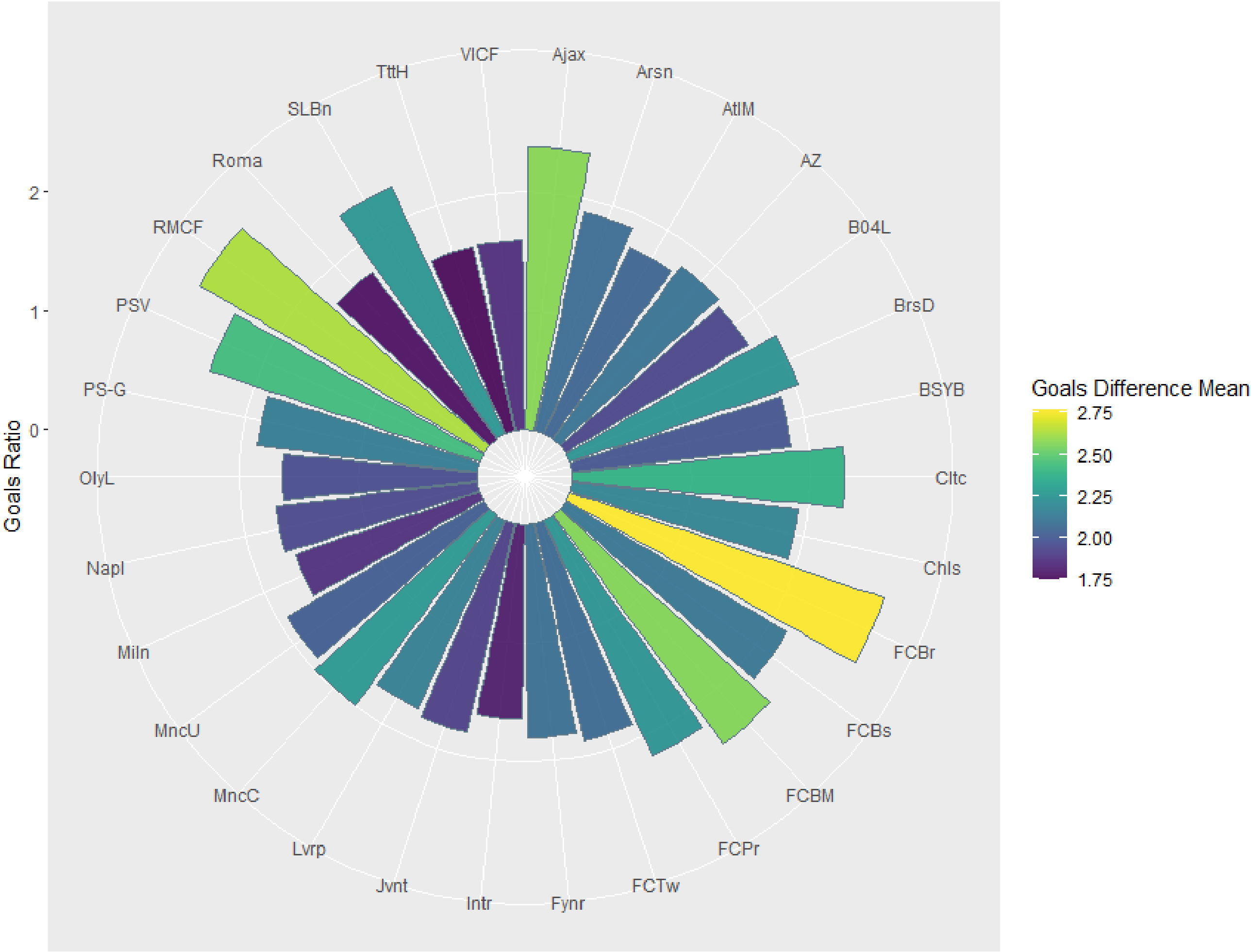


Figure (3-11) Goals difference mean of top winner teams

3.1.2. Possible Comparisons between Levels of Input Data:

Suppose one wants to look at data as a material for a predictive model. In that case, the first thing to think about is which variables represent the input data and which represent the output data. In **European Soccer Data**, teams' and players' attributes and leagues could be input data for the match, and the game results could be output data.

After determining input and output data, one may think about if the input variables have the same effect or not. In statistics, usually, Analysis of Variance is the most convenient tool to use. Still, as it is known, ANOVA depends on the number of factors, levels, and the nature of the experiment in general. So, we will study some possible factors and comparisons in a straightforward way.

3.1.2.1. Teams' Attributes Effect Among Different Leagues:

From the teams' data, the attributes have categorized to three main features in which the mean and standard deviation were calculated with respect to their leagues as follow.

Table (3-12) Mean and standard deviation of team's attributes of the leagues

	League's Name	Build Up Play		Chance Creation		Defense	
		Mean	SD	Mean	SD	Mean	SD
1	Belgium Jupiler League	51.55	4.51	50.61	4.36	51.13	6.6
2	England Premier League	53.44	4.38	54.21	3.57	49.21	4.31
3	France Ligue 1	51.13	3.599	52.6	4.37	49.07	4.1
4	Germany 1. Bundesliga	52.61	4.18	53.95	4.22	49.87	4.19
5	Italy Serie A	49.78	5.065	53.24	4.24	47.6	3.09
6	Netherlands Eredivisie	44.47	4.79	50.70	3.39	45.55	6.3
7	Poland Ekstraklasa	47.86	6.84	52.12	3.4	52.14	6.19
8	Portugal Liga ZON Sagres	46.34	5.74	52.81	4.998	44.5	5.898
9	Scotland Premier League	55.43	6.02	55.07	5.3	51.38	5.59
10	Spain LIGA BBVA	46.88	5.95	54.04	4.7	50.7	3.53
11	Switzerland Super League	50.23	3.12	53.29	4.61	47.88	2.47

The above results show that the **chance creation** has less variability among the leagues, while the **Germany 1. Bundesliga** has the lowest variability with respect to the attributes.

To zoom on the teams' attributes for all the leagues, each of them branched to three components where the mean and the standard deviation has been calculated in the following tables to distinguish the variability among them.

Table (3-13) Mean and standard deviation of build up play team's attributes of the leagues

	League's Name	Build up play speed		Build up play dribbling		Build up play passing	
		Mean	SD	Mean	SD	Mean	SD
1	Belgium Jupiler League	53.39	9.08	47.15	4.04	49.95	8.23
2	England Premier League	56.2	10.6	37.59	7.13	54.36	12.25
3	France Ligue 1	53.2	9.37	53.76	8.73	48.46	8.99
4	Germany 1. Bundesliga	56.4	10.68	50.12	9.36	48.9	9.79
5	Italy Serie A	54.6	12.3	53.04	12.26	45.05	11.48
6	Netherlands Eredivisie	45.96	10.64	41.06	5.95	44.7	9.99
7	Poland Ekstraklasa	50.08	14.3	51.23	6.49	48.29	11.19
8	Portugal Liga ZON Sagres	48.28	12.3	53.59	8.46	43.8	9.1

9	Scotland Premier League	56.5	10.3	50.81	4.62	54.77	10.09
10	Spain LIGA BBVA	47.37	11.17	48.76	5.91	45.84	10.99
11	Switzerland Super League	51.48	9.07	52.9	7.85	50.15	8.14

Table (3-14) Mean and standard deviation of chance creation team's attributes of the leagues

	League's Name	Chance creation passing		Chance creation crossing		Chance creation shooting	
		Mean	SD	Mean	SD	Mean	SD
1	Belgium Jupiler League	50.3	9.38	53.35	9.38	50.04	8.68
2	England Premier League	53.04	11.75	57.09	11.12	52.5	10.44
3	France Ligue 1	50.41	8.47	53.92	9.77	52.79	10.74
4	Germany 1. Bundesliga	54.47	10.36	52.84	11.59	54.35	10.43
5	Italy Serie A	51.55	12.54	51.75	12.98	57.04	10.31
6	Netherlands Eredivisie	49.09	10.25	51.74	10.16	51.85	10.36
7	Poland Ekstraklasa	51.71	10.35	49.34	12.73	57.56	11.3
8	Portugal Liga ZON Sagres	54.24	6.58	55.46	8.48	53.27	7.74
9	Scotland Premier League	52.69	9.42	56.01	9.47	55.52	10.6
10	Spain LIGA BBVA	53.17	10.43	53.72	10.84	54.94	10.26
11	Switzerland Super League	52.32	9.13	55.58	11.29	53.27	8.4

Table (3-15) Mean and standard deviation of defense team's attributes of the leagues

	League's Name	Defense pressure		Defense aggression		Defense team width	
		Mean	SD	Mean	SD	Mean	SD
1	Belgium Jupiler League	49.14	10.05	51.29	8.18	54.3	9.68
2	England Premier League	45.79	10.58	50.24	10.64	51.6	9.2
3	France Ligue 1	45.68	10.41	48.78	8.5	52.14	9.94
4	Germany 1. Bundesliga	48.49	10.91	51.1	9.21	51.12	9.16
5	Italy Serie A	42.1	9.14	50.18	9.76	50.47	8.67
6	Netherlands Eredivisie	43.82	10.5	46.65	10.85	47.76	10.5
7	Poland Ekstraklasa	49.6	11.6	50.8	11.6	54.03	9.12
8	Portugal Liga ZON Sagres	41.520	9.84	44.86	9.94	49.97	11.29
9	Scotland Premier League	47.49	7.6	50.35	9.88	55.01	9.75
10	Spain LIGA BBVA	47.83	9.23	48.57	8.68	55.85	8.79
11	Switzerland Super League	45.29	5.7	46.34	7.47	52.48	4.28

It is clear that the range of dispersion for the **passing** in the **build-up play** attribute is the smallest as well as the **crossing** in the **chance creation** attribute as well as the **aggression** in the **defense** attribute. But, from the leagues point of view, the France and Italian leagues have the lowest dispersion in the **build-up play** attribute, while the Dutch league has the smallest variability in both chance creation and **defense**.

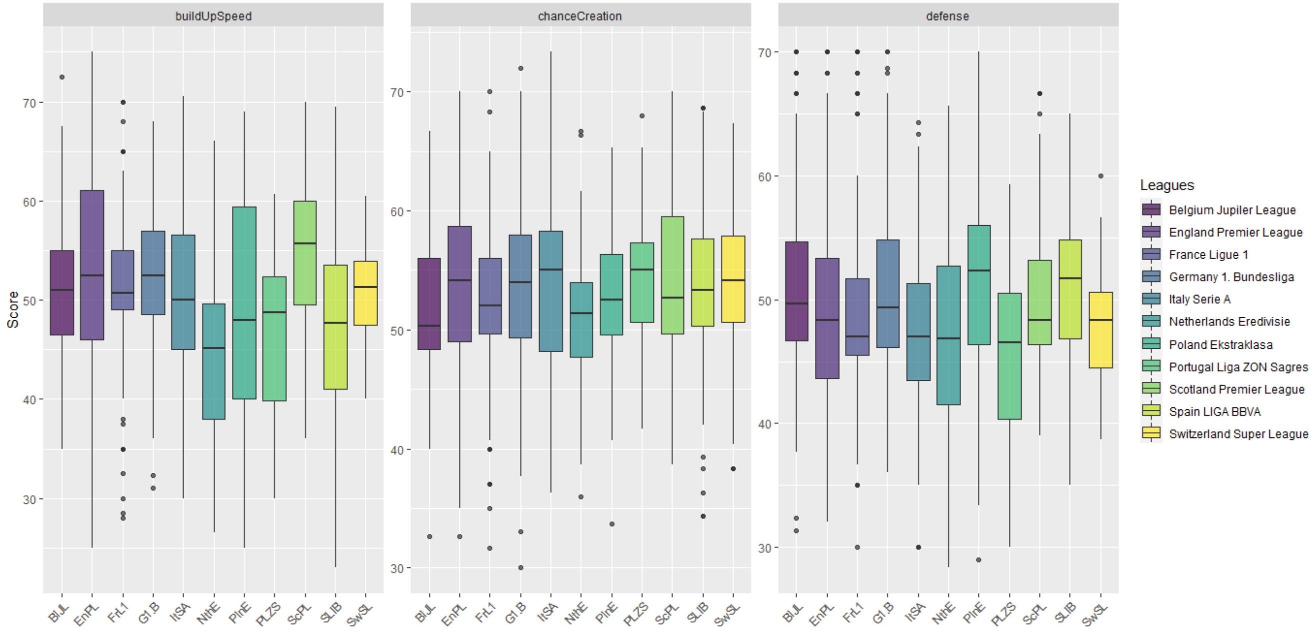


Figure (3-12) Boxplot of team's attributes for the leagues per each attribute

The **Figure (3-12)** shows from left to right, existence of variability within each attribute for all the leagues, while the middle graph of the **chance creation** has the lowest variability among the rest of attributes.

From different approach, the following **Figure (3-13)** confirms variability between the three attributes for all the leagues except the **Belgium Jupiler League** which has the smallest variability among all the leagues.

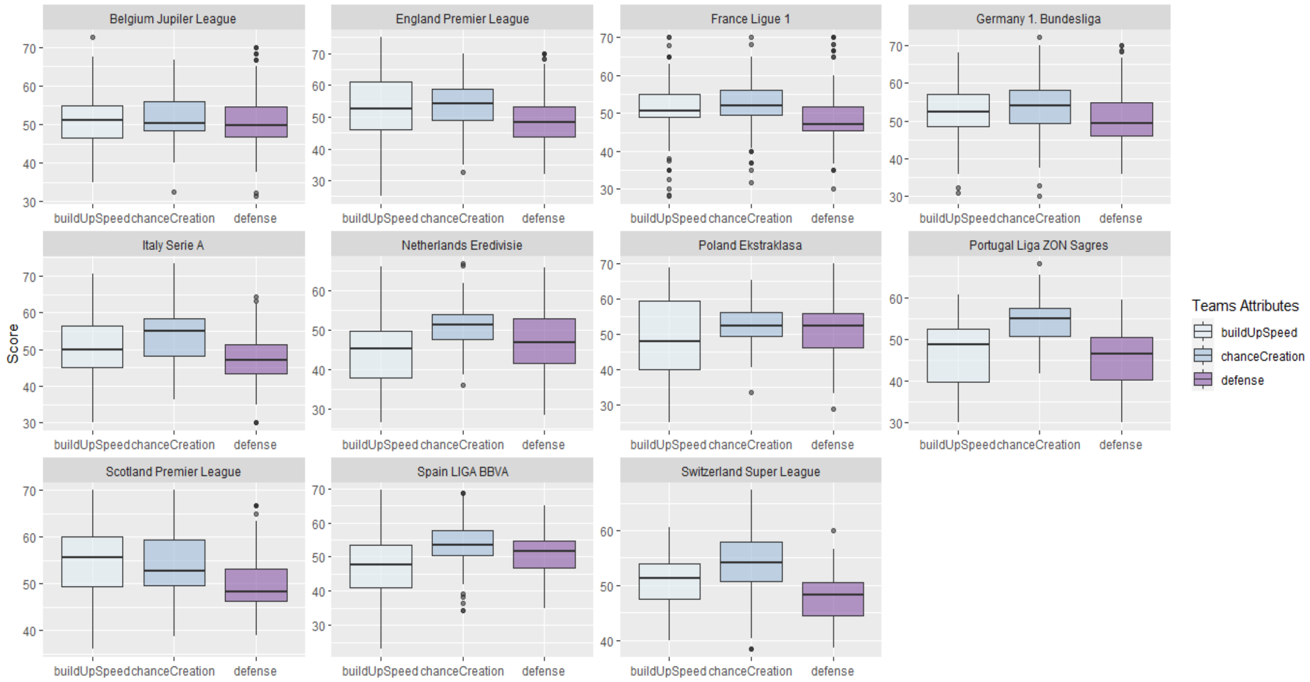


Figure (3-13) Boxplot of team's attributes per each league

All the previous tables and figures proves expectance of variabilities either between teams' attributes, or between the leagues, which will give rise to some questions as:

1- Do the attributes have different effects within the leagues?
2- Do the leagues have different effects within the attributes?
3- Do the leagues and the attributes have interaction effects between them?

In this case, a design of factorial experiment with interaction can be suggested, were the different teams representing the experimental units and the randomization has to be applied to distribute random sample of them all over the factor levels.

Practically, a random sample of twelve teams have been selected from every league without replacement, then four teams randomly allocated to every level of attribute factor by using a Randomization Algorithm to form the following table.

Table (3-16) The random experimental units for each league per every attribute

League	Build Up Play	Chance Creation	Defense
Belgium Jupiler League	KAA Gent KV Kortrijk RAEC Mons Lierse SK	Club Brugge KV Royal Excel Mouscron KVC Westerlo Sporting Charleroi	Standard de Liège Beerschot AC Oud-Heverlee Leuven KSV Roeselare
England Premier League	Cardiff City Swansea City Norwich City Blackpool	Tottenham Hotspur Wolverhampton Wanderers Newcastle United Hull City	Wigan Athletic Southampton Bournemouth Everton
France Ligue 1	FC Sochaux-Montbéliard Évian Thonon Gaillard FC AC Arles-Avignon AJ Auxerre	Olympique de Marseille Toulouse FC FC Lorient SM Caen	Valenciennes FC Le Mans FC AS Monaco RC Lens
Germany 1. Bundesliga	Fortuna Düsseldorf FC Augsburg Eintracht Braunschweig FC Bayern Munich	Eintracht Frankfurt VfB Stuttgart FC Schalke 04 Hannover 96	Hamburger SV 1. FC Nürnberg SV Darmstadt 98 SpVgg Greuther Fürth
Italy Serie A	Torino Pescara Milan Lazio	Carpi Parma Frosinone Reggio Calabria	Chievo Verona Hellas Verona Sampdoria Cagliari
Netherlands Eredivisie	Go Ahead Eagles Feyenoord Ajax FC Dordrecht	VVV-Venlo AZ NAC Breda N.E.C.	Vitesse FC Twente SC Cambuur Excelsior
Poland Ekstraklasa	Zagłębie Lubin Piast Gliwice Jagiellonia Białystok Arka Gdynia	Lechia Gdańsk Ruch Chorzów Górnik Łęczna Odra Wodzisław	GKS Bełchatów Lech Poznań Śląsk Wrocław Polonia Bytom
Portugal Liga ZON Sagres	Gil Vicente FC CF Os Belenenses FC Porto S.C. Olhanense	Boavista FC Moreirense FC Leixões SC Vitória Guimarães	FC Arouca Estoril Praia CS Marítimo SL Benfica

Scotland Premier League	Ross County FC Dundee FC St. Mirren Rangers	Falkirk Inverness Caledonian Thistle Heart of Midlothian Motherwell	Kilmarnock Celtic Hibernian Partick Thistle F.C.
Spain LIGA BBVA	Real Sporting de Gijón Córdoba CF CA Osasuna Real Madrid CF	Elche CF RC Celta de Vigo Rayo Vallecano RCD Mallorca	FC Barcelona Real Sociedad Real Zaragoza RC Recreativo
Switzerland Super League	FC Luzern FC Vaduz FC Zürich Neuchâtel Xamax	FC Sion BSC Young Boys FC Aarau FC Thun	FC Lausanne-Sports FC St. Gallen Servette FC AC Bellinzona

By selecting the observed unit corresponding to every team in the previous table randomly, the resulting table will look like the following:

Table (3-17) The random measurable units for each league per every attribute

League	Build Up Play	Chance Creation	Defense
Belgium Jupiler League	48.500 40.000 56.667 48.000	43.000 46.667 62.000 42.500	30.000 60.333 56.667 57.667
England Premier League	46.667 50.000 59.333 55.000	56.500 44.000 70.000 50.000	30.000 40.667 46.333 52.333
France Ligue 1	50.333 51.667 54.333 54.333	50.000 57.000 56.667 43.333	48.500 37.667 46.667 52.500
Germany 1. Bundesliga	45.667 47.667 56.667 54.000	45.000 36.667 50.000 68.333	60.333 54.000 55.000 57.667

Italy Serie A	55.000 46.333 47.000 50.000	55.000 55.000 43.667 68.333	55.000 46.667 45.500 46.667
Netherlands Eredivisie	63.500 40.000 46.667 42.500	46.333 45.000 56.333 54.000	55.000 45.500 47.333 58.667
Poland Ekstraklasa	58.333 52.333 43.000 48.667	56.667 58.333 46.333 51.667	51.000 47.000 51.667 53.333
Portugal Liga ZON Sagres	45.000 48.500 47.667 47.333	51.000 37.667 51.000 51.333	48.500 48.667 50.000 47.000
Scotland Premier League	51.667 46.333 48.000 70.000	56.667 45.500 37.500 47.000	56.667 46.667 57.500 40.000
Spain LIGA BBVA	51.667 52.000 62.333 59.667	51.667 49.333 46.000 33.333	55.000 46.667 53.000 43.667
Switzerland Super League	51.667 57.000 33.333 55.000	44.000 46.000 50.667 66.667	45.000 54.667 46.667 52.500

The mathematical model for a completely randomized two-factor factorial design can be written as:

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \epsilon_{ijk} \ ; \quad \forall\, i = 1, \dots, 11 \, ; j = 1, 2, 3 \, ; k = 1, \dots, 4$$

Where: y_{ijk} represents every observation, μ represents the general mean, α_i represents the effect of the ith level of first factor which is Leagues, in other words it represents the difference between the ith League's attributes mean and the general data mean, β_j represents the effect of the jth level of second factor which is teams' attributes, in other words it represents the difference between the jth Attribute's mean and the general data mean, $(\alpha\beta)_{ij}$ represents the effect of interaction between ith League and jth Attribute, in other words it represents the difference between the cell mean and general mean, and finally, ϵ_{ijk} the random error represents the difference between every observation and its cell mean (Lawson, 2014).

Where model's errors must guarantee the usual assumptions of normality, $\epsilon_{ijk}\sim N(0,\sigma^2)$, and independence. The independence assumption is guaranteed as the treatment combinations are randomly assigned to the experimental units, and the equal variance and normality assumptions may be verified with a residual versus predicted plot and a normal probability plot of the residuals as described in the following figure:

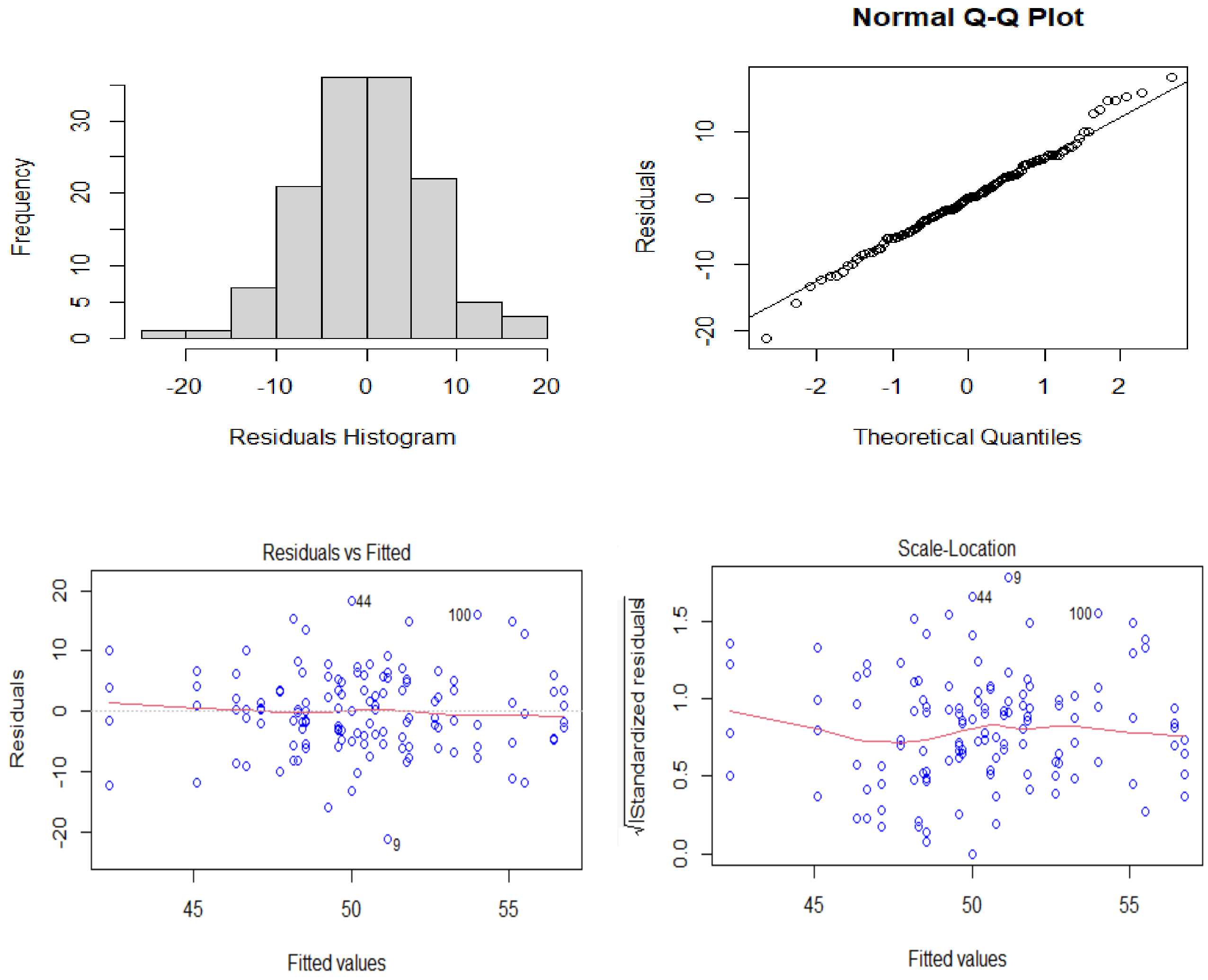


Figure (3-14) Graphs of satisfying the design model assumptions

The statistical hypotheses of this design can be stated as the following:

1- $H_o: \alpha_i = 0 \quad vs \quad H_1: \alpha_i \neq 0$
2- $H_o: \beta_j = 0 \quad vs \quad H_1: \beta_j \neq 0$
3- $H_o: (\alpha\beta)_{ij} = 0 \quad vs \quad H_1: (\alpha\beta)_{ij} \neq 0$

By applying the suitable analysis of variance for **Table (3-17)**, the result has showed in the following table:

Table (3-18) Results of the ANOVA table

Factors	D.F	Sum Sq.	Mean Sq.	F value	Pr. (>F)
Leagues	10	177	17.70	0.301	0.979
Team's Attributes	2	40	20.08	0.341	0.712
Team's Attributes* Leagues	20	1095	54.74	0.930	0.552
Residuals	99	5828	58.87		

Note that all the probability values for the leagues, team's attributes, and the interactions between them are greater than the level of significance, so the test is not significance and all the leagues and all the team's attributes have the same effect with no interactions between them.

3.1.2.2. Players' Attributes Effect Among Different Leagues:

As mentioned before about players' data, the FIFA have categorized attributes to four main features in which the mean and standard deviation were calculated with respect to their leagues as follow.

Table (3-19) Mean and standard deviation of player's attributes for leagues

	League's Name	**Skills**		**Physical**		**Mental**		**GK_Attributes**	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD
1	Belgium Jupiler League	54.41	11.08	66.41	6.89	53.49	11.68	15.65	15.07
2	England Premier League	60.44	13.12	70.55	6.96	60.14	12.45	16.54	17.02
3	France Ligue 1	56.98	12.51	67.97	7.51	56.97	12.80	16.14	16.17
4	Germany 1. Bundesliga	59.34	12.43	69.64	7.05	58.51	12.53	16.11	16.44
5	Italy Serie A	60.53	12.57	69.49	7.26	59.75	12.44	14.85	15.83
6	Netherlands Eredivisie	55.61	11.10	67.24	6.41	54.50	11.92	15.94	15.56
7	Poland Ekstraklasa	50.30	11.12	64.15	8.24	50.52	11.43	15.65	15.07
8	Portugal Liga ZON Sagres	56.90	12.05	67.97	7.13	56.15	11.97	16.54	17.02
9	Scotland Premier League	50.01	11.15	65.29	7.39	50.41	10.74	16.14	16.17
10	Spain LIGA BBVA	59.23	12.67	69.55	6.86	58.63	12.62	16.11	16.44
11	Switzerland Super League	51.90	11.14	64.77	7.52	51.50	10.92	14.85	15.83

The above results show that the **Physical Attributes** has less variability among the leagues, although the high dispersion, still the **Poland Ekstraklasa** which has the lowest variability with respect to attributes.

Because of the large number of attributes that branched from the four main categories, it is hard to view all of them by details for every league; so, some boxplot figures would be sufficient.

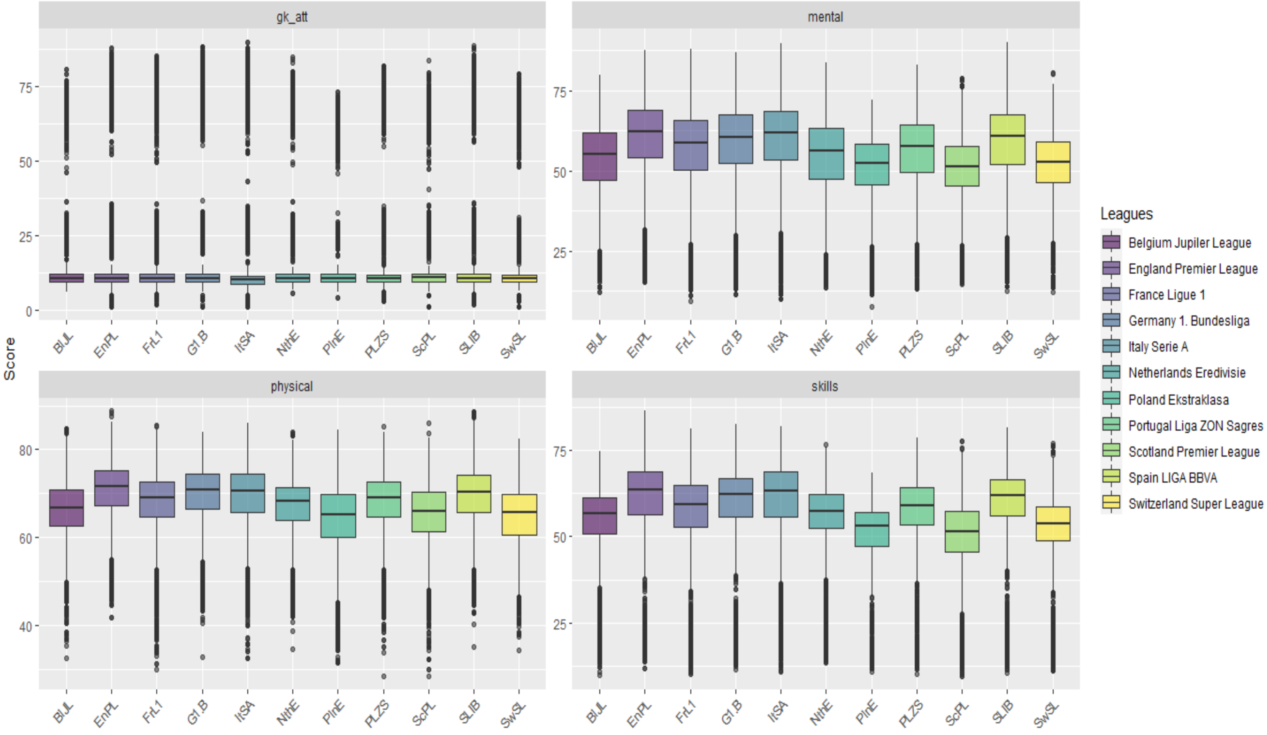


Figure (3-15) Boxplot of player's attributes for leagues per attribute

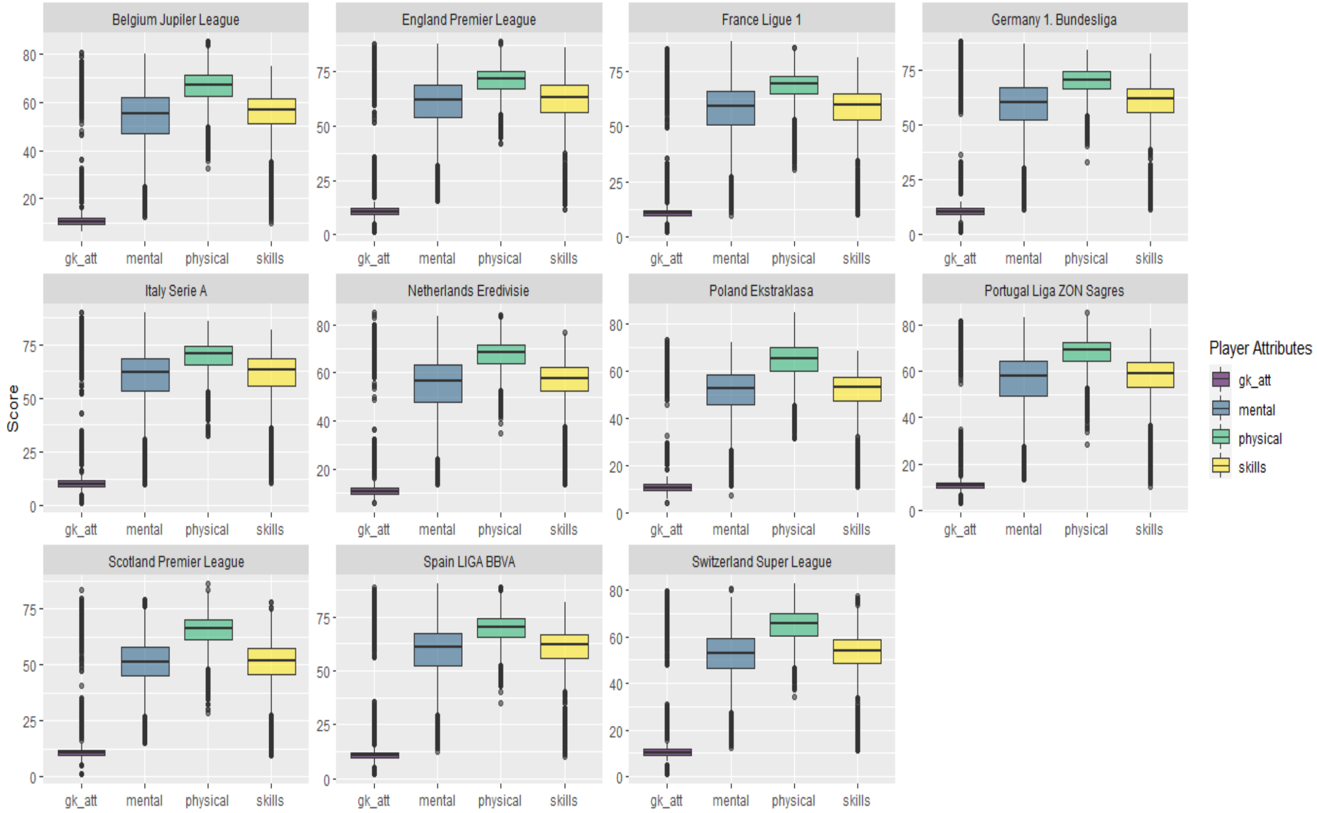


Figure (3-16) Boxplot of player's attributes per league

The last two figures show huge variability and outliers within the attributes for every league, and within the leagues for every attribute, which may due to different positions of the players that the Kaggle database does not provide any detail information about it. The large variability and the outliers between the observations require a plane to control the random error, which may appear in the model in order to reduce its effect.

In this case blocking introduced to control the variability and reducing the error, despite no idea about which factor could be used to represent the blocks, that will lead to use one of the data mining methods, the so-called **Unsupervised Classification**, specifically, the **Clustering Analysis**.

3.1.2.2.1. Clustering Definition:

Many applications require the partitioning of data points into intuitively similar groups. The partitioning of a large number of data points into a smaller number of groups helps greatly in summarizing the data and understanding it for a variety of data mining applications (Tan, Steinbach & Kumar, 2006). An informal and intuitive definition of clustering is as follows:

*"*Given a set of data points, partition them into groups containing very similar data points.*"*

This is a very rough and intuitive definition because it does not state much about the different ways in which the problem can be formulated, such as the number of groups, or the objective criteria for similarity. Another definition can be as follows: Cluster analysis is a process used to form groups or clusters, so that observations within a cluster have high similarity, but are very dissimilar to observations in other clusters. Dissimilarities and similarities are assessed based on the observations and often involve distance measures (Han, Pei, & Kamber, 2011). The key idea is to characterize the clusters in ways that would be useful for the aims of the analysis. Clustering analysis has a vast benefits and applications in many practical fields such as:

- **Marketing**: It helps marketers find out distinctive groups among their customers bases, and this knowledge helps them improve their targeted marketing programs.
- **City-planning:** It also helps in identifying clusters of houses based on house type, geographical location, and value.
- **Biology studies:** Clustering helps in defining plant and animal classifications, identifying gens with similar functionalities, and in gaining insights into structures inherent to populations (Aggarwal, 2015).

Since clustering is popular in many fields, there exist a great number of clustering methods, such as: **K-means**, **Mean-Shift Clustering**, **Hierarchical Clustering**, **Grid-Based Clustering**, and many other methods, which would be differ with respect to partitioning levels and similarity measurements they use.

Clustering is so challenging field, with too many requirements and aspects that help in comparing between clustering methods and deciding which technique is more convenient to use. While this thesis is not purpose to studying clustering analysis in details, so we used the simplest and most common clustering method the so-called **K-means Clustering** (Han, Pei, & Kamber, 2011).

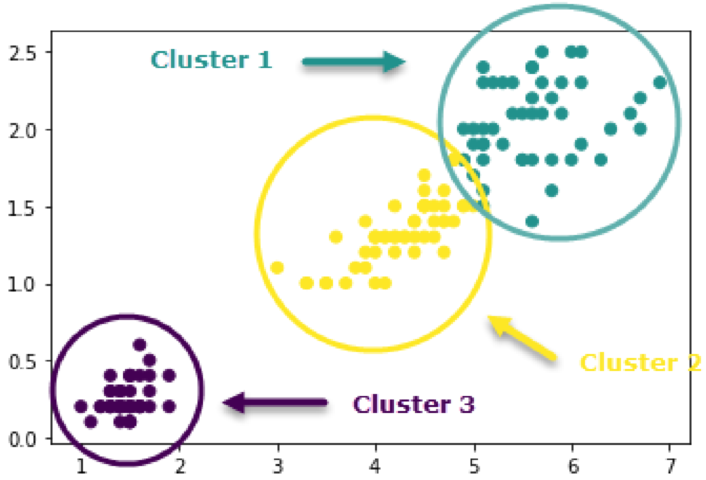


Figure (3-17) K-mean clustering

K-means clustering is a simple and elegant approach for partitioning data set into K distinct, non-overlapping clusters. To perform K-means clustering, we must first specify the desired number of clusters K which has no way to be determine except the researcher intuition; then the K-means algorithm will assign each observation to exactly one of the K clusters. This process could be explained in the following steps:

Step 1: Start with a selection of the value of K:

In this step, the K observations (centroids) c_i are selected randomly, where each one of them represents an initial centroid of its cluster, then the rest of the remaining $(N - k)$ observations are assigned to the clusters with the nearest centroid by using ***Euclidean distance***:

$$dist(x, c_i) = \sqrt{(x - c_i)^2}$$

So, each observation assigned to the cluster which have a minimal $dist$ with its centroid.

Step 2: Recalculating the K centroids:

In this step, the mean of each cluster is calculated and assigned as the new centroid of the cluster.

Step 3: Recalculating the *Euclidean distances*:

Step 1 will be repeated with the new centroids that calculated in step 2, then reassigning every observation to the cluster of the nearest centroid. Note that, in this step some observations would change their possession from cluster to another with respect to the minimal $dist$.

Step 4: Repeating until no changes noticed:

Repeating steps 2 and 3, until the observations stop changing their clusters.

Note that the previous steps assumes that the data contain one variable X with m observations, which is not our case study, so keep in mind that the ***Euclidean distance*** will be little different with more variables:

$$D_i = dist(X, c_i) = \sqrt{(x_{1i} - c_i)^2 + (x_{2i} - c_i)^2 + \cdots + (x_{mi} - c_i)^2} \; ; \; \forall \, i = 1,2, \dots , k$$

and so, the process will be more complicated with multi-dimensional distances with a huge number of iterations (Bhatia, 2019).

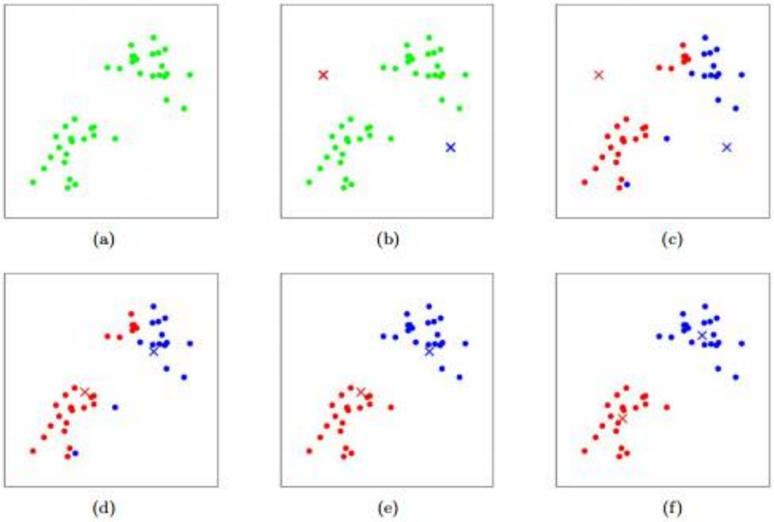


Figure (3-18) 2-mean clustering

The K-Means Algorithm:

1. Randomly assign a number, from 1 to K, to each of the observations. These serve as initial cluster assignments for the observations.
2. Iterate until the cluster assignments stop changing:
   a) For each of the K clusters, compute the cluster centroid. The Kth cluster centroid is the statistical mean for the observations in the Kth cluster.
   b) Assign each observation to the cluster whose centroid is closest (where closest is defined using Euclidean distance).

The K mean algorithm can be shown in the following flow-chart:

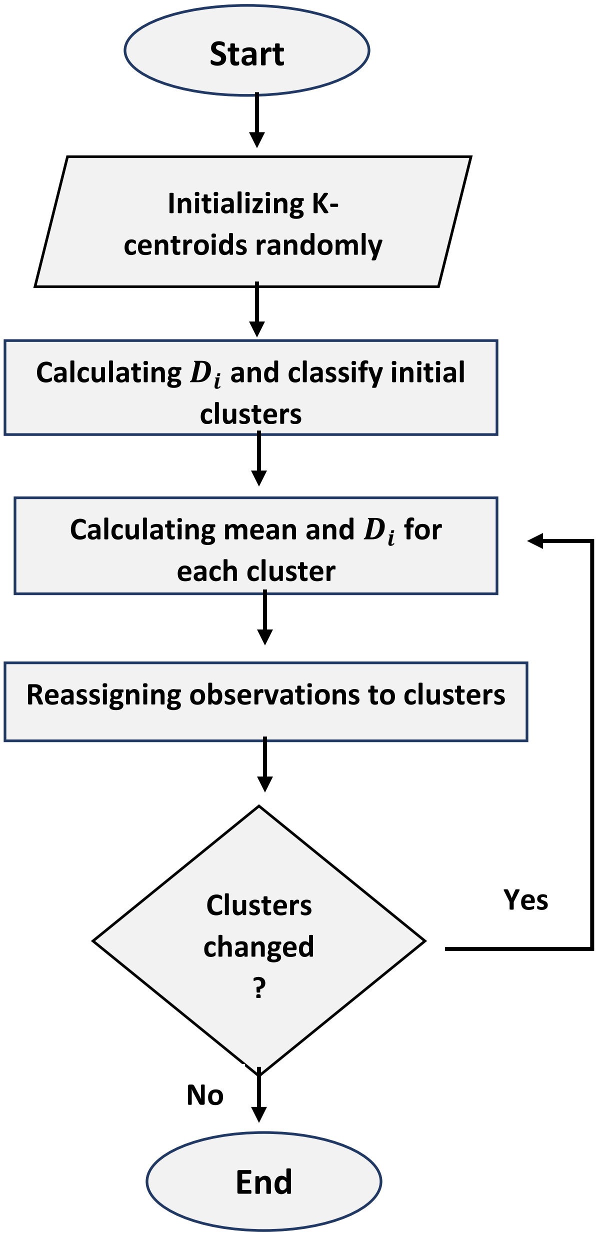

Start
Initializing K-centroids randomly
Calculating D_i and classify initial clusters
Calculating mean and D_i for each cluster
Reassigning observations to clusters
Clusters changed ?
Yes
No
End

Figure (3-19) K-mean algorithm flowchart

Specifically in our case study, the number of players is 22338 observations with four attributes (variables) in which intuitively different number of clusters would be chosen from 3 to 6 as an instance. Then the K-mean algorithm has been run four times with different K every time.

The mean and standard deviation were calculated for every attribute among all the clusters in order to pick the best number of clusters to represent the number of blocks, as shown in the following tables.

Table (3-20) Selecting clusters number with respect to their mean and standard deviation of player's attributes

Number of Clusters		**Skills**		**Physical**		**Mental**		**GK_Attributes**	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD
Three Clusters	**1**	58.31	8.84	68.32	6.38	63.78	7.64	11.81	5.51
	2	21.89	5.20	53.34	6.61	26.38	7.49	70.37	7.25
	3	61.76	7.17	70.61	5.62	52.74	8.17	11.53	5.02
Four Clusters	**1**	65.88	5.43	71.61	4.99	67.95	5.42	12.13	6.15
	2	51.02	5.64	64.75	5.82	58.80	6.69	11.57	4.88
	3	23.16	8.90	53.90	7.29	27.33	9.19	68.57	12.50
	4	60.60	6.58	70.50	5.64	50.94	6.75	11.39	4.73
Five Clusters	**1**	23.16	8.90	53.90	7.29	27.33	9.19	68.57	12.50
	2	65.39	5.41	71.54	4.95	68.06	5.43	12.09	6.08
	3	55.31	4.39	67.76	5.44	47.17	5.85	11.12	4.05
	4	50.80	5.59	64.69	5.80	58.93	6.45	11.50	4.79
	5	66.25	4.25	72.91	4.84	55.77	5.90	11.84	5.58
Six Clusters	**1**	50.71	5.53	64.59	5.82	58.50	6.40	10.90	3.74
	2	23.15	8.90	53.91	7.28	27.33	9.19	68.59	12.46
	3	65.80	5.40	71.40	5.07	67.34	5.27	10.01	1.72
	4	64.48	6.46	72.04	4.79	68.50	7.24	28.11	2.35
	5	60.35	6.55	70.44	5.66	50.46	6.47	10.78	3.57
	6	58.37	8.55	67.09	6.27	62.40	11.57	26.11	2.70

By comparing the standard deviation between the attributes for every group of clusters, clearly the first group of three clusters have the lowest range of variabilities. Also, the following graphs show existence of variability with tendency that the three clusters group is the best choice.

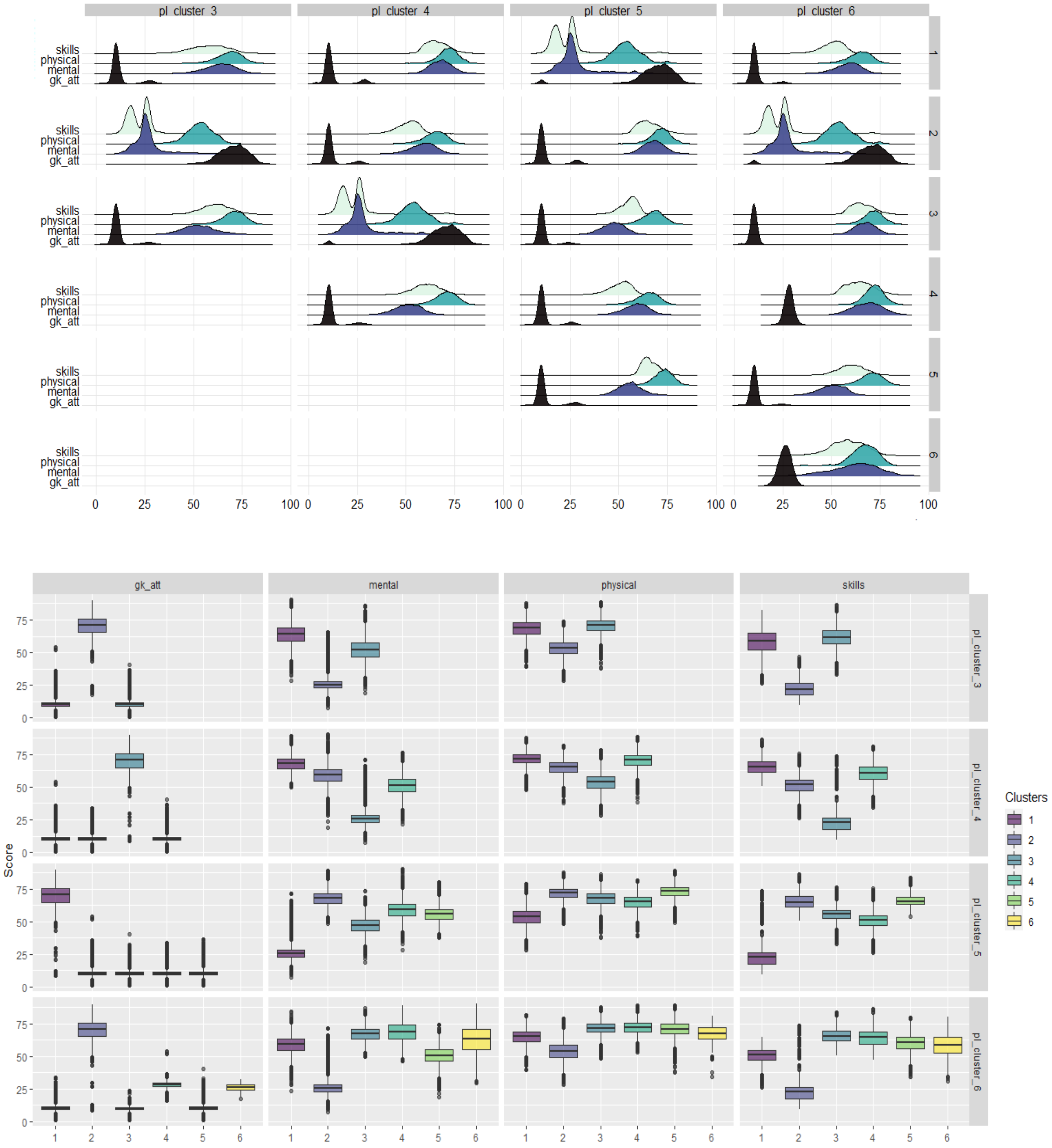


Figure (3-20) Distributions and boxplot of selecting clusters' number of player's attributes

Based on this fact, a factorial experimental design of three blocks would be introduced for further analysis; since, all the previous tables and figures clarifying variability either between players' attributes or between the leagues, which will give rise to some questions as:

1- Do the attributes have different effects within the leagues?
2- Do the leagues have different effects within the attributes?
3- Does the clustering have effect in controlling random error?
4- Do the leagues and the attributes have interaction effects between them?

In this case, a design of factorial experiment with blocks and interaction can be suggested, were the different players representing the experimental units and the randomization has to be applied to distribute random sample of them all over the factors and blocks levels.

Practically, from every league, three random samples of size four players were selected, so that one sample was taken from each cluster, then the four experimental units of each cluster was allocated to the four levels of attributes by using a Randomization Algorithm as can be seen in the following table:

Table (3-21) The random experimental units for each league per every attribute and every cluster

League		**Skills**	**Mental**	**Physical**	**GK_att**
Belgium Jupiler League	**1**	Joseph Akpala	Christophe Bertjens	Christian Kabasele	Ibrahim Ayew
	2	Jens Cools	Benjamin Nicaise	Thomas Buffel	Stephen Buyl
	3	Nana Asare	Deni Milosevic	Robin Henkens	Gunther Vanaudenaerde
England Premier League	**1**	Mathieu Flamini	Patrick Kenny,30	Antonio Luna	Daniel Agger
	2	Samir Nasri	Steven Pienaar	Armand Traore	Stephane Sessegnon
	3	Tom Cleverley	Matthew Etherington	Kevin Nolan	El Hadji Diouf
France Ligue 1	**1**	Benoit Pedretti	Morgan Sanson	Frederic Guilbert	Franck Tabanou
	2	Gael Danic	Adrien Regattin	Alexandre Raineau	Drissa Diakite
	3	Samuel Umtiti	David Ospina	Cedric Mongongu	Steed Malbranque
Germany 1. Bundesliga	**1**	Hamit Altintop	Markus Miller	Hajime Hosogai	Marco Fabian
	2	Niko Bungert	Jerome Boateng	Lars Stindl	Noah-Joel Sarenren-Bazee
	3	Johannes Flum	Per Nilsson	Yuya Osako	Mario Goetze

Italy Serie A	**1**	Jose Angel Crespo	Zeljko Brkic	Leonardo Blanchard	Nicola Dal Monte
	2	Antonio Di Natale	Afriyie Acquah	Lorenzo Squizzi	Felipe Anderson
	3	Davide Biondini	Davide Moscardelli	Thiago Ribeiro	Jonathan
Netherlands Eredivisie	**1**	Jeffrey Leiwakabessy	Nourdin Boukhari	Mitchell Schet	Rydell Poepon
	2	Nacer Chadli	Shiran Yeini	Trent Sainsbury	Joey Didulica
	3	Soeren Rieks	Johan Kappelhof	Joel Veltman	Ted van de Pavert
Poland Ekstraklasa	**1**	Lukasz Surma	Mateusz Lewandowski	Tadeusz Socha	Gergo Lovrencsics
	2	Tomasz Holota	Andre Micael	Maciej Jankowski	Michal Peskovic
	3	Pawel Oleksy	Sebastian Dudek	Mateusz Machaj	Eduards Visnakovs
Portugal Liga ZON Sagres	**1**	Bruno Mendes	Fredy Monter	Yacine Brahimi	Nilson
	2	Dani Abalo	Hernani	Diego Reyes	Goncalo Guedes
	3	Vukasin Devic	Neto	Hector Quinones	Ruca
Scotland Premier League	**1**	Jamie MacDonald	Darko Bodul	Michael McGovern	Henrik Ojamaa
	2	Paul Cairney	Nick Ross	John Potter	Stevie May
	3	Daryl Murphy	Saidy Janko	Efe Ambrose	Ian Black
Spain LIGA BBVA	**1**	Saul Berjon	Jordi Figueras	Willian Jose	Julio Alvarez
	2	Alvaro Vadillo	Juanfran	Marc Bertran	Tiago
	3	Dealbert	Marcelo	Wakaso Mubarak	Joseba Zaldua
Switzerland Super League	**1**	Marco Mathys	Sally Sarr	Birkir Bjarnason	Christoph Spycher
	2	Jahmir Hyka	Dejan Janjatovic	Breel Embolo	Giovanni Sio
	3	Sandro Lauper	Amir Abrashi	Fabian Schaer	Daniel Lopar

By choosing a random sample of size two for every player, the observed units corresponding to the previous table have been selected, and the resulting table will look like the following:

Table (3-22) The random measurable units for each league per every attribute and every cluster

League		**Skills**	**Mental**	**Physical**	**GK_att**
Belgium Jupiler League	**1**	57.00 57.00	34.40 34.40	68.00 71.12	7.00 7.00
	2	59.13 59.13	61.20 66.00	70.38 68.50	11.00 10.00
	3	65.00 65.67	46.20 47.40	67.25 67.25	8.60 8.60

England Premier League	**1**	69.53 69.87	27.60 27.60	74.88 76.62	12.40 12.40
	2	71.33 71.33	71.40 72.20	73.50 73.88	29.00 29.00
	3	66.13 67.00	58.00 58.00	70.25 72.25	26.60 26.60
France Ligue 1	**1**	73.07 70.00	71.20 43.00	67.25 67.25	10.80 11.80
	2	64.73 64.20	68.40 69.20	56.88 70.62	13.20 13.20
	3	70.47 61.27	19.20 19.20	69.12 66.00	8.60 8.60
Germany 1. Bundesliga	**1**	74.40 74.07	45.40 43.20	72.88 73.00	12.20 12.20
	2	54.00 55.47	68.00 66.40	73.62 73.88	11.20 11.20
	3	67.47 67.40	61.80 61.20	70.38 70.38	9.80 9.80
Italy Serie A	**1**	52.87 55.27	17.60 25.00	65.75 64.50	9.80 10.80
	2	73.20 73.00	69.80 71.00	60.75 43.50	8.60 8.60
	3	62.47 65.00	47.40 49.60	70.25 70.38	9.00 8.00
Netherlands Eredivisie	**1**	58.27 58.67	52.80 52.60	72.38 72.38	11.80 23.20
	2	68.20 68.37	60.40 60.40	72.12 75.00	71.60 69.40
	3	56.87 58.00	62.40 62.40	70.88 73.00	11.00 11.00
Poland Ekstraklasa	**1**	50.60 57.13	58.20 58.20	67.50 49.38	9.80 9.80
	2	43.67 50.07	50.60 51.00	74.88 74.62	62.60 62.40
	3	47.13 50.67	53.00 53.60	70.38 69.62	10.20 11.20
Portugal Liga ZON Sagres	**1**	48.58 48.58	58.40 56.80	74.00 74.38	76.00 11.60
	2	59.13 59.13	38.80 46.20	69.75 69.50	9.00 10.00
	3	6.00 36.00	26.20 65.80	67.88 68.25	9.40 9.40
	1	16.40	44.00	44.38	11.00

Scotland Premier League		16.40	51.20	43.88	11.00
	2	53.93 57.87	62.40 37.40	63.50 64.40	8.40 8.40
	3	58.40 57.47	58.20 51.40	67.00 64.75	27.80 10.60
Spain LIGA BBVA	**1**	59.33 68.40	65.00 63.40	60.25 61.88	29.60 29.80
	2	57.13 57.20	76.00 66.80	68.62 74.00	9.60 29.80
	3	57.20 53.67	77.00 50.20	83.62 84.75	10.60 10.60
Switzerland Super League	**1**	58.40 53.53	55.40 57.20	73.62 73.12	8.60 8.60
	2	53.33 57.07	66.80 66.80	70.62 79.50	12.00 12.00
	3	45.53 45.53	67.40 65.00	62.38 74.62	59.20 59.20

The mathematical model for a randomized complete block factorial design can be written as:

$$y_{ijkl} = \mu + \alpha_i + \beta_j + b_k + (\alpha\beta)_{ij} + \epsilon_{ijkl} \;; \quad \forall\, i = 1, \dots, 11\,; j = 1, \dots, 4\,; k = 1, 2, 3; l = 1,2.$$

Where: y_{ijkl} represents every observation, μ represents the general mean, α_i represents the effect of the ith level of first factor which is Leagues, in other words it represents the difference between the ith League's attributes mean and the general data mean, β_j represents the effect of the jth level of second factor which is players' attributes, in other words it represents the difference between the jth Attribute's mean and the general data mean, b_k represents the effect of kth block, $(\alpha\beta)_{ij}$ represents the effect of interaction between ith League and jth Attribute, and finally, ϵ_{ijkl} the random error represents the difference between every observation and its cell mean (Lawson, 2014).

Where model's errors must guarantee the usual assumptions of normality, $\epsilon_{ijkl} \sim N(0, \sigma^2)$, and independence. The independence assumption is guaranteed as the treatment combinations are randomly assigned to the experimental units, and the equal variance and normality assumptions may be verified with a residual versus predicted plot and a normal probability plot of the residuals as described in the following figure:

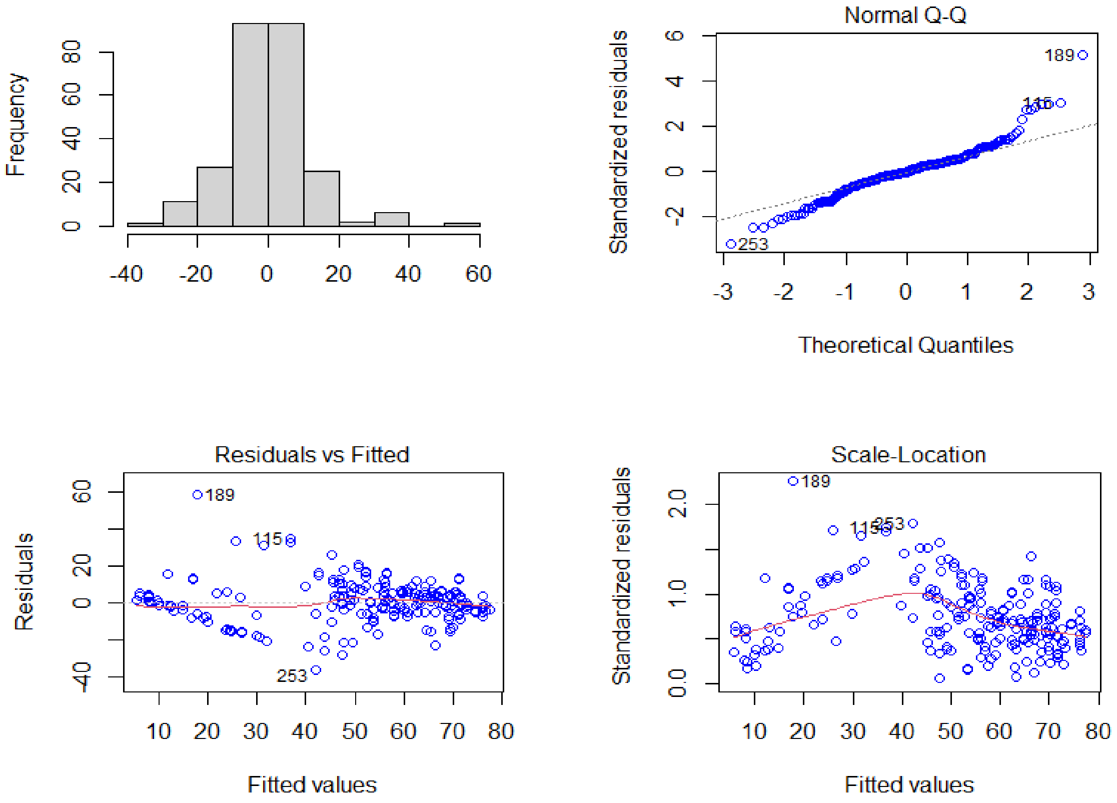


Figure (3-22) Graphs of satisfying the design model assumptions

The statistical hypotheses of this design can be stated as the following:

1- $H_o: \alpha_i = 0 \quad vs \quad H_1: \alpha_i \neq 0$
2- $H_o: \beta_j = 0 \quad vs \quad H_1: \beta_j \neq 0$
3- $H_o: b_k = 0 \quad vs \quad H_1: b_k \neq 0$
4- $H_o: (\alpha\beta)_{ij} = 0 \quad vs \quad H_1: (\alpha\beta)_{ij} \neq 0$

By applying the suitable analysis of variance for **Table (3-22),** the result has showed in the following table:

Table (3-23) Results of the ANOVA table

Factors	D.F	Sum Sq.	Mean Sq.	F value	Pr. (<F)
Leagues	10	5100	510	3.282	0.00057
Player's Attributes	3	93288	31096	200.13	< 2e-16
Blocks (Clusters)	2	1883	941.5	6.059	0.00276
Player's Attributes* Leagues	30	8075	269.17	1.732	0.01407
Residuals	219	33096	151.123		

Note that rejecting the null hypothesis of the blocks effect gives an evidence on the clustering effect in controlling the random error, on the other hand, all the probability values for the leagues, the player's attributes and the interaction between them are less than the level of significance, so the test is significant, so the leagues and the player's attributes have different effects, with clear existence of interactions between leagues and player's attributes as showed in the following figure.

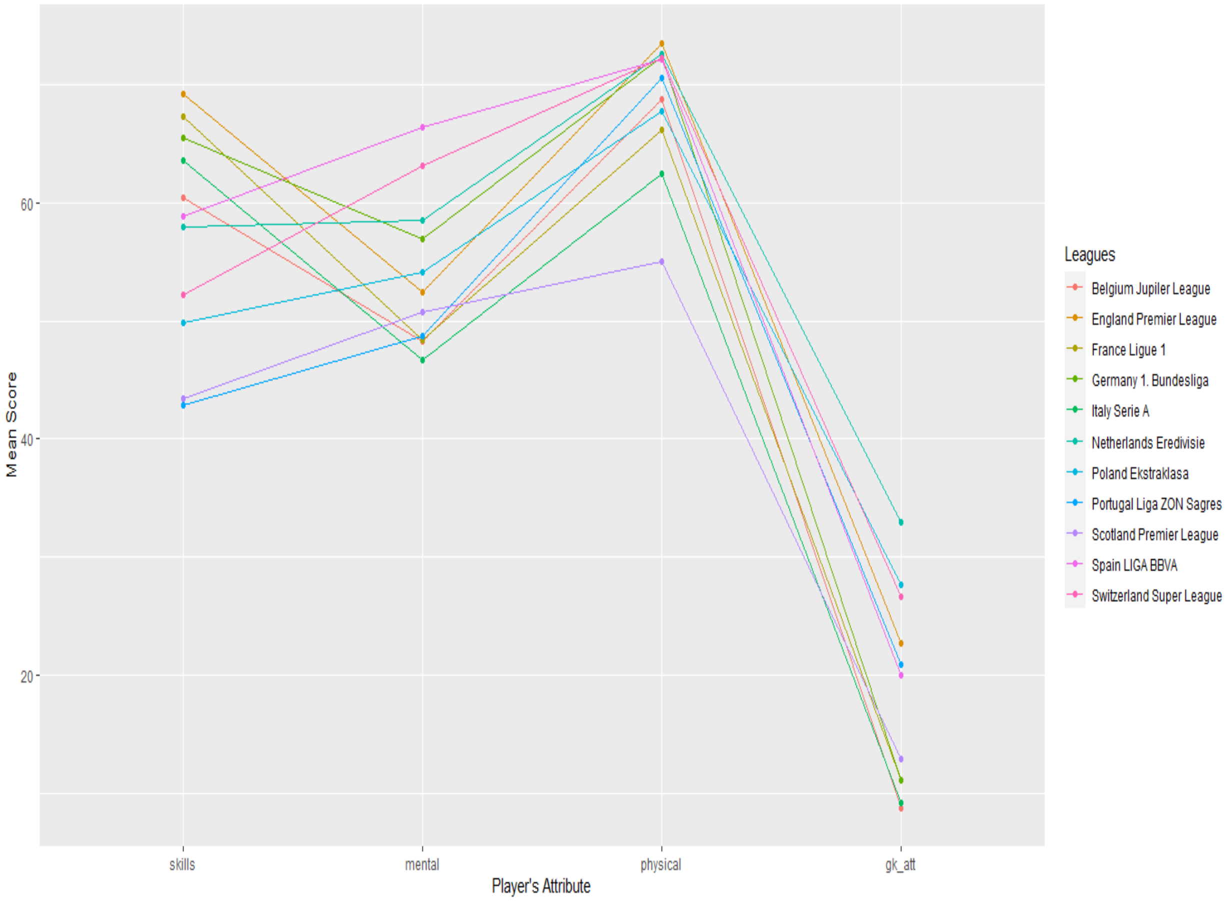


Figure (3-23) The interaction between leagues and player's attributes levels

Due to rejection of the null hypothesis, it would be fruitful to know which pairs of player's attributes and pairs of leagues are responsible for these differences according to the following hypothesis.

1- $H_o: \alpha_i = \alpha_j \quad vs \quad H_1: \alpha_i \neq \alpha_j; \ \forall \ i \neq j$

2- $H_o: \beta_i = \beta_j \quad vs \quad H_1: \beta_i \neq \beta_j; \ \forall \ i \neq j$

Because of large number of possible comparison pairs, only the p-values with rejecting of the null hypothesis for Bonferroni and Tukey tests have shown in the following table.

Table (3-24) Multiple comparisons by Tukey and Bonferroni for leagues and player’s attributes levels

Player’ s Attributes	Comparisons	Bonf. P-adj	Tukey P-adj
	mental-gk_att	0.0000	0.0000
	skills-gk_att	0.0000	0.0000
	physical-gk_att	0.0000	0.0000
	physical-mental	0.0000	0.0000
	physical-skills	0.0000	0.0000
Leagues	**Netherlands Eredivisie-Scotland Premier League**	1	0.0040
	Spain LIGA BBVA- Scotland Premier League	1	0.00204
	Switzerland Super League-Scotland Premier League	1	0.00402
	England Premier League-Scotland Premier League	1	0.00183
	England Premier League-Italy Serie A	1	0.03150
	Netherlands Eredivisie-Italy Serie A	1	0.01141
	Spain LIGA BBVA- Italy Serie A	1	0.03351
	England Premier League-Portugal Liga ZON Sagres	1	0.03619
	Spain LIGA BBVA- Portugal Liga ZON Sagres	1	0.03835

It is clear from the above table that all player’s attributes pairs have different effects except the **mental** and **skills** attributes, while only the showed nine leagues’ pairs from all the fifty-five have different effects.

In general, we cannot depend on Bonferroni compression test as the possible compression pairs are fifty-five which will decrease its efficiency comparing with Tukey test.

3.1.3. View on Correlations between Input and Output Data:

To build an accurate and convenient predictive model, one need to have a close look on the relationship between input and output variables, how powerful it is, and the trend of its direction.

As mentioned before that players’ and teams’ attributes would be input variables, while matches’ results would be output variables, and with matches’ results we mean home, away and total goals ratios, number of wins, and which team won the match, home, away or ended with draw. In this section correlation matrices will be calculated to explore the relationships using Pearson and Spearman coefficients of correlation depending on the type of data, quantitative or qualitative.

The following figure studies the behavior of three input variables which are **Build Up Play**, **Chance Creation**, and **Defense**, in respect to three output variables which are **Away Goals Ratio**, **Home Goals Ratio**, and **Total Goals Ratio**, using Pearson's coefficient of correlation.

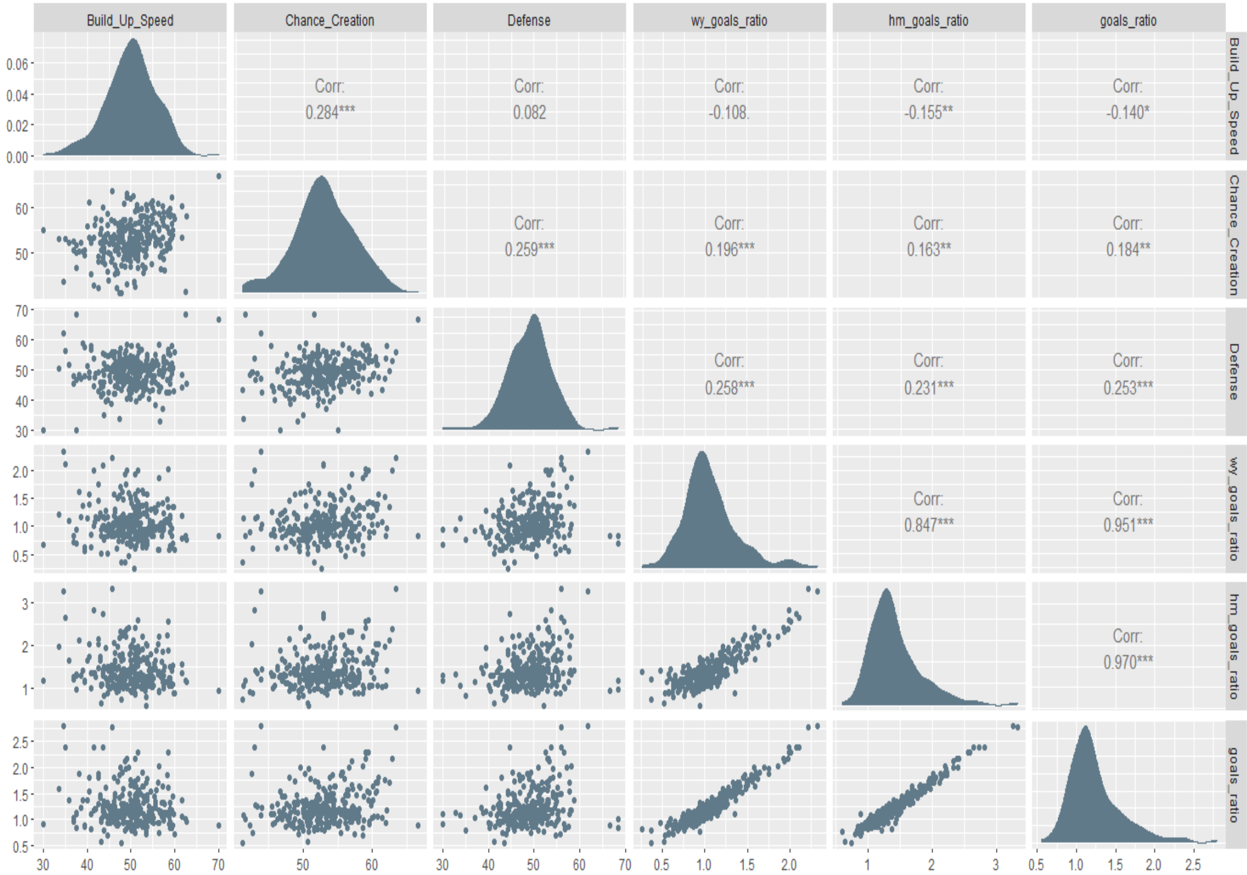


Figure (3-24) Pearson's coefficient of correlation between team's attributes and goals results

First of all, it is clear that the correlations between input and output variables are not so high, but still there is something to say. Note that there is an inverse proportion between **Build Up Play** and the **Goals Ratios**, especially the **Home Goals Ratio**, that means, for teams, the more focusing on building up the play the less goals they score. While, **Defense** is directly proportional to **Goals Ratios**, which means that teams with highly defense have the opportunity to score more goals.

The following figure studies the correlations between teams' attributes and the number of matches they won, and the mean difference of goals they score:

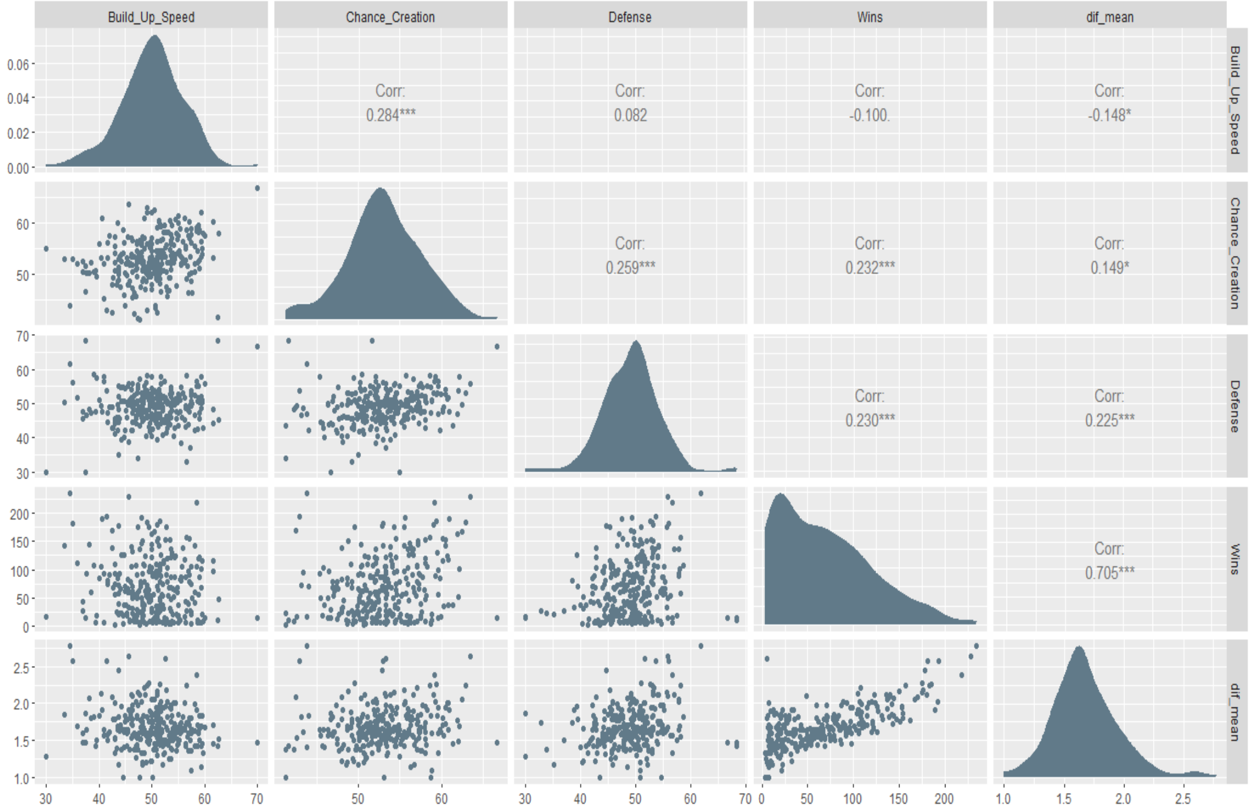


Figure (3-25) Pearson's coefficient of correlation between team's attributes and teams' results

It does not sound so different from the other; teams focusing on building up the play have less opportunity to win while focusing on defending, and chance creation gives them more chance to win.

Repeating the same calculation, but with players' attributes as the input variables showing that Skills, Mental and Physical are directly proportional with goals ratios and number of wins. Note that the players' attributes have a more robust correlation with matches' results.

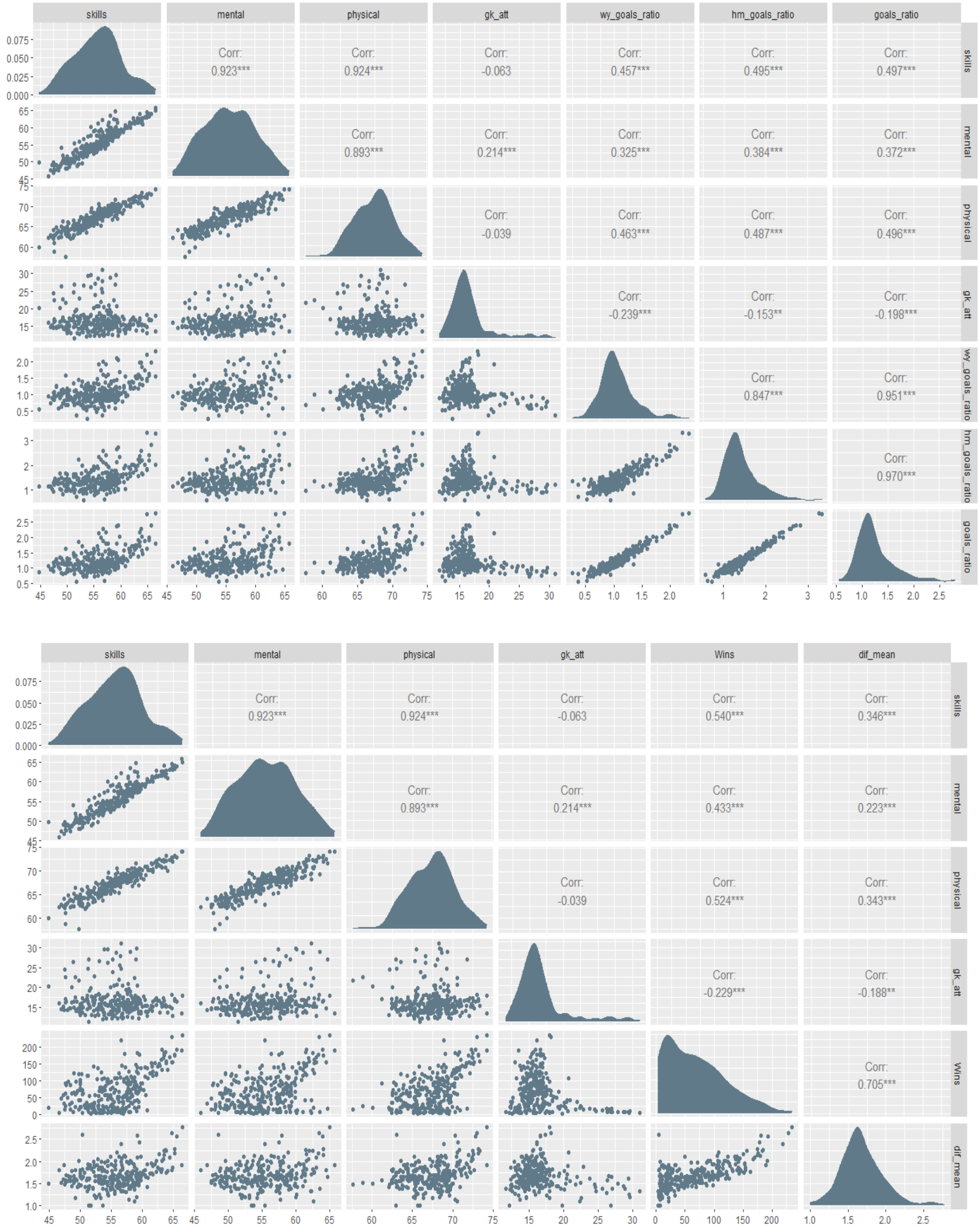


Figure (3-26) Pearson's coefficient of correlation between player's attributes and matches results

Obviously, the skills of the players are different according to their position on the field; that is to say that the midfielder and attackers have low skills of the goalkeepers as well as the goalkeepers have low skills of the midfielder and the attackers, that explain why the correlation between them is negative as shown from the two previous figures.

Note that in previous figures the correlations calculated for every team between its input (teams' and players' attributes) and its output (team results), from now on, the correlations will be calculated for every match between its input (attributes for the two teams and their players) and its output (match results).

Moving to qualitative input and output variables, the next figure shows the Spearman coefficients of correlation between **Winner** and the two teams' players' attributes. Where **Winner** represents which team have won the match ranked ascendingly: **Home Team Won**, **Draw**, and **Away Team won**, while players' attributes categorized according to the FIFA's scales to:

Table (3-25) FIFA categorical ranks

Excellent	90 to 99
Very Good	80 to 89
Good	70 to 79
Fair	50 to 69
Poor	40 to 49
Very Poor	0 to 39

Although the correlations between Winner and other variables is low, but some insights could be gotten, such as, there is an inverse proportion between Winner and the Physical attributes of the players of home team, while there is direct proportion between winner and the Physical attributes of the players of away team, that means the team with the higher score of physical attributes has the better opportunity to win, and the same thing for the Mental players' attributes.

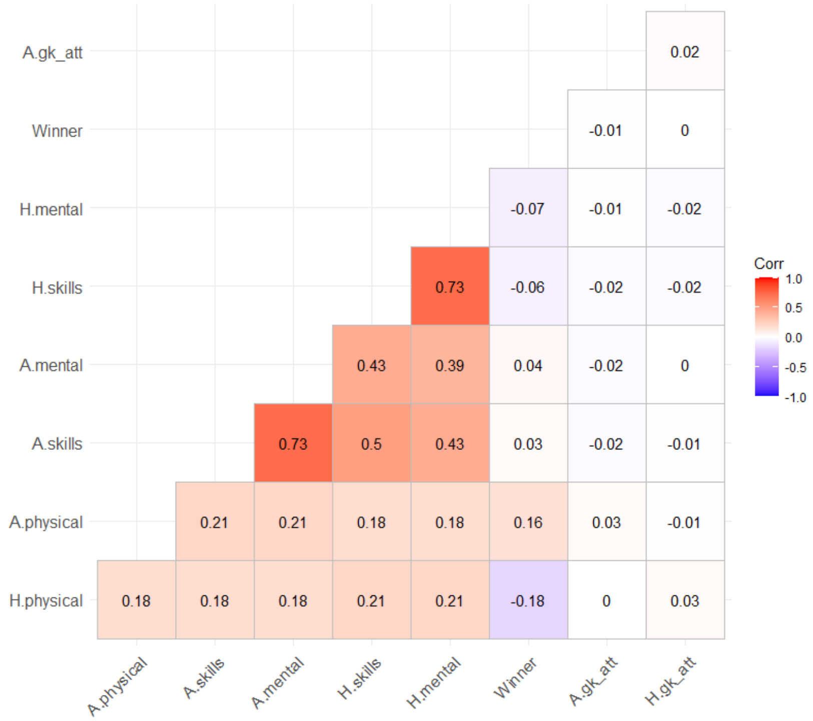


Figure (3-27) Spearman’s coefficient of correlation between plyer’s attributes and matches results

The next figure shows correlations between Winner and teams’ attributes of bulding up the play:

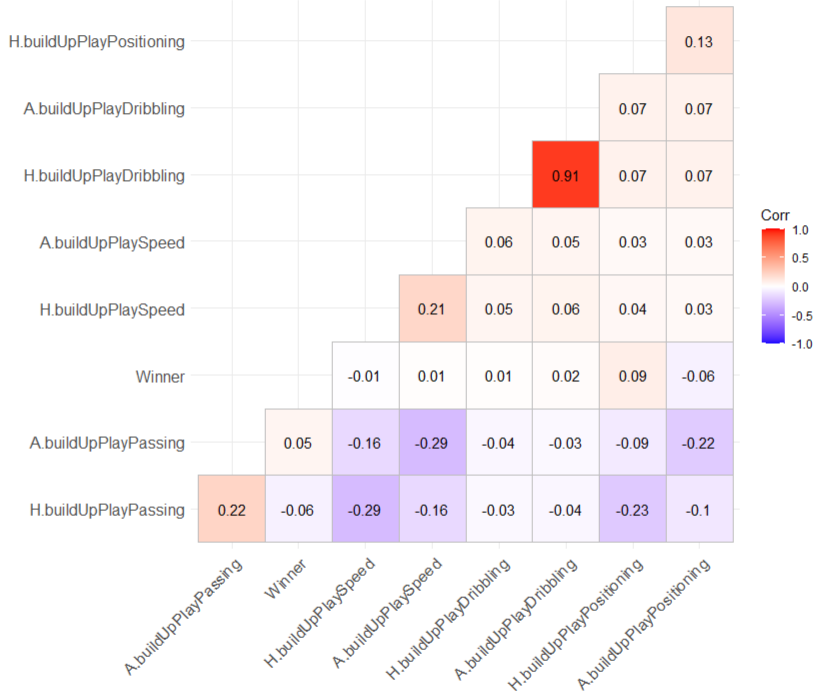


Figure (3-28) Spearman’s coefficient of correlation between build up play attributes and matches results

Although the correlations between Winner and other variables is low, but still gives some ideas, such as, the inverse proportion between Winner and Build Up Play Positioning of the away team gives a hint that the team which focuses on organized positioning in building up the play have the less opportunity to win. While the direct proportion between Winner and Build Up Play Passing of the away team gives a hint that the team which focuses on building up play by short passing have the better opportunity to win.

The following figure shows correlations between Winner and teams' attributes of Chance Creation:

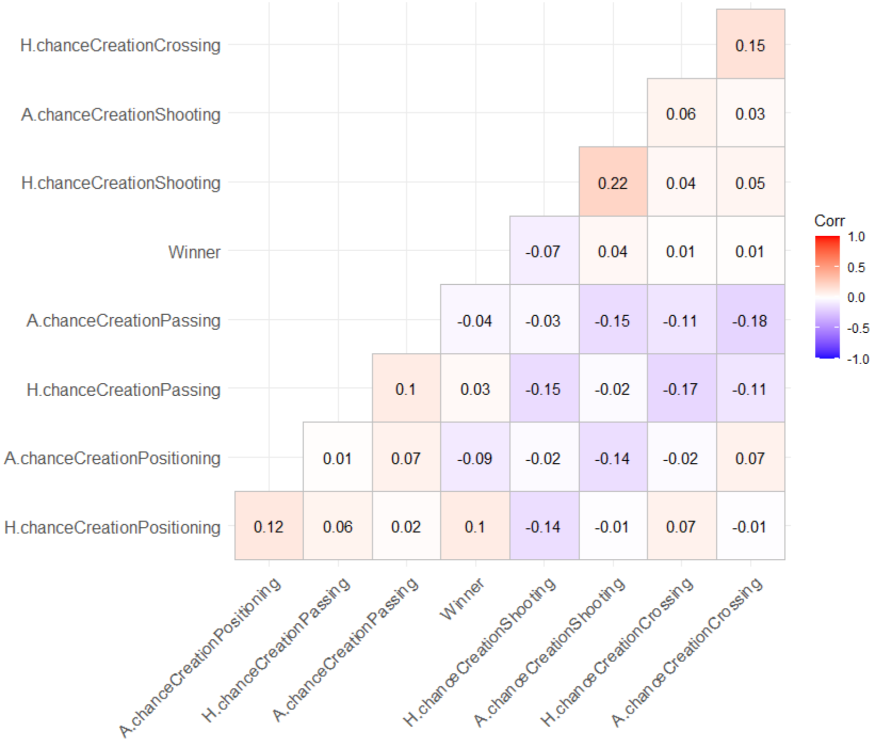


Figure (3-29) Spearman's coefficient of correlation between chance creation attributes and matches results

The inverse proportion between Winner and Chance Creation Positioning of the away team gives a hint that the team which focuses on organized positioning while creating the chance have the less opportunity to win. While the direct proportion between Winner and Chance Creation Shooting of the away team gives a hint that the team which focuses on shooting while creating the chance have the better opportunity to win.

Finally, from the following figure, the inverse proportion between Winner and Defense Defender Line of the home team gives a hint that the team which play with offside trap on defender line have a better opportunity to win, while the direct proportion between Winner and Defense Pressure of away team gives a hint that the team which play with high defense pressure have better opportunity to win.

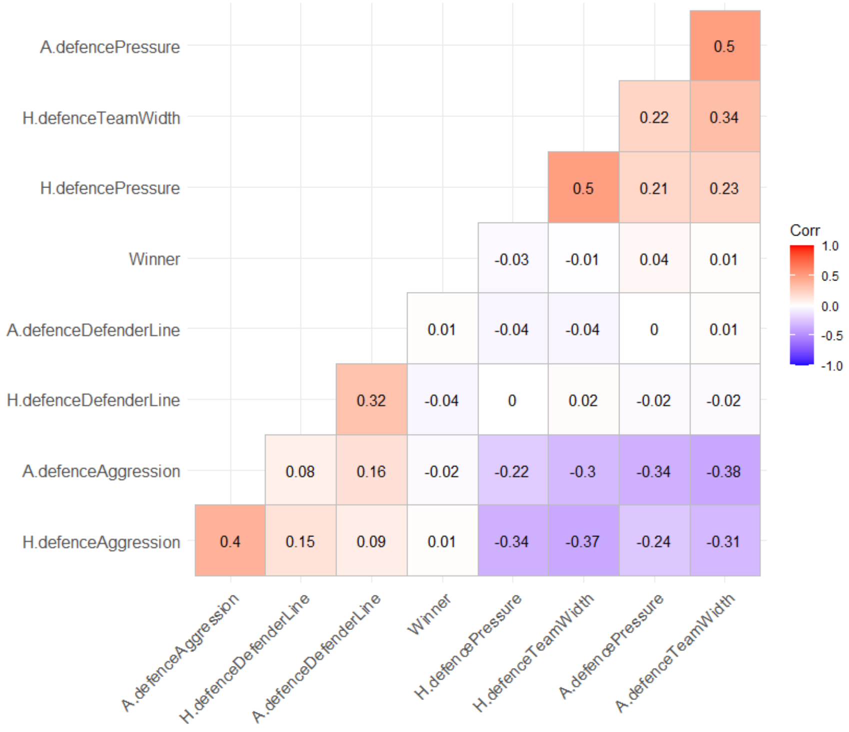


Figure (3-30) Spearman's coefficient of correlation between defense attributes and matches results

3.1.4. Summary of Exploratory Analysis

After all the presented statistical measures and inference, we may summarize the results of the exploratory analysis in the following points:

1. In this case study, teams' attributes, players' attributes, and leagues information represent the probably input data for any predictive model, while matches' results represent the output data.
2. From the general view on teams' and players' attributes, we noticed that teams' variables are more homogenous than players', while the quantitative attributes could be aggregated in three variables for teams and four variables for players, respectively, to FIFA's categorization.

3. All the confidence intervals calculated for teams' and players' attributes are short, especially for players' attributes; this could be due to the large sample size, which gives more accuracy to the estimators, and evidence of their consistency property.
4. In matches results, it is clear that: most of the matches end with home team's win, Barcelona and Real Madrid have the best results in teams with a slight lead for Barcelona, while unexpectedly, the Dutch and Swiss leagues have the highest goals ratios, and this may be due to the lack of competition between their teams.
5. Using the experimental design models to study the effects of teams' and players' attributes according to their leagues, we concluded that there is no statistical evidence of a difference in the effects of teams' attributes. In contrast, the players' attributes have different effects according to leagues, except the same mental and physical attributes. Also, there are interactions between players' attributes and their leagues.
6. Cluster analysis has been shown an effectively partition the experimental units into blocks to control the random error.
7. The players' attributes increase the chance of scoring goals and winning more than the team's attributes, especially the physical and mental attributes. In contrast, the team's defense attributes have the most contribution in increasing the chance to win comparing to all teams' attributes.
8. By studying the qualitative teams' attributes, we noticed that focusing on building up the play and creating the chance with organized positioning may decrease the team's opportunity to win, while building up the play with short passing and creating the chance with shooting may increase opportunity. In addition, playing with high defense pressure and using the offside trap on the defender line increase the team's opportunity to win.
9. From 7 and 8, we can say that the teams whom have the proficiency to deal with counterattacks, whether they are attacking or defending, have more opportunities to win the game.

3.2. Setting Goals:

The fourth step of the data science process is to set goals, which are to identify all possible appropriate predictive models. In our case study, many goals could be established, depending on output variables, such as:

1. **Decision Tree Model:** to predict the value of Winner variable depending on qualitative variables of teams' and player's attributes for each team in the match in addition to league's variable, since the dependent variable of the model (Winner) is qualitative, ordinal, and with more than two classes.
2. **Poisson Regression Model:** to predict the number of match goals, or each team goals, depending on qualitative variables of teams' and player's attributes for each team in the match in addition to league's variable, since the dependent variable of the model (Match goals or Team goals) is representing the quantity of goals and probably distributed as Poisson distribution.
3. **Logistic Regression Model:** to predict the result of every single team in each match if it is going to win or not, depending on quantitative players' and teams' attributes, since the dependent variable of the model is binary and probably distributed as Binomial distribution.
4. **Time Series Model:** to predict the Overall Rating variable of every player, depending on the rest of the players' attributes, attributes of the team, and the league that he belongs to, in order of time.
5. **Multiple Regression Model:** to predict the mean goals difference for each team per year, depending on its attributes, players' attributes, and league, using Principal Component Analysis to detect the most valuable contributions of input variables.
6. **Neural Network Model:** to take advantage of every single observation that in the database and use it to predict the results of every league's champion.

Bearing in mind the following:

1. The mentioned models are not all the possible models for this data.
2. Before using any model, one needs to check its assumptions and conditions.

In the next chapter, we will introduce **Classification Analysis** and **Decision Tree** with some details, then use them to build a model that predicts the value of the Winner variable of each match.

Chapter 4

Chapter 4:

Data Modelling and Making Predictions

With prepared data and a good understanding of it, now the process is ready to build models for making accurate predictions. This phase is much more focused than the exploratory analysis step because we know what we are looking for, and what we want the outcome to be. As suggested before, many models could be fitting with the data; one of them is using a decision tree to predict the results of matches. Before doing that, some details about classification analysis must be concerned, bearing in mind that all the upcoming procedures belong to **Data Mining** and **Machine Learning** fields, so some terms of computer sciences and database systems will be used frequently, such as tuple, learning, features, etc.

4.1. Introduction for Classification:

One of the main goals of statistical analysis is to use a given set of observations to catch the hidden relationships in the data, establish connections between random input and output variables, and build a dependence model. The **Model of dependence** is the mathematical record of how one or several variables depend on other variables. The Model can be described as a formula, an equation or system of equations, a set of logic statements, or graphically as a decision tree. This Model can be used for forecasting values of a variable depending on a tuple of responses of other variables, also used to make critical decisions. Nowadays, databases are used for making intelligent decisions (Tan, Steinbach & Kumar, 2006).

Two forms of data analysis, namely **Regression**, and **Classification** are used for predicting future trends by analyzing existing data. Regression models predict a value of a scaled variable (Quantitative data), while Classification models predict a value or class of nominal or ordinal variable (Qualitative data) (Aggarwal, 2015). For example, building a classification model is used to predict whether a football team will win the match or not, while regression can be used to predict the number of goals that this team will score in the match.

4.1.1. Definition: Classification is a classical method used by statisticians and machine learning researchers to predict the outcome of unknown samples. It is used to categorize objects (or things) into a given discrete number of classes. In other words, It's the process of dividing the datasets into different categories or groups by adding the data points to a particular labeled group based on some conditions or models, such models, called classifiers. For example, we can build a classification model to categorize whether a student's results in the first semesters will allow him to graduate or not. Classification has numerous applications, including fraud detection, target marketing, performance prediction, manufacturing, and medical diagnosis (Han, Pei, & Kamber, 2011).

According to the dependent variable, classification problems are of two types, either binary or multiclass. In binary classification, the target variable can only have two possible values (classes); for example, a team will either win or lose, a student will graduate or not. While in multiclass classification, the target variable can have more than two values; for example, news stories can be classified as weather, finance, entertainment, or sports news; the match will end for the home team, away team, or with a draw (Shmueli, et al., 2017).

4.1.2. Steps of classification:

There are two classification steps: first is **training** the model, which is also called the **learning step**, and the second is **testing** the model for accuracy, which is also called the **prediction step**. In the first step, a classifier is built based on the **training data** by analyzing tuples of input variables (independent variables) and the associated class label in the output variable (dependent variable). By analyzing training data, the system estimates and creates some prediction rules, which could be understood as a summary model for the responses of the dependent variable in the training data set. In the second step, these prediction rules are tested on observed data, i.e., **test data**. In this step, rules are used to predict the classes' labels of the dependent variable, then calculate the accuracy of the classifier's predictions by comparing predicted responses with the observed ones.

If the accuracy of the classifier is satisfactory, then one can use it to classify future data observations with unknown class labels. Since, the quality of the classifier depends upon the quality of the training data. If there are two or more classes, sufficient training data should be available for each class (Bhatia, 2019).

4.1.3. Classification Techniques:

Due to the widespread use of classification in numerous fields, there are various techniques of classification, such as **decision tree** classifiers, **rule-based** classifiers, **neural networks**, and **Naïve-Bayes** classifiers. Each technique employs an algorithm to identify a model that fits a relationship between the input data set and the class label of the output data. The model generated by an algorithm should do both: provide the input data well and correctly predict the unknown records of class labels. Therefore, a vital objective of the algorithm is to build models with good generalization capability, i.e., models that accurately predict the class labels of previously unknown records (Han, Pei, & Kamber, 2011).

Statisticians and computer scientists have developed several classification methods. Regarding understanding these classification methods, it would be helpful to give a few words about each of them where the **Decision Tree** will be discussed with some details in this chapter.

4.1.3.1. Decision Tree:

- Graphical representation of all the possible solutions.
- Is a flowchart-like tree structure, where each **internal node** denotes a test on a variable, each **branch** represents an outcome of the test, and each **leaf node** (or **terminal node**) holds a class label. The topmost node in a tree is the **root node**.
- Based on some conditions, which can be easy to explain.

4.1.3.2. Rule-Based Classifiers:

- Uses a set of "if-then" rules $R = \{R1 \ldots Rm\}$ to match *antecedents* to *consequents*.
- A rule is typically expressed in the following form:

  IF *Condition* **THEN** *Conclusion*.

- A decision tree may be viewed as a particular case of a rule-based classifier, in which each path of the decision tree corresponds to a rule.

4.1.3.3. Naïve-Bayes:

- Classification technique based on Bayes' theorem for conditional probabilities.
- Assumes that the effect of a variable value on a given class is independent of the values of the other variables. This assumption is called **class conditional independence**. It is made to simplify the computations involved, and in this sense, is considered "**Naïve**."

4.1.3.4. Neural Networks:

- A model of simulation of the human nervous system. The human nervous system is composed of cells, referred to as neurons.
- **Neural network** is a set of connected input/output units in which each connection has a weight associated with it. During the learning phase, the network learns by adjusting the weights to predict the correct class label of the input data.

- ❖ It has a high tolerance of noisy data and their ability to classify patterns on which they have not been trained. It can be used when you may have little knowledge of the relationships between variables and classes (Han, Pei, & Kamber, 2011).

4.2.1. Introduction to Decision Trees:

Decision trees are a classification methodology wherein the classification process is modeled using a set of **hierarchical** decisions on the input variables, arranged in a flowchart like tree structure. In the decision tree classifier, predictions are made using multiple '**if…then...**' conditions similar to the control statements in different programming languages.

Decision tree-based classification is very similar to a '**20 questions game**'. In this game, one player writes something on a page, and another player has to find what was written by asking at most 20 questions, which their answers can only be yes or no. Similarly, we can solve a classification problem by asking a series of carefully crafted questions about the input variables. Each time we receive an answer, a follow-up question is asked until we conclude the class label of the data tuple. The series of questions and their possible answers can be organized in the form of a decision tree (Tan, Steinbach & Kumar, 2006).

4.2.2. Structure and Terminology:

The decision tree structure consists of a root node, branches and leaf nodes. Each internal node represents a condition on some independent variable, each branch specifies the outcome of the condition and each leaf node holds a class label. The root node is the topmost node in the tree.

To understand the structure of a decision tree, let us consider the following example of a recognition problem. During a doctor's examination of some patients the following variables are determined:

Suppose $\underline{X} = [\, X_1, X_2, X_3]$ is a vector of three independent variables represents the symptoms of disease; Where: $X_1 =$ temperature, $X_2 =$ coughing, $X_3 =$ a reddening throat.

While $Y = \{\, y_1, y_2, y_3, y_4, y_5\}$, represents the dependent random variable, with five labels (classes): a cold, quinsy, influenza, pneumonia, healthy; Which from the possible diagnoses.

Now, it's required to find a model that diagnose the symptoms and determine the patient's case, where Y depends on $\underline{X}$. The example **Figure (4-1)** illustrates such a model, which can be seen as a decision tree (Shmueli, et al., 2017).

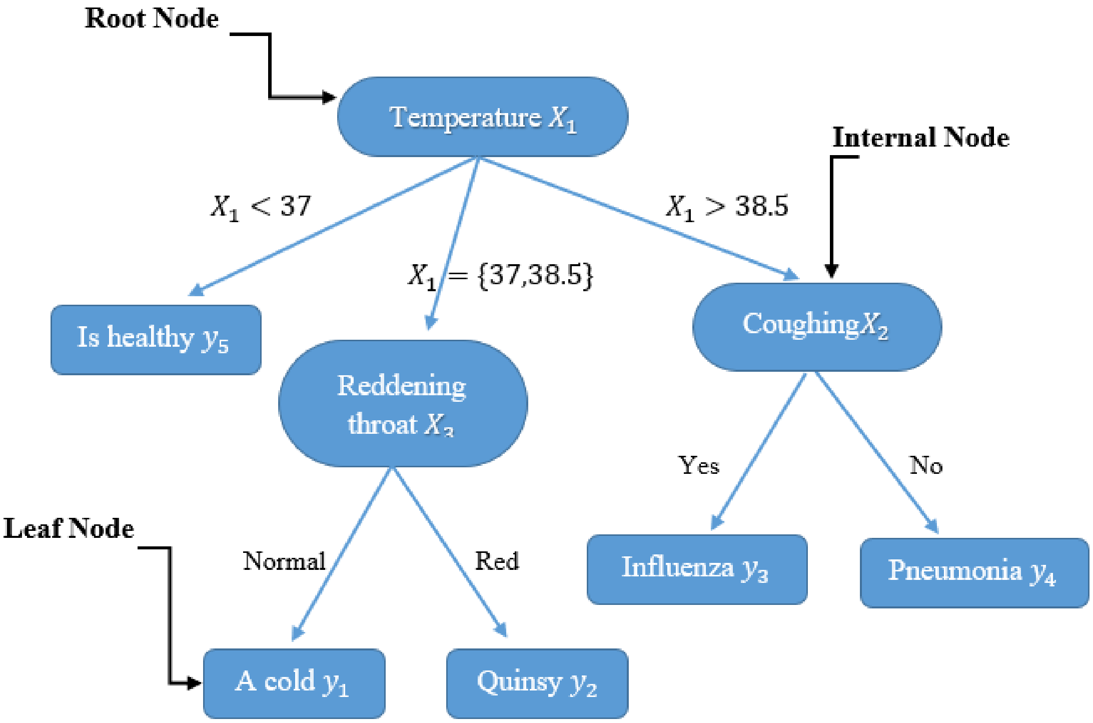


Figure (4-1) Decision tree for diagnosis

Furthermore, here is some of the decision tree terminology:

- **Root Node:** It represents the entire population or sample, and this further gets divided into two or more homogeneous sets. A root node has no incoming edges and zero or more outgoing edges.
- **Internal Nodes:** each of which has exactly one incoming edge and two or more outgoing edges.
- **Leaf Node:** A node cannot be further divided into other nodes. Leaf Nodes each of which has exactly one incoming edge and no outgoing edges.
- **Branch:** Is formed by splitting the node.

A decision tree is usually drawn from left to right or beginning from the root downwards, since from each internal node (i.e., not a leaf) may grow out two or more branches, while each node corresponds with a variable, and the branches correspond with a range of values which must give a partition of the whole domain of the given variable. while class Y is ascribed for each terminal node of a tree (named "leaf") (Tan, Steinbach & Kumar, 2006).

For any tuple of $\underline{X}$, using a decision tree, we can find the predicted response Y; for this purpose, first starting with a root of a tree by considering the variable that corresponds to the root, then define to which branch the observed value of the given variable corresponds, after that, considering the node in which the given branch comes, and repeat the same operations for this node, etc. until reaching a leaf. The value y_i ascribed to ith leaf will be the forecast for $\underline{x}$. Thus, the decision tree gives the model T of dependence Y from $\underline{X}$: $Y = T(\underline{X})$.

A set of logic statements about values of characteristics corresponds to decision trees; each statement is obtained by passing the way from root to leaf, so, for example, for the tree represented on **Figure (4-1)**, the following list of statements corresponds to:

1. If $X_1 < 37$, then $Y =$*"is health"*.
2. If $X_1 \in [37,38.5]$ and $X_3 =$*"there is no reddening of throat"*, then $Y =$*"to catch cold"*.
3. If $X_1 \in [37,38.5]$ and $X_3 =$*"there is reddening of throat"*, then $Y =$*" Quinsy"*.
4. If $X_1 > 38.5$ and $X_2 =$*"there is no cough"*, then $Y =$*"influenza"*.
5. If $X_1 > 38.5$ and $X_2 =$*"there is cough"*, then $Y =$*"pneumonia"*.

Thus, the decision tree represents a logical model of regularities of the researched phenomenon.

4.2.3. How to Build a Decision Tree?

The decision at a particular node of the tree, which is referred to as the ***split criterion***, is typically a condition on one or more variables in the training data, since the split criterion divides the training data into two or more parts, for example, from **Figure (4-1)** consider the case where *temperature* is a variable, and the split criterion is *temperature* ≤ 37; in this case, the left branch of the decision tree contains all training examples with *temperature* at most 37, whereas other branches contain all examples with *temperature* greater than 37. The goal is to identify a split criterion that could present the best possible partitioning to the tree nodes. Each node in the decision tree logically represents a subset of the data space defined by the combination of split criteria in the nodes above it. The goal of the split criterion is to maximize the separation of the different classes among the internal nodes.

During the late 1970s and early 1980s, J. Ross Quinlan, a researcher in machine learning, developed a decision tree algorithm known as **ID3**. Quinlan later proposed **C4.5** (a successor of ID3), which became a benchmark to which newer supervised learning algorithms are often compared. The decision tree is a standard machine learning technique implemented in many machine learning tools, like **Weka**, **R**, **MATLAB**, and some programming languages such as **Python**, **Java**, etc. (Bhatia, 2019).

Many of these algorithms require the determination of the “best” split from a set of choices; specifically, it is needed to choose from multiple variables for splitting each variable. Therefore, a measure of split quality is required which also called **Attribute Selection Measures (ASM)**. The algorithms mentioned before are based on the concept of **Information Gain** and **Gini Index**.

4.2.3.1. Split Criterion Based on the Information Gain:

Since some decision tree algorithms work based on **information Gain**; so, let us first understand the concept of **information theory**. It has been observed that information is directly related to uncertainty. If there is uncertainty, then there is information. If there is no uncertainty, then there is no information; for example, if a coin is biased having a head on both sides, then the result of tossing it does not give any information, but if a coin is unbiased, having a head or a tail then the result of the toss provides some information.

Usually, the newspaper carries the news that provides maximum information, for example, consider the case of a (Brazil vs. Tunisia) world cup football match, it appears certain that Brazil will beat Tunisia, so this news will not appear on the front page as main headlines, but if Tunisia beats Brazil in a world cup football match, then this news being very unexpected (uncertain) will appear on the front page as headlines.

Let us consider another example, if in a university or college, there is a holiday on Friday, then a notice regarding the same will not carry any information (because it is certain), but if some particular Friday becomes a working day, then it will be information and henceforth becomes a piece of news (Han, Pei, & Kamber, 2011).

From these examples, we can observe that information is related to the probability of occurrence of an event, since the vital question to consider is whether the probability of occurrence of an event is more; the information gain will be more frequent or less frequent? It is guaranteed from the above examples that “more certain” events such as Brazil defeating Tunisia in football or Friday being a holiday carry very little information. However, if Tunisia beats Brazil or Friday is a working day, then even though the probability of these events is less than the previous event, it will carry more information. Hence, less probability means more information.

Information theory was developed by **Claude Shannon**. Whose introduce the concept of **Entropy** which is defined as the average amount of information given by a source of data. Entropy is measured as follows.

$$Entropy\ (P_1, P_2, \dots, P_k) = -P_1 \log_2 P_1 - P_2 \log_2 P_2 - \cdots - P_k \log_2 P_k = -\sum_i^k P_i \log_2 P_i$$

Which also known as the **overall information** (I) for an event.

In this, information is defined as $-P_i \log_2 P_i$, where P_i is the probability of some event. Since, probability P_i is always less than 1, $\log_2 P_i$ is always negative; thus, getting the overall information as positive.

To calculate the information for the event of throwing an unbiased dice, with six possible outcomes and equally likely probabilities:

$$P_i = \frac{1}{6} \text{ ; } \forall \ i = 1, \dots ,6$$

$$I = Entropy\ (P_1, P_2, \dots , P_k) = -\sum_{i=1}^{6} \frac{1}{6} \log_2 \frac{1}{6}$$

$$I = 6 \left(\frac{1}{6} \log_2 \frac{1}{6}\right) = 2.585$$

But, if the dice is biased such that there is a 50% chance of getting a 6, then the information content of rolling the die would be lower as given below:

$$\mathrm{I} = -0.5 \log_2 0.5 - \sum_{i=2}^{6} 0.1 \log_2 0.1 = 2.16$$

And if the dice is further biased such that there is a 75% chance of getting a 6, then the information content of rolling the die would be further low as given below:

$$\mathrm{I} = -0.75 \log_2 0.75 - \sum_{2}^{6} 0.05 \log_2 0.05 = 1.75$$

It is obvious that as the certainty of an event goes up, the total information goes down.

Information plays a crucial role in selecting the root and internal nodes for building a decision tree; in other words, it plays an essential role in choosing a **split variable**. The split variable is a variable that reduces the uncertainty by the most significant amount. Ideally, each variable value should provide us with objects that belong to only one class and therefore have zero information.

Information Gain (I.G): Specifies the amount of information that is gained by knowing the value of the variable. It measures the “goodness” of an input variable for predicting the output variable. The variable with the highest information gain is selected as the following split variable. Mathematically, it is defined as the entropy of the distribution before the split minus the entropy of the distribution after the split.

I.G = (Entropy of distribution before the split) – (Entropy of distribution after the split)

The most significant information gain is equivalent to the smallest entropy or minimum information. It means that if the result of an event is certain, i.e., the probability of an event is 1, then its information is zero while the information gain will be the largest; thus, it should be selected as a split variable.

Thus, after computing the information gain for every variable, the variable with the highest information gain is selected as a split variable (Tan, Steinbach & Kumar, 2006).

4.2.3.2. Split Criterion Based on the Gini Index:

The Gini Index is used to represent level of equality or inequality among objects. It can also be used to make decision trees. It was developed by Italian scientist **Corrado Gini** in (1912), and was used to analyze equality distribution of income among people. We will consider the example of wealth distribution in society in order to understand the concept of the Gini Index. The Gini Index always ranges between 0 and 1. It was designed to define the gap between the rich and the poor people, with 0 signifying perfect equality where all people have the same income, while 1 demonstrating perfect inequality where only one person gets everything in terms of income and rest of the others get nothing (Han, Pei, & Kamber, 2011).

From this, if Gini Index is very high, then it is evident there will be huge inequality in income distribution. Therefore, we will be interested in knowing person's income. But in a society where everyone has same income then no one will be interested in knowing each other's income because they know that everyone is at the same level. Thus, the variable of interest can be decided on the basis that if variable has a high value Gini Index, then it carriers more information and if it has a low value Gini Index then its information content is low.

To define the index, a graph is plotted by considering the percentage of income of the population as the Y-axis and the percentage of the population as the X-axis as shown in **Figure (4-2).**

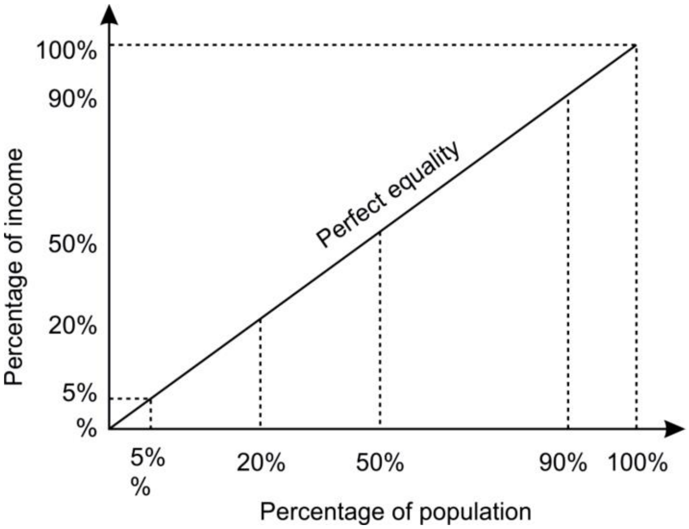


Figure (4-2) Gini Index representing perfect equality

In case of total equality in society, 5% of the people own 5% of the wealth, 20% of the people own 20% of the wealth similarly 90% of people own 90% of wealth. The line at 45-degrees thus represents perfect equal distribution of income in society.

The Gini Index is the ratio of the area between the **Lorenz curve** and the 45-degree line to the area under the 45-degree line given as follows:

$$\text{Gini Index (G)} = \frac{area\ between\ the\ Lorenz\ curve\ and\ the\ 45-degree\ line}{area\ under\ the\ 45-degree\ line}$$

As shown in **Figure (4-3)**, the area that lies between the line of equality and the **Lorenz curve** is

marked with *A* and the total area under the line of equality is represented by *(A + B)* in the **Figure (4-3)**. Therefore:

$$G = \frac{A}{A+B}$$

The most equal society will be the one in which every person has the same income, making $A = 0$, thus making $G = 0$.

However, the most unequal society will be the one in which a single person receives 100% of the total wealth thus making $B = 0$ and $G = A/A = 1$. From this, we can observe that G always lies between 0 and 1.

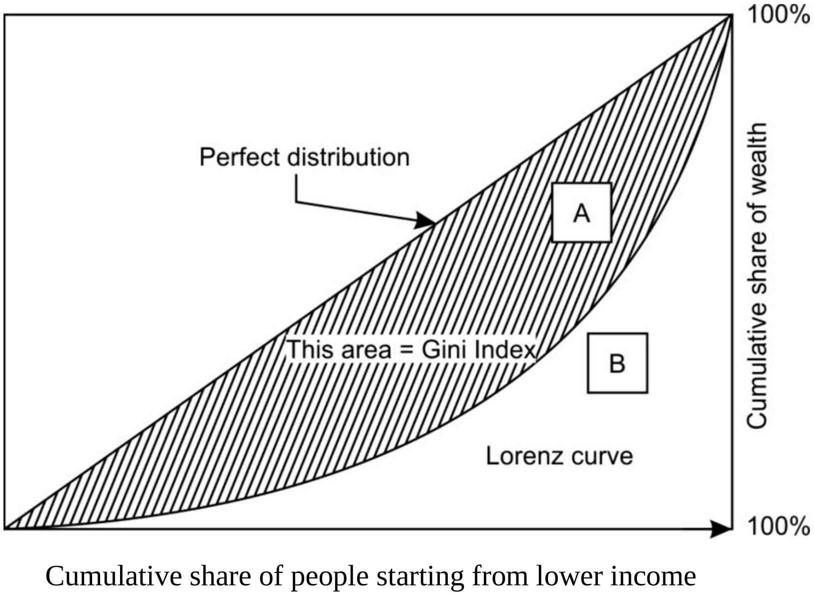

100%
Perfect distribution
A
This area = Gini Index
B
Lorenz curve
Cumulative share of wealth
100%
Cumulative share of people starting from lower income

Figure (4-3) Lorenz curve

From this, it can be concluded that, if income were distributed with perfect equality, the **Lorenz curve** would coincide with the 45-degree line and the Gini Index would be zero. However, if income were distributed with perfect inequality, the **Lorenz curve** would coincide with the horizontal axis and the right vertical axis and the index would be 1. (Tan, Steinbach & Kumar, 2006).

Gini Index can also be calculated in terms of probability and if a dataset *D* contains instances from k classes, the Gini Index, G(D), is defined as:

$$G(D) = 1 - \sum_{i=1}^{k} (P_i)^2$$

Here, P_i is the relative frequency or probability of class i in *D*.

Let us calculate the Gini Index for a dice with six possible outcomes with equal probability as:

$$G = 1 - 6\left(\frac{1}{6}\right)^2 = \frac{5}{6} = 0.833$$

If the dice is biased, let us say there is 50% chance of getting 6 and remaining 50% is being shared by other 5 numbers leaving only a 10% chance of getting each number other than 6 then the Gini Index is:

$$G = 1 - 5(0.1)^2 - (0.5)^2 = 0.7$$

And if the dice is further biased such that there is a 75% chance of getting a 6, then the Gini Index is:

$$G = 1 - 5(0.05)^2 - (0.75)^2 = 0.425$$

Here, Gini Index has been reduced from 0.833 to 0.70 to 0.425. Clearly, the high value of index means high uncertainty. (Bhatia, 2019).

It is important to observe that the Gini Index behaves in the same manner as the information gain discussed in Section (**4.2.3.1.). Table (4-1)** clearly shows that same trend in both the cases.

Table (4-1)

Event	Information Gain	Gini Index
Toss of unbiased coin	1	0.5
Toss of biased coin (60% Heads)	0.881	0.42
Throw of unbiased dice	2.585	0.83
Throw of biased dice (50% chance of a 6)	2.16	0.7

Now, after we had some knowledge about the **Attribute Selection Measures (ASM),** let us see how the decision tree algorithm works.

4.2.4. Decision Tree Algorithm:

The basic idea behind any decision tree algorithm is as follows:

1. Select the best variable using Attribute Selection Measures (ASM) to split the records.

2. Make that variable a decision node and breaks the training dataset into smaller subsets.

3. Start tree building by repeating this process recursively for each internal node until one of the conditions will match:

- ❖ All the tuples belong to the same output variable value (class).
- ❖ There are no more remaining variables could be used as splitting variable.
- ❖ There are no more observations in training dataset. (Shmueli, et al., 2017).

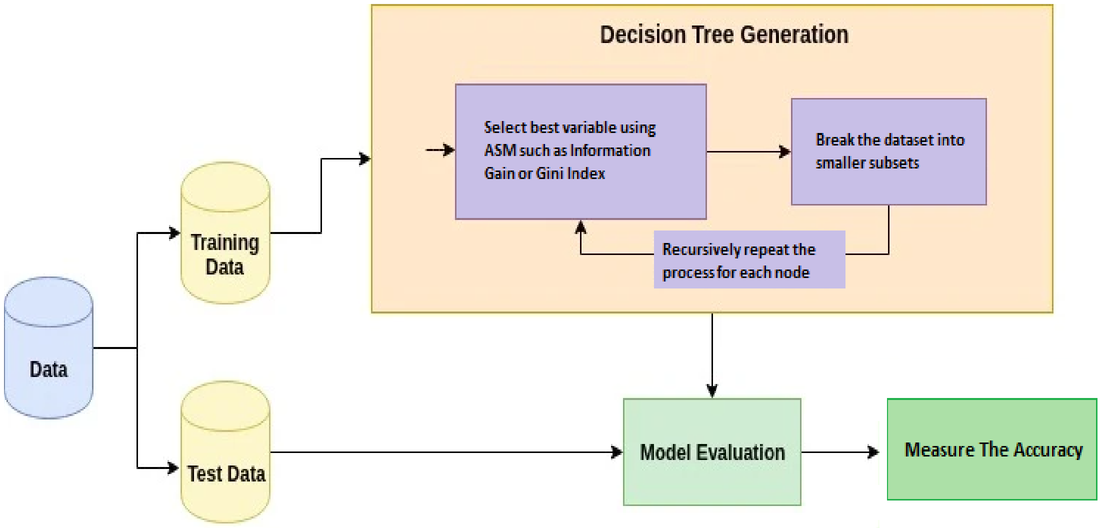

Decision Tree Generation
Select best variable using ASM such as Information Gain or Gini Index
Break the dataset into smaller subsets
Recursively repeat the process for each node
Training Data
Data
Test Data
Model Evaluation
Measure The Accuracy

Figure (4-4) Decision tree algorithm flowchart

An example explained in details in **Appendix [2].**

4.3. Building the Case Study Model:

Recall our case study to build a decision tree model to predict which team will win the match' home team, away team, or it will end with a draw. Using the league's name, home and away qualitative teams' attributes, and players' attributes for home and away teams.

Decision Tree Model: Winner ~ Leagues + Teams' Attributes + Players' Attributes.

4.3.1. Training the Model:

As mentioned before, to build any classification model, there are two steps; training and testing. First, the data will be partitioned into two parts by random, the first part with 80% of the dataset used to train the model, and the second part with the rest 20% of the dataset, used to test the model's accuracy.

A decision tree model has been trained with 34 variables and 14812 observations; by using the **rpart** package in **R**, results are as follows:

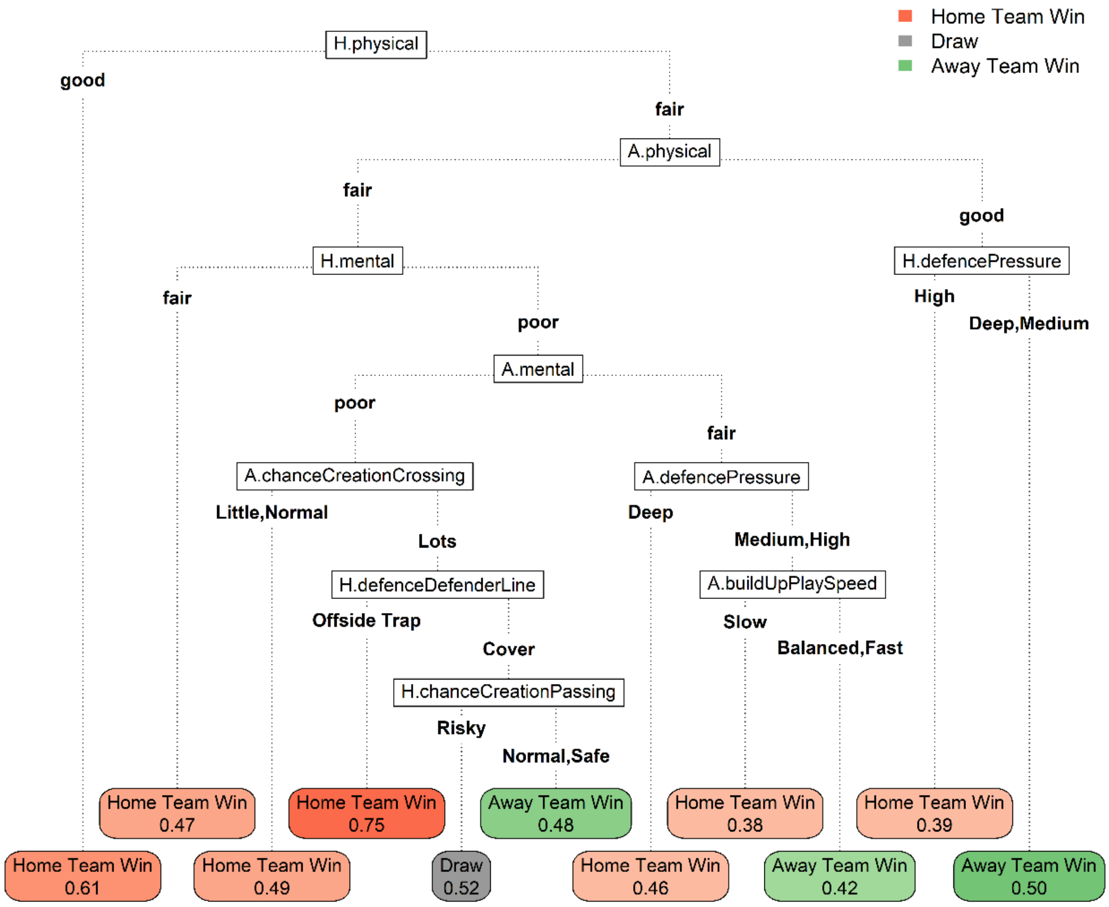


Figure (4-5) Case study decision tree model

The previous decision tree could be interpreted with some logical statements; each statement is obtained by passing the way from root to leaf. For example:

If the physical attributes of the players are “fair” for the home team and “good” for the away team, and the defense pressure attribute of the home team is “deep” or “medium”; then the away team will win the match.

While, if the physical attributes of the players are “fair” for both home and away teams, and the mental attributes of the players are “poor” for both home and away teams, and the chance creation crossing attribute of the away team is “little” or “normal”; then the home team will win the match. And so on for the rest branches.

After we have trained the model, we need to calculate its accuracy.

4.3.2. Testing the Model:

By using the rest 3704 observations of the dataset, model will be tested by using it to predict the winner of every match in the test data, and comparing the predicted values with the real values:

Table (4-2)

Predicted Vs Real	**Home Team Win**	**Draw**	**Away Team Win**
Home Team Win	1413	100	119
Draw	416	420	124
Away Team Win	447	60	625

And by using the following formula:

$$\boldsymbol{Accuracy = \frac{True\ Predictions}{True\ Predictions + False\ Predictions}}$$

The accuracy of decision tree model is 71.38%, which means that the model has successfully predicted 71.38% of the matches' results.

In the appendix [3], there are some codes represents the whole operations for building a decision tree model, from preparing to calculating accuracy.

Chapter 5

Chapter 5:

Conclusions and Recommendations

From the beginning this thesis aimed to introduce data science, by studying its steps in details to know the role of statistics in data science field, and to determine the differences between statistics and data science.

Now, after a data science project has been done, we can summaries the interactions with statistics in the data science process steps as following:

1. **Retrieving Data:** in traditional statistical studies, we used to deal with structured data, stored in excel sheet, SPSS file, or any tabular form. But in data science process one has to deal with any type of data, even unstructured data such as voice records, videos, images, text, etc.
2. **Preparing Data:** since one deal with unstructured huge data, he needs many preprocessing steps to prepare raw data, and transform it to tidy data which able to deal with any statistical analysis. This step is unusual for traditional statistician used to deal with structured data, but baring in mind that the statistician is more qualified than anyone to prepare data in suitable way, because he knows what he needs from it.
3. **Exploring Data:** this step is completely using statistical analysis (descriptive and inferential), but with a new vision. In traditional statistical studies we do not need this step, because the statistician knows what he is looking for in data, so he collects data in away guarantee that it will answers his questions, while data scientist using statistics to search for questions that can be answered with data.
4. **Setting Goals:** in traditional statistical studies this step comes before collecting data, statistician determine research questions and sets hypotheses, then he collects data to answer these questions and testing the hypotheses, while data scientist trying to discover questions in data then answering it.
5. **Data Modelling:** the main aim of any data science process is to build a predictive model; huge data requires new advanced modeling techniques, such as data mining techniques, statisticians do not used to deal with this kind of models, because they used to deal with

small samples of data, but we have to mention that all these advanced techniques are based on statistical principles.

6. **Automation:** the flowing of data requires models that evaluating automatically, here data scientist using his brilliant skills in coding to write a program that automate the previous steps to deal with new data.
7. **Presentation:** visualization techniques have been improved, so statisticians need to improve their tools to visualize data analysis in simplest way.

After this summary we can answer the question: **How to become a data scientist?!**

Statistician needs to improve his skills in programming languages, because data science requires advanced tools in programming, in addition, he needs flexibility to deal with any type of data, and to fix any problem, he needs to give up with his traditional methodology and accept that some times data comes without goals, he must be flexible to mine in data to find questions that data can answer, and sets the goals then reach them.

As faculty members in the Statistics Department, it was natural to think about how to prepare statistician students to be data scientists as much as they are data analysts; so, we recommend to add four advanced courses to undergraduate program in Statistics Department, assuming that students familiar with using R, and here are the courses and their syllables:

1. **Data Science Toolbox:**
   - Introducing the idea of big data storing, the different formats of storing data, the different data structures, the different sources, and how to deal with all of these using R.
   - Dealing with different databases, especially rational ones such as SQL.
   - Introducing the concepts of computational thinking and designing algorithms.
   - Get some knowledge about data science learning resources such as GitHub, Stack Overflow, and Kaggle; which are useful to communicate with data scientists around the world for exchanging knowledge, where it is always important to know how to get help through the learning process.
2. **Data Science Pre-Processing:**
   - Training students to preparing data to statistical analysis, by training them to deal with raw unstructured data and convert it to tidy data, and introducing the

convenient concepts and packages to do that, such as reshaping, cleaning and merging data.

- Teaching students how to use all what they have learned of statistical methods to explore data, discover patterns, and get insights about which models to use.

3. **Data Science Modeling (I):**
    - Introducing primary Data Mining techniques, such as:
        - Classification methods.
        - Association methods.
        - Clustering methods.
        - Outlier analysis methods.
4. **Data Science Modeling (II):**
    - Introducing Machine Learning and the concept of automation.
    - Dealing with advanced Data Mining methods, especially Neural Networks and Text Mining.
    - Introducing Artificial Intelligence techniques.
    - Ending with an applied case study of data science project.

Keep in mind, the field of data science is too broad to be covered by just four subjects, but it can be a good start for graduated students to build their career in data science.

Finally, in the end of this thesis, we can say that: ***Data Science is a statistical analysis of big data, using advanced tools, and a modern methodology.***

Appendix [1]

<u>The detailed information of the variables supplied by FIFA:</u>

Here are some definitions of tables' variables:

1	**Country**	
	Name	Name of country
2	**League**	
	Name	Name of league
3	**Team**	
	Team long name	Team long name
	Team short name	Team short name
4	**Team's Attributes**	
	date	Date of record attributes rates
	Build up play speed	The speed in which attacks are put together. Rated as slow, balanced, fast
	Build up play dribbling	The ability to carry the ball and past an opponent while being in control. Rated as little, normal, lots.
	Build up play passing	Affects passing distance & support from teammates. Rated as short, mixed, long.
	Build up play positioning	A team's freedom of movement in the 1st two thirds of the pitch. Rated as free form, organized.
	Chance creation passing	Amount of risk in pass decision and run support. Rated as risky, normal, safe.
	Chance creation crossing	The tendency / frequency of crosses into the box. Rated as little, normal, lots.
	Chance creation shooting	The tendency / frequency of shots taken. Rated as little, normal, lots.
	Chance creation positioning	A team's freedom of movement in the final third of the pitch. Rated as free form, organized.
	Defense pressure	Affects how high up the pitch the team will start pressuring. Rated as deep, medium, high.

	Defense aggression	Affect the team’s approach to tackling the ball possessor. Rated as contain, press, double.
	Defense team width	Affects how much the team will shift to the ball side. Rated as wide, normal, narrow.
	Defense defender line	Affects the shape and strategy of the defense. Rated as cover, offside trap.
	Note That: teams attributes existing in two ways, classes (Qualitative) as showed above, and as rates ranges from 0 to 100 (Quantitative).	
5	**Player**	
	Name	Player name
	Birthday	Player birthday
	Wight	Player wight
	Hight	Player hight
6	**Players’ Attributes**	
	date	Date of record attributes rates
	Preferred foot	Which foot that player prefer to play with, Right of Left.
	Attacking work rate	How a player participates in attacks. Rated as low, medium and high.
	Defensive work rate	How a player participates in defensive plays. Rated as low, medium and high.
	Overall rating	is the average of the key Player Attributes rates of a player within their Potential rate calculated based on their position and international reputation, The OVR rating of a player determines their general performance quality and their value
	Potential	Potential is how much a player can grow during your career mode save.
	Crossing	the accuracy and the quality of a player's crosses. (Skills)
	Finishing	the ability of a player to score (ability for finishing - How well they can finish an opportunity with a score). (Skills)
	Heading accuracy	player's accuracy when using the head to pass, shoot or clear the ball. (Skills)
	Short passing	a player's accuracy for the short passes. (Skills)
	Volleys	a player's ability for performing volleys. (Skills)
	Dribbling	a player's ability to carry the ball and past an opponent while being in control. (Skills)

	Curve	a player's ability to curve the ball when passing and shooting. (Skills)
	Free kick accuracy	a player's accuracy for taking the Free Kicks. (Skills)
	Long passing	a player's accuracy for the long and aerial passes. (Skills)
	Ball control	the ability of a player to control the ball on the pitch. (Skills)
	Acceleration	specifies how fast a player can reach their maximum sprint speed. (Physical)
	Sprint speed	the speed rate of a player's sprinting (running). (Physical)
	Agility	how quick and graceful a player is able to control the ball. (Physical)
	Reactions	the acting speed of a player in response to the situations happening around them. (Physical)
	Balance	the even distribution of enabling a player to remain upright and steady when running, carrying and controlling the ball. (Physical)
	Shot power	the strength of a player's shootings. (Skills)
	Jumping	a player's ability and quality for jumping from the surface for headers. (Physical)
	Stamina	a player's ability to sustain prolonged physical or mental effort in a match. (Physical)
	Strength	the quality or state of being physically strong of a player. (Physical)
	Long shots	a player's accuracy for the shots taking from long distances. (Skills)
	Aggression	determines the aggression level of a player on pushing, pulling and tackling. (Mental)
	Interceptions	a player's capability to intercept the ball - to catch the opposing team's passes. (Mental)
	Positioning	a player's ability to place themselves in the right position to receive/catch the ball, score goals or do a tactical move. (Mental)
	Vision	determines a player's mental awareness about his teammates' positioning, for passing the ball to them. (Mental)
	Penalties	a player's accuracy for the shots taking from the penalty kicks. (Skills)
	Marking	a player's capability to mark an opposition player or players to prevent them from taking control of the ball. (Mental)
	Standing tackle	the ability of performing standing tackle of a player. (Skills)
	Sliding tackle	the ability of performing sliding tackle of a player in a match. (Skills)
	Gk diving	a player's ability to dive as a goalkeeper.

	Gk handling	a player's ability to handle the ball and hold onto it using their hands as a goalkeeper.
	Gk kicking	a player's ability to kick the ball as a goalkeeper.
	Gk positioning	determines that how well a player is able to perform the positioning on the goal line as a goalkeeper.
	Gk reflexes	a player's ability and speed to react (reflex) for catching/saving the ball as a goalkeeper.
	Note That: players attributes could be rated as: very poor, poor, fair, good, very good, excellent. While could be categorized to: skills, mental, physical, and goal-keeper attributes.	
7	**Match**	
	Date	Match date
	Home team goals	Number of home team goals on a particular match
	Away team goals	Number of away team goals on a particular match
	Finally, keep in mind that all the tables linked together by id variables are not mentioned above.	

Appendix [2]

Example of decision tree process in details:

Here is an example to explain how to build a decision tree with Information Gain and Gini Index, with details.

Example 1. Let us consider an example and build a decision tree for the dataset given in **Table**. It has 4 independent (input) variables $\underline{X} = [X_1, X_2, X_3, X_4]$ which represent temperature, outlook, humidity and windy, respectively. Here, 'play' is the dependent (output) variable Y, with two classes (Yes, No), $k = 2$. the 14 records ($N = 14$) contain the information about weather conditions based on which it was decided if a play took place or not.

Temperature	Outlook	Humidity	Windy	Play
Hot	Sunny	High	False	No
Hot	Sunny	High	True	No
Hot	Overcast	High	False	Yes
Cool	Rain	Normal	False	Yes
Cool	Overcast	Normal	True	Yes
Mild	Sunny	High	False	No
Cool	Sunny	Normal	False	Yes
Mild	Rain	Normal	False	Yes
Mild	Sunny	Normal	True	Yes
Mild	Overcast	High	True	Yes
Hot	Overcast	Normal	False	Yes
Mild	Rain	High	True	No
Cool	Rain	Normal	True	No
Mild	Rain	High	False	Yes

Building a Decision Tree with Information Gain:

First of all, we need to calculate information of the whole dataset on the basis of whether (Play) is held or not:

$$I = -\sum_{i=1}^{k=2} P(y_i) \log_2 P(y_i)$$

Where: y_1 =Yes, y_2 =No, and $P(y_i) = \frac{n(y_i)}{N}$, $i = 1,2$. Then:

$$I = -\frac{n(y_1)}{N} \log_2 \frac{n(y_1)}{N} - \frac{n(y_2)}{N} \log_2 \frac{n(y_2)}{N}$$

$$I = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14}$$

$$I = 0.94029$$

Now, let us consider each variable one by one as split variable, and calculate the information for each variable.

- ***Temperature (X_1):***

As given in dataset there is three possible values for 'Temperature X_1': Hot X_{11}, Cool X_{12}, Mild X_{13}. And to calculate the information of X_1, first we count each value:

$$n(X_{11}) = 4\,, \qquad n(X_{12}) = 4, \qquad n(X_{13}) = 6$$

And:

$$I(X_{11}) = -\sum_{i=1}^{k=2} P(y_i|X_{11})\log_2 P(y_i|X_{11})$$

Where $P(y_i|X_{11}) = \frac{n(y_i|X_{11})}{n(X_{11})}$, $n(y_1|X_{11}) = 2$, $n(y_2|X_{11}) = 2$, then:

$$I(X_{11}) = -\frac{n(y_1|X_{11})}{n(X_{11})}\log_2\frac{n(y_1|X_{11})}{n(X_{11})} - \frac{n(y_2|X_{11})}{n(X_{11})}\log_2\frac{n(y_2|X_{11})}{n(X_{11})}$$

$$I(X_{11}) = -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4}$$

$$I(X_{11}) = 1$$

We repeat these steps for the other values of 'Temperature':

$$I(X_{12}) = -\sum_{i=1}^{k=2} P(y_i|X_{12})\log_2 P(y_i|X_{12})$$

Where $P(y_i|X_{12}) = \frac{n(y_i|X_{12})}{n(X_{12})}$, $n(y_1|X_{12}) = 3$, $n(y_2|X_{12}) = 1$, then:

$$I(X_{12}) = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4}$$

$$I(X_{12}) = 0.81129$$

$$I(X_{13}) = -\sum_{i=1}^{k=2} P(y_i|X_{13})\log_2 P(y_i|X_{13})$$

Where $P(y_i|X_{13}) = \frac{n(y_i|X_{13})}{n(X_{13})}$, $n(y_1|X_{13}) = 4$, $n(y_2|X_{13}) = 2$, then:

$$I(X_{13}) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6}$$

$$I(X_{13}) = 0.9183$$

The total information for X_1:

$$I(X_1) = P(X_{11}) * I(X_{11}) + P(X_{12}) * I(X_{12}) + P(X_{13}) * I(X_{13})$$

Where $P\left(X_{1j}\right) = \frac{n(X_{1j})}{N}$, $i = 1{,}2{,}3$. So:

$$I(X_1) = \frac{4}{14} * 1 + \frac{4}{14} * 0.81129 + \frac{6}{14} * 0.9183$$

$$I(X_1) = 0.911069$$

- ***Outlook (X_2):***

As given in dataset there is three possible values for 'Outlook X_2': Sunny X_{21}, Overcast X_{22}, Rain X_{23}. And to calculate the information of X_2, first we count each value:

$$n(X_{21}) = 5\,, \qquad n(X_{22}) = 4, \qquad n(X_{23}) = 5$$

And:

$$I(X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{21}) \log_2 P(y_i|X_{21})$$

Where $P(y_i|X_{21}) = \frac{n(y_i|X_{21})}{n(X_{21})}$, $n(y_1|X_{21}) = 2$, $n(y_2|X_{21}) = 3$, then:

$$I(X_{21}) = -\frac{n(y_1|X_{21})}{n(X_{21})}\log_2\frac{n(y_1|X_{21})}{n(X_{21})} - \frac{n(y_2|X_{21})}{n(X_{21})}\log_2\frac{n(y_2|X_{21})}{n(X_{21})}$$

$$I(X_{21}) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}$$

$$I(X_{21}) = 0.9709$$

We repeat these steps for the other values of 'Outlook':

$$I(X_{22}) = -\sum_{i=1}^{k=2} P(y_i|X_{22}) \log_2 P(y_i|X_{22})$$

Where $P(y_i|X_{22}) = \frac{n(y_i|X_{22})}{n(X_{22})}$, $n(y_1|X_{22}) = 4$, $n(y_2|X_{22}) = 0$, then:

$$I(X_{22}) = -1\log_2 1 - 0\log_2 0$$

$$I(X_{22}) = 0$$

$$I(X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{23}) \log_2 P(y_i|X_{23})$$

Where $P(y_i|X_{23}) = \frac{n(y_i|X_{23})}{n(X_{23})}$, $n(y_1|X_{23}) = 3$, $n(y_2|X_{23}) = 2$, then:

$$I(X_{23}) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}$$

$$I(X_{23}) = 0.9709$$

The total information for X_2:

$$I(X_2) = P(X_{21}) * I(X_{21}) + P(X_{22}) * I(X_{22}) + P(X_{23}) * I(X_{23})$$

Where $P(X_{2j}) = \frac{n(X_{2j})}{N}$, $i = 1,2,3$. So:

$$I(X_2) = \frac{5}{14} * 0.9709 + \frac{4}{14} * 0 + \frac{5}{14} * 0.9709$$

$$I(X_2) = 0.6935$$

- ***Humidity (X_3):***

As given in dataset there is two possible values for 'Humidity X_3': High X_{31}, Normal X_{32}.

And to calculate the information of X_3, first we count each value:

$$n(X_{31}) = 7\,, \qquad n(X_{32}) = 7$$

And:

$$I(X_{31}) = -\sum_{i=1}^{k=2} P(y_i|X_{31}) \log_2 P(y_i|X_{31})$$

Where $P(y_i|X_{31}) = \frac{n(y_i|X_{31})}{n(X_{31})}$, $n(y_1|X_{31}) = 3$, $n(y_2|X_{31}) = 4$, then:

$$I(X_{31}) = -\frac{n(y_1|X_{31})}{n(X_{31})} \log_2 \frac{n(y_1|X_{31})}{n(X_{31})} - \frac{n(y_2|X_{31})}{n(X_{31})} \log_2 \frac{n(y_2|X_{31})}{n(X_{31})}$$

$$I(X_{31}) = -\frac{3}{7} \log_2 \frac{3}{7} - \frac{4}{7} \log_2 \frac{4}{7}$$

$$I(X_{31}) = 0.9852$$

$$I(X_{32}) = -\sum_{i=1}^{k=2} P(y_i|X_{32}) \log_2 P(y_i|X_{32})$$

Where $P(y_i|X_{32}) = \frac{n(y_i|X_{32})}{n(X_{32})}$, $n(y_1|X_{32}) = 6$, $n(y_2|X_{32}) = 1$, then:

$$I(X_{32}) = -\frac{6}{7} \log_2 \frac{6}{7} - \frac{1}{7} \log_2 \frac{1}{7}$$

$$I(X_{32}) = 0.5917$$

The total information for X_3:

$$I(X_3) = P(X_{31}) * I(X_{31}) + P(X_{32}) * I(X_{32})$$

Where $P(X_{3j}) = \frac{n(X_{3j})}{N}$, $i = 1,2$. So:

$$I(X_3) = \frac{7}{14} * 0.9852 + \frac{7}{14} * 0.5917$$

$$I(X_3) = 0.78845$$

- ***Windy (X_4):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the information of X_4, first we count each value:

$$n(X_{41}) = 8\,, \qquad n(X_{42}) = 6$$

And:

$$I(X_{41}) = -\sum_{i=1}^{k=2} P(y_i|X_{41}) \log_2 P(y_i|X_{41})$$

Where $P(y_i|X_{41}) = \frac{n(y_i|X_{41})}{n(X_{41})}$, $n(y_1|X_{41}) = 6$, $n(y_2|X_{41}) = 2$, then:

$$I(X_{41}) = -\frac{n(y_1|X_{41})}{n(X_{41})}\log_2\frac{n(y_1|X_{41})}{n(X_{41})} - \frac{n(y_2|X_{41})}{n(X_{41})}\log_2\frac{n(y_2|X_{41})}{n(X_{41})}$$

$$I(X_{41}) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8}$$

$$I(X_{41}) = 0.81128$$

$$I(X_{42}) = -\sum_{i=1}^{k=2} P(y_i|X_{42}) \log_2 P(y_i|X_{42})$$

Where $P(y_i|X_{42}) = \frac{n(y_i|X_{42})}{n(X_{42})}$, $n(y_1|X_{42}) = 3$, $n(y_2|X_{42}) = 3$, then:

$$I(X_{42}) = -\frac{3}{6}\log_2\frac{3}{6} - \frac{3}{6}\log_2\frac{3}{6}$$

$$I(X_{42}) = 1$$

The total information for X_4:

$$I(X_4) = P(X_{41}) * I(X_{41}) + P(X_{42}) * I(X_{42})$$

Where $P\left(X_{4j}\right) = \frac{n(X_{4j})}{N}$, $i = 1{,}2$. So:

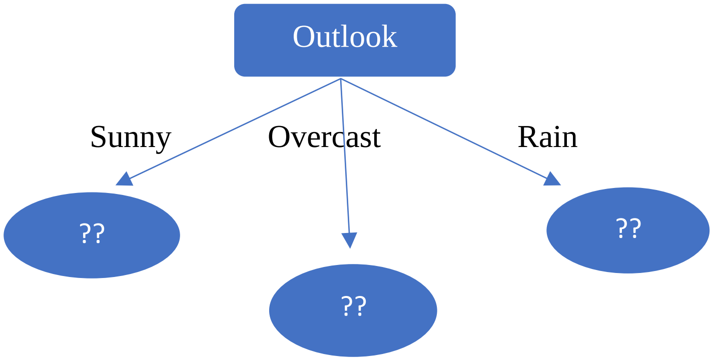
$$I(X_3) = \frac{8}{14} * 0.81128 + \frac{6}{14} * 1$$

$$I(X_4) = 0.89216$$

Now, the Information Gain can be computed:

Split Variable	Information Before splitting	Information After splitting	Information Gain
Temperature	0.94029	0.911069	0.029221
Outlook	0.94029	0.6935	0.24679
Humidity	0.94029	0.78845	0.15184
Windy	0.94029	0.8916	0.04813

From the above table, it is clear that the largest information gain is provided by the variable ‘Outlook’ so it is used as Root Node:



For Outlook X_2, as there are three possible values, Sunny X_{21}, Overcast X_{22}, Rain X_{23}, the dataset will be split into three subsets based on distinct values of the Outlook variable, as we show below:

Dataset for Outlook ‘Sunny’ ($X_2 = X_{21}$):

Temperature	Humidity	Windy	Play
Hot	High	False	No
Hot	High	True	No
Mild	High	False	No
Cool	Normal	False	Yes
Mild	Normal	True	Yes

In this case, the new dataset has three independent variables: Temperature X_1, Humidity X_3 and Windy X_4.

Again, we need first to calculate information of the whole dataset (when $X_2 = X_{21}$) on the basis of whether (Play) is held or not:

$$I = -\sum_{i=1}^{k=2} P(y_i)\log_2 P(y_i)$$

Where: y_1 =Yes, y_2 =No, and $P(y_i) = \frac{n(y_i)}{N}$, $i = 1,2$. Then:

$$I = -\frac{n(y_1)}{N}\log_2\frac{n(y_1)}{N} - \frac{n(y_2)}{N}\log_2\frac{n(y_2)}{N}$$

$$I = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}$$

$I = 0.9709$

Now, let us consider each variable one by one as split variable, and calculate the information for each variable.

- ***Temperature / Outlook 'Sunny' ($X_1|X_{21}$):***

As given in dataset there is three possible values for 'Temperature X_1': Hot X_{11}, Cool X_{12}, Mild X_{13}. And to calculate the information of X_1, first we count each value:

$$n(X_{11}|X_{21}) = 2\,, \qquad n(X_{12}|X_{21}) = 1, \qquad n(X_{13}|X_{21}) = 2$$

And:

$$I(X_{11}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{11}, X_{21}) \log_2 P(y_i|X_{11}, X_{21})$$

Where $P(y_i|X_{11}, X_{21}) = \frac{n(y_i|X_{11}, X_{21})}{n(X_{11}, X_{21})}$, $n(y_1|X_{11}, X_{21}) = 0$, $n(y_2|X_{11}, X_{21}) = 2$, then:

$$I(X_{11}|X_{21}) = -\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2}$$

$$I(X_{11}|X_{21}) = 0$$

$$I(X_{12}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{12}, X_{21}) \log_2 P(y_i|X_{12}, X_{21})$$

Where $P(y_i|X_{12}, X_{21}) = \frac{n(y_i|X_{12}, X_{21})}{n(X_{12}, X_{21})}$, $n(y_1|X_{12}, X_{21}) = 1$, $n(y_2|X_{12}, X_{21}) = 0$, then:

$$I(X_{12}|X_{21}) = -\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}$$

$$I(X_{12}|X_{21}) = 0$$

$$I(X_{13}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{13}, X_{21}) \log_2 P(y_i|X_{13}, X_{21})$$

Where $P(y_i|X_{13}, X_{21}) = \frac{n(y_i|X_{13}, X_{21})}{n(X_{13}, X_{21})}$, $n(y_1|X_{13}, X_{21}) = 1$, $n(y_2|X_{13}, X_{21}) = 1$, then:

$$I(X_{13}|X_{21}) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$I(X_{13}|X_{21}) = 1$$

The total information for $(X_1|X_{21})$:

$$I(X_1|X_{21}) = P(X_{11}|X_{21}) * I(X_{11}|X_{21}) + P(X_{12}|X_{21}) * I(X_{12}|X_{21}) + P(X_{13}|X_{21}) * I(X_{13}|X_{21})$$

Where $P\left(X_{1j}|X_{21}\right) = \frac{n(X_{1j}|X_{21})}{N}$, $i = 1{,}2{,}3$. So:

$$I(X_1) = \frac{2}{5} * 0 + \frac{1}{5} * 0 + \frac{2}{5} * 1$$

$$I(X_1|X_{21}) = 0.4$$

- ***Humidity / Outlook 'Sunny' ($X_3|X_{21}$):***

As given in dataset there is two possible values for 'Humidity X_3': High X_{31}, Normal X_{32}. And to calculate the information of X_3, first we count each value:

$$n(X_{31}|X_{21}) = 3\,, \qquad n(X_{32}|X_{21}) = 2$$

And:

$$I(X_{31}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{31}, X_{21}) \log_2 P(y_i|X_{31}, X_{21})$$

Where $P(y_i|X_{31}, X_{21}) = \frac{n(y_i|X_{31}, X_{21})}{n(X_{31}, X_{21})}$, $n(y_1|X_{31}, X_{21}) = 0$, $n(y_2|X_{31}, X_{21}) = 3$, then:

$$I(X_{31}|X_{21}) = -\frac{0}{3}\log_2\frac{0}{3} - \frac{3}{3}\log_2\frac{3}{3}$$

$$I(X_{31}|X_{21}) = 0$$

$$I(X_{32}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{32}, X_{21}) \log_2 P(y_i|X_{32}, X_{21})$$

Where $P(y_i|X_{32}, X_{21}) = \frac{n(y_i|X_{32}, X_{21})}{n(X_{32}, X_{21})}$, $n(y_1|X_{32}, X_{21}) = 2$, $n(y_2|X_{32}, X_{21}) = 0$, then:

$$I(X_{32}|X_{21}) = -\frac{2}{2}\log_2\frac{2}{2} - \frac{0}{2}\log_2\frac{0}{2}$$

$$I(X_{32}|X_{21}) = 0$$

The total information for $(X_3|X_{21})$:

$$I(X_3|X_{21}) = P(X_{31}|X_{21}) * I(X_{31}|X_{21}) + P(X_{32}|X_{21}) * I(X_{32}|X_{21})$$

Where $P(X_{3j}|X_{21}) = \frac{n(X_{3j}|X_{21})}{N}$, $i = 1{,}2$. So:

$$I(X_3|X_{21}) = \frac{3}{5} * 0 + \frac{2}{5} * 0$$

$$I(X_3|X_{21}) = 0$$

- ***Windy / Outlook 'Sunny' ($X_4|X_{21}$):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the information of X_4, first we count each value:

$$n(X_{41}|X_{21}) = 3\,, \qquad n(X_{42}|X_{21}) = 2$$

And:

$$I(X_{41}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{41}, X_{21}) \log_2 P(y_i|X_{41}, X_{21})$$

Where $P(y_i|X_{41}, X_{21}) = \frac{n(y_i|X_{41},X_{21})}{n(X_{41},X_{21})}$, $n(y_1|X_{41}, X_{21}) = 1$, $n(y_2|X_{41}, X_{21}) = 2$, then:

$$I(X_{41}|X_{21}) = -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$$

$$I(X_{41}|X_{21}) = 0.9183$$

$$I(X_{42}|X_{21}) = -\sum_{i=1}^{k=2} P(y_i|X_{42}, X_{21}) \log_2 P(y_i|X_{42}, X_{21})$$

Where $P(y_i|X_{42}, X_{21}) = \frac{n(y_i|X_{42},X_{21})}{n(X_{42},X_{21})}$, $n(y_1|X_{42}, X_{21}) = 1$, $n(y_2|X_{42}, X_{21}) = 1$, then:

$$I(X_{42}|X_{21}) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$I(X_{42}|X_{21}) = 1$$

The total information for $(X_4|X_{21})$:

$$I(X_4|X_{21}) = P(X_{41}|X_{21}) * I(X_{41}|X_{21}) + P(X_{42}|X_{21}) * I(X_{42}|X_{21})$$

Where $P\left(X_{4j}|X_{21}\right) = \frac{n(X_{4j}|X_{21})}{N}$, $i = 1{,}2$. So:

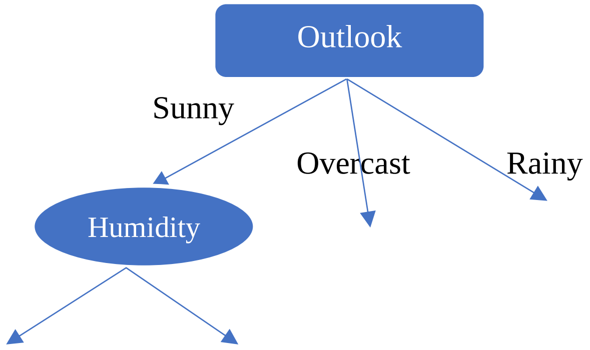
$$I(X_4|X_{21}) = \frac{3}{5} * 0.9183 + \frac{2}{5} * 1$$

$$I(X_4|X_{21}) = 0.95098$$

Now, the Information Gain can be computed:

Split Variable	Information Before splitting	Information After splitting	Information Gain
Temperature	0.9709	0.4	0.5709
Humidity	0.9709	0	0.9709
Windy	0.9709	0.9509	0.02

From the above table, it is clear that the largest information gain is provided by the variable ‘Humidity’ so it is used as the first Internal Node:



There are two possible values of humidity so data is split into two parts, i.e., humidity 'high' and humidity 'Normal' as shown below:

Dataset for Humidity 'High' / Overcast 'Sunny' ($X_3 = X_{31} | X_2 = X_{21}$):

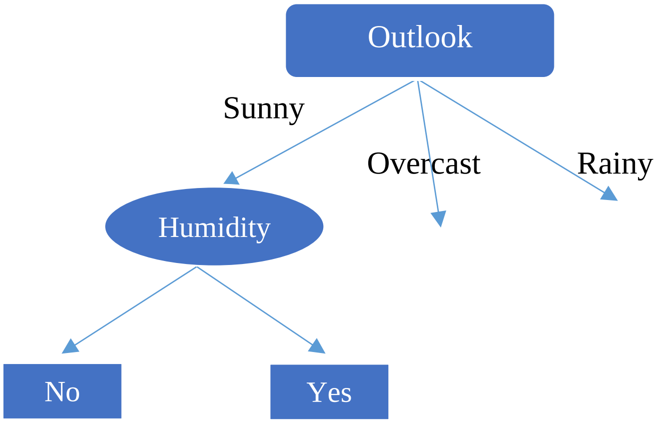
Temperature	Windy	Play
Hot	False	No
Hot	True	No
Mild	False	No

For this dataset, we have two independent variables Temperature X_1 and Windy X_4. As the dataset represents the same class 'No' for all the records, therefore for Humidity value 'High' ($X_3 = X_{31}$), the dependent variable is always 'No', $Y = y_2$.

Dataset for Humidity 'Normal' / Overcast 'Sunny' ($X_3 = X_{32} | X_2 = X_{21}$):

Temperature	Windy	Play
Cool	False	Yes
Mild	True	Yes

When Humidity's value is 'Normal', ($X_3 = X_{32}$), the output class is always 'Yes', in other words the dependent variable is always 'Yes', $Y = y_1$. Thus, decision tree will look like as:

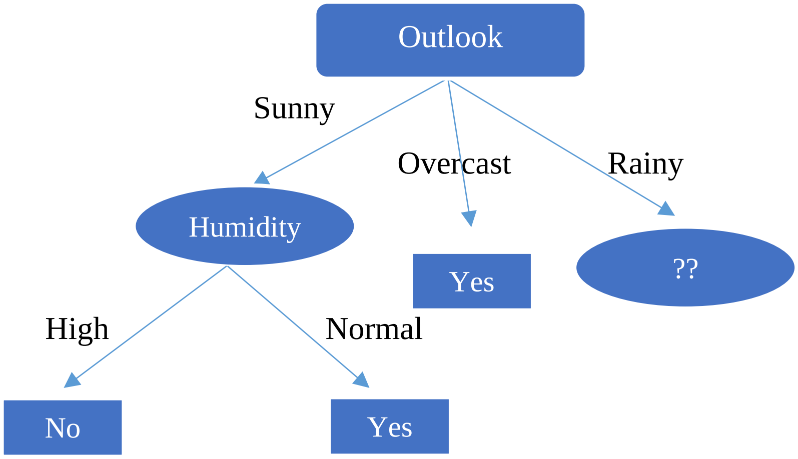


Now, the analysis of Sunny dataset is over. Let us take next subset which has Outlook as 'Overcast' for further analysis.

Dataset for Outlook 'Overcast' ($X_2 = X_{22}$):

Temperature	Humidity	Windy	Play
Hot	High	False	yes
Cool	Normal	True	Yes
Mild	High	True	Yes
Hot	Normal	False	yes

For outlook Overcast, the output class is always 'Yes'. in other words, the dependent variable is always 'Yes', $Y = y_1$. Thus, decision tree will look like as:



Now, we have to select another split variable for the outlook rainy and subset of the dataset for this is given below.

Temperature	Humidity	Windy	Play
Cool	Normal	False	Yes
Mild	Normal	False	Yes
Mild	High	True	No
Cool	Normal	True	No
Mild	High	False	Yes

In this case, the new dataset has three independent variables: Temperature X_1, Humidity X_3 and Windy X_4.

Again, we need first to calculate information of the whole dataset (when $X_2 = X_{23}$) on the basis of whether (Play) is held or not:

$$I = -\sum_{i=1}^{k=2} P(y_i) \log_2 P(y_i)$$

Where: y_1 =Yes, y_2 =No, and $P(y_i) = \frac{n(y_i)}{N}$, $i = 1,2$. Then:

$$I = -\frac{n(y_1)}{N}\log_2\frac{n(y_1)}{N} - \frac{n(y_2)}{N}\log_2\frac{n(y_2)}{N}$$

$$I = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}$$

$$I = 0.9709$$

Now, let us consider each variable one by one as split variable, and calculate the information for each variable.

- ***Temperature / Outlook 'Rain' ($X_1|X_{23}$):***

  As given in dataset there is three possible values for 'Temperature X_1': Cool X_{12}, Mild X_{13}. And to calculate the information of X_1, first we count each value:

  $$n(X_{12}|X_{23}) = 2, \qquad n(X_{13}|X_{23}) = 3$$

  And:

  $$I(X_{12}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{12}, X_{23}) \log_2 P(y_i|X_{12}, X_{23})$$

  Where $P(y_i|X_{12}, X_{23}) = \frac{n(y_i|X_{12}, X_{23})}{n(X_{12}, X_{23})}$, $n(y_1|X_{12}, X_{23}) = 1$, $n(y_2|X_{12}, X_{23}) = 1$, then:

  $$I(X_{12}|X_{23}) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$I(X_{12}|X_{23}) = 1$

$$I(X_{13}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{13}, X_{23}) \log_2 P(y_i|X_{13}, X_{23})$$

Where $P(y_i|X_{13}, X_{23}) = \frac{n(y_i|X_{13}, X_{23})}{n(X_{13}, X_{23})}$, $n(y_1|X_{13}, X_{23}) = 2$, $n(y_2|X_{13}, X_{21}) = 1$, then:

$$I(X_{13}|X_{23}) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}$$

$I(X_{13}|X_{23}) = 0.9183$

The total information for $(X_1|X_{23})$:

$$I(X_1|X_{23}) = P(X_{12}|X_{23}) * I(X_{12}|X_{23}) + P(X_{13}|X_{23}) * I(X_{13}|X_{23})$$

Where $P(X_{1j}|X_{23}) = \frac{n(X_{1j}|X_{23})}{N}$, $i = 2{,}3$. So:

$$I(X_1) = \frac{2}{5} * 1 + \frac{3}{5} * 0.9183$$

$$I(X_1|X_{23}) = 0.95098$$

- ***Humidity / Outlook 'Rain' $(X_3|X_{23})$:***

As given in dataset there is two possible values for 'Humidity X_3': High X_{31}, Normal X_{32}.

And to calculate the information of X_3, first we count each value:

$$n(X_{31}|X_{23}) = 2\,, \qquad n(X_{32}|X_{21}) = 3$$

And:

$$I(X_{31}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{31}, X_{23}) \log_2 P(y_i|X_{31}, X_{23})$$

Where $P(y_i|X_{31}, X_{23}) = \frac{n(y_i|X_{31}, X_{23})}{n(X_{31}, X_{23})}$, $n(y_1|X_{31}, X_{23}) = 1$, $n(y_2|X_{31}, X_{23}) = 1$, then:

$$I(X_{31}|X_{23}) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$I(X_{31}|X_{23}) = 1$

$$I(X_{32}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{32}, X_{23}) \log_2 P(y_i|X_{32}, X_{23})$$

Where $P(y_i|X_{32}, X_{23}) = \frac{n(y_i|X_{32}, X_{23})}{n(X_{32}, X_{23})}$, $n(y_1|X_{32}, X_{23}) = 2$, $n(y_2|X_{32}, X_{23}) = 1$, then:

$$I(X_{32}|X_{23}) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}$$

$I(X_{32}|X_{23}) = 0.9183$

The total information for $(X_3|X_{23})$:

$$I(X_3|X_{23}) = P(X_{31}|X_{23}) * I(X_{31}|X_{23}) + P(X_{32}|X_{23}) * I(X_{32}|X_{23})$$

Where $P\left(X_{3j}|X_{23}\right) = \frac{n(X_{3j}|X_{23})}{N}$, $i = 1,2$. So:

$$I(X_3) = \frac{2}{5} * 1 + \frac{3}{5} * 0.9183$$

$$I(X_3|X_{23}) = 0.95098$$

- ***Windy / Outlook 'Rain' ($X_4|X_{23}$):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the information of X_4, first we count each value:

$$n(X_{41}|X_{23}) = 3\,, \qquad n(X_{42}|X_{23}) = 0$$

And:

$$I(X_{41}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{41}, X_{23}) \log_2 P(y_i|X_{41}, X_{23})$$

Where $P(y_i|X_{41}, X_{23}) = \frac{n(y_i|X_{41}, X_{23})}{n(X_{41}, X_{23})}$, $n(y_1|X_{41}, X_{23}) = 3$, $n(y_2|X_{41}, X_{23}) = 0$, then:

$$I(X_{41}|X_{23}) = -\frac{3}{3}\log_2\frac{3}{3} - \frac{0}{3}\log_2\frac{0}{3}$$

$$I(X_{41}|X_{23}) = 0$$

$$I(X_{42}|X_{23}) = -\sum_{i=1}^{k=2} P(y_i|X_{42}, X_{23}) \log_2 P(y_i|X_{42}, X_{23})$$

Where $P(y_i|X_{42}, X_{23}) = \frac{n(y_i|X_{42}, X_{23})}{n(X_{42}, X_{23})}$, $n(y_1|X_{42}, X_{23}) = 0$, $n(y_2|X_{42}, X_{23}) = 2$, then:

$$I(X_{42}|X_{23}) = -\frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2\frac{2}{2}$$

$$I(X_{42}|X_{23}) = 0$$

The total information for $(X_4|X_{23})$:

$$I(X_4|X_{23}) = P(X_{41}|X_{23}) * I(X_{41}|X_{23}) + P(X_{42}|X_{23}) * I(X_{42}|X_{23})$$

Where $P\left(X_{4j}|X_{23}\right) = \frac{n(X_{4j}|X_{23})}{N}$, $i = 1,2$. So:

$$I(X_3) = \frac{2}{5} * 0 + \frac{3}{5} * 0$$

$$I(X_4|X_{23}) = 0$$

Now, the Information Gain can be computed:

Split Variable	Information Before splitting	Information After splitting	Information Gain
Temperature	0.9709	0.95098	0.01992
Humidity	0.9709	0.95098	0.01992
Windy	0.9709	0	0.9709

From the above table, it is clear that the largest information gain is provided by the variable 'Windy' so it is used as the last Internal Node.

Dataset for Windy 'False' / Outlook 'Rain' ($X_4 = X_{41} | X_2 = X_{23}$):

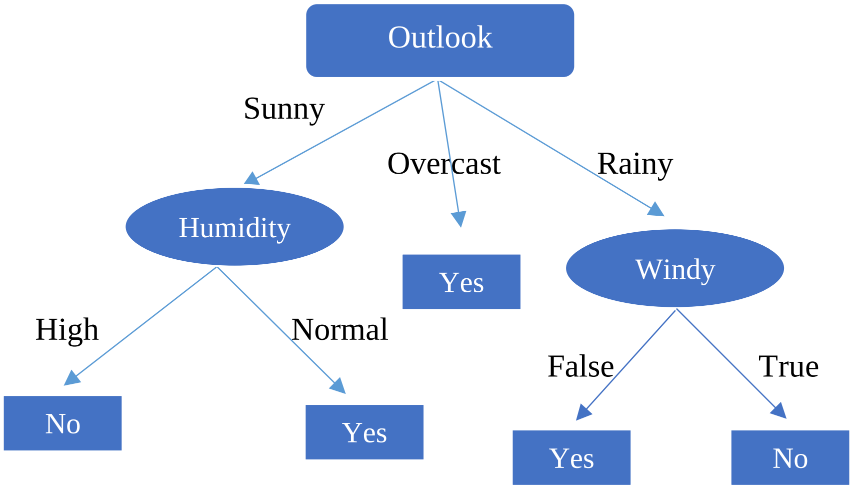
Temperature	Humidity	Play
Cool	Normal	Yes
Mild	Normal	Yes
Mild	High	yes

When Windy's value is 'False', ($X_4 = X_{41}$), the output class is always 'Yes', in other words the dependent variable is always 'Yes', $Y = y_1$.

Dataset for Windy 'True' / Outlook 'Rain' ($X_4 = X_{42} | X_2 = X_{23}$):

Temperature	Humidity	Play
Mild	High	No
Cool	Normal	No

When Windy's value is 'True', ($X_4 = X_{42}$), the output class is always 'No', in other words the dependent variable is always 'No', $Y = y_2$. Thus, decision tree will look like as:



Building a Decision Tree with Gini Index:

First of all, we need to calculate Gini Index of the whole dataset on the basis of whether (Play) is held or not:

$$G = 1 - \sum_{i=1}^{k=2} (P(y_i))^k$$

Where: y_1 =Yes, y_2 =No, and $P(y_i) = \frac{n(y_i)}{N}$, $i = 1,2$. While $n(y_1) = 9$, and $n(y_2) = 5$, Then:

$$G = 1 - \left(\frac{n(y_1)}{N}\right)^2 - \left(\frac{n(y_2)}{N}\right)^2$$

$$G = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2$$

$$G = 0.4592$$

Let us consider each variable one by one as split variables and calculate the Gini Index for each variable.

- ***Temperature (X_1):***

As given in dataset there is three possible values for 'Temperature X_1': Hot X_{11}, Cool X_{12}, Mild X_{13}. And to calculate the Gini Index of X_1, first we count each value:

$$n(X_{11}) = 4 \,, \qquad n(X_{12}) = 4, \qquad n(X_{13}) = 6$$

$$n(y_1|X_{11}) = 2, \qquad n(y_1|X_{12}) = 3, \qquad n(y_1|X_{13}) = 4$$

$$n(y_2|X_{11}) = 2, \qquad n(y_2|X_{12}) = 1, \qquad n(y_2|X_{13}) = 2$$

And:

$$G(X_{11}) = 1 - \left(\frac{n(y_1|X_{11})}{n(X_{11})}\right)^2 - \left(\frac{n(y_2|X_{11})}{n(X_{11})}\right)^2$$

$$G(X_{11}) = 1 - \left(\frac{2}{4}\right)^2 - \left(\frac{2}{4}\right)^2$$

$$G(X_{11}) = 0.5$$

$$G(X_{12}) = 1 - \left(\frac{n(y_1|X_{12})}{n(X_{12})}\right)^2 - \left(\frac{n(y_2|X_{12})}{n(X_{12})}\right)^2$$

$$G(X_{12}) = 1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^2$$

$$G(X_{12}) = 0.375$$

$$G(X_{13}) = 1 - \left(\frac{n(y_1|X_{13})}{n(X_{13})}\right)^2 - \left(\frac{n(y_2|X_{13})}{n(X_{13})}\right)^2$$

$$G(X_{13}) = 1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2$$

$$G(X_{13}) = 0.4444$$

The total Gini Index for 'Temperature' is:

$$G(X_1) = P(X_{11}) * G(X_{11}) + P(X_{12}) * G(X_{12}) + P(X_{13}) * G(X_{13})$$

$$G(X_1) = \frac{4}{14} * 0.5 + \frac{4}{14} * 0.375 + \frac{6}{14} * 0.4444$$

$$G(X_1) = 0.4405$$

- ***Outlook (X_2):***

As given in dataset there is three possible values for 'Outlook X_2': Sunny X_{21}, Overcast X_{22}, Rain X_{23}. And to calculate the Gini Index of X_2, first we count each value:

$n(X_{21}) = 5$,	$n(X_{22}) = 4,$	$n(X_{23}) = 5$
$n(y_1|X_{21}) = 2,$	$n(y_1|X_{22}) = 4,$	$n(y_1|X_{23}) = 3$
$n(y_2|X_{21}) = 3,$	$n(y_2|X_{22}) = 0,$	$n(y_2|X_{23}) = 2$

And:

$$G(X_{21}) = 1 - \left(\frac{n(y_1|X_{21})}{n(X_{21})}\right)^2 - \left(\frac{n(y_2|X_{21})}{n(X_{21})}\right)^2$$

$$G(X_{21}) = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2$$

$$G(X_{21}) = 0.48$$

$$G(X_{22}) = 1 - \left(\frac{n(y_1|X_{22})}{n(X_{22})}\right)^2 - \left(\frac{n(y_2|X_{22})}{n(X_{22})}\right)^2$$

$$G(X_{22}) = 1 - \left(\frac{4}{4}\right)^2 - \left(\frac{0}{4}\right)^2$$

$$G(X_{22}) = 0$$

$$G(X_{23}) = 1 - \left(\frac{n(y_1|X_{23})}{n(X_{23})}\right)^2 - \left(\frac{n(y_2|X_{23})}{n(X_{23})}\right)^2$$

$$G(X_{23}) = 1 - \left(\frac{3}{5}\right)^2 - \left(\frac{2}{5}\right)^2$$

$$G(X_{13}) = 0.48$$

The total Gini Index for 'Outlook' is:

$G(X_2) = P(X_{21}) * G(X_{21}) + P(X_{22}) * G(X_{22}) + P(X_{23}) * G(X_{23})$

$G(X_2) = \frac{5}{14} * 0.48 + \frac{4}{14} * 0 + \frac{5}{14} * 0.48$

$G(X_2) = 0.343$

- ***Humidity (X_3):***

As given in dataset there is two possible values for 'Humidity X_3': High X_{31}, Normal X_{32}. And to calculate the Gini Index of X_3, first we count each value:

$n(X_{31}) = 7\,, \qquad n(X_{32}) = 7$

$n(y_1|X_{31}) = 4, \qquad n(y_1|X_{32}) = 6$

$n(y_2|X_{31}) = 3, \qquad n(y_2|X_{32}) = 1$

And:

$$G(X_{31}) = 1 - \left(\frac{n(y_1|X_{31})}{n(X_{31})}\right)^2 - \left(\frac{n(y_2|X_{31})}{n(X_{31})}\right)^2$$

$$G(X_{31}) = 1 - \left(\frac{4}{7}\right)^2 - \left(\frac{3}{7}\right)^2$$

$$G(X_{31}) = 0.4898$$

$$G(X_{32}) = 1 - \left(\frac{n(y_1|X_{32})}{n(X_{32})}\right)^2 - \left(\frac{n(y_2|X_{32})}{n(X_{32})}\right)^2$$

$$G(X_{32}) = 1 - \left(\frac{6}{7}\right)^2 - \left(\frac{1}{7}\right)^2$$

$$G(X_{32}) = 0.2449$$

The total Gini Index for 'Humidity' is:

$G(X_3) = P(X_{31}) * G(X_{31}) + P(X_{32}) * G(X_{32})$

$G(X_3) = \frac{7}{14} * 0.4898 + \frac{7}{14} * 0.2449$

$G(X_3) = 0.36735$

- ***Windy (X_4):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the Gini Index of X_4, first we count each value:

$n(X_{41}) = 8\,, \qquad n(X_{42}) = 6$

$n(y_1|X_{41}) = 6, \qquad n(y_1|X_{42}) = 3$

$$n(y_2|X_{41}) = 2, \qquad n(y_2|X_{42}) = 3$$

And:

$$G(X_{41}) = 1 - \left(\frac{n(y_1|X_{41})}{n(X_{41})}\right)^2 - \left(\frac{n(y_2|X_{41})}{n(X_{41})}\right)^2$$

$$G(X_{41}) = 1 - \left(\frac{6}{8}\right)^2 - \left(\frac{2}{8}\right)^2$$

$$G(X_{31}) = 0.375$$

$$G(X_{42}) = 1 - \left(\frac{n(y_1|X_{42})}{n(X_{42})}\right)^2 - \left(\frac{n(y_2|X_{42})}{n(X_{42})}\right)^2$$

$$G(X_{42}) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2$$

$$G(X_{42}) = 0.5$$

The total Gini Index for ‘Windy’ is:

$$G(X_4) = P(X_{41}) * G(X_{41}) + P(X_{42}) * G(X_{42})$$

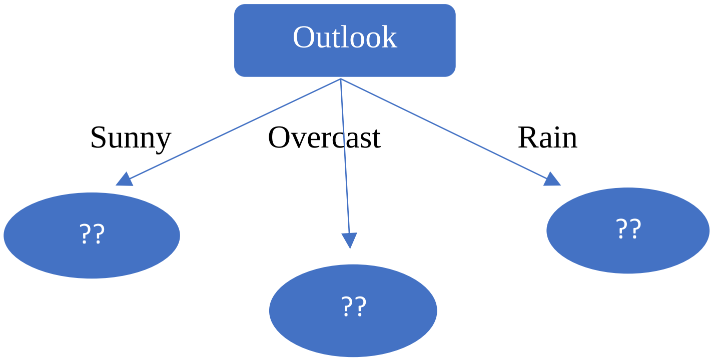
$$G(X_4) = \frac{8}{14} * 0.375 + \frac{6}{14} * 0.5$$

$$G(X_4) = 0.4286$$

Now, the Gini Index can be computed:

Split Variable	Gini Index Before splitting	Gini Index After splitting	Gini Index
Temperature	0.4592	0.4405	0.0187
Outlook	0.4592	0.343	0.1162
Humidity	0.4592	0.3674	0.0919
Windy	0.4592	0.4286	0.0306

From the above table, it is clear that the largest Gini Index is provided by the variable ‘Outlook’ so it is used as Root Node:



For Outlook X_2, as there are three possible values, Sunny X_{21}, Overcast X_{22}, Rain X_{23}, the dataset will be split into three subsets based on distinct values of the Outlook variable, as we show below:

Dataset for Outlook 'Sunny' ($X_2 = X_{21}$):

Temperature	Humidity	Windy	Play
Hot	High	False	No
Hot	High	True	No
Mild	High	False	No
Cool	Normal	False	Yes
Mild	Normal	True	Yes

In this case, the new dataset has 5 records ($N = 5$), and three independent variables: Temperature X_1, Humidity X_3 and Windy X_4.

Again, we need first to calculate Gini Index of the whole dataset when 'Outlook' is 'Sunny' ($X_2 = X_{21}$) on the basis of whether (Play) is held or not:

While $n(y_1) = 2$, and $n(y_2) = 3$, Then:

$$G = 1 - \left(\frac{n(y_1)}{N}\right)^2 - \left(\frac{n(y_2)}{N}\right)^2$$

$$G = 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2$$

$$G = 0.48$$

Let us consider each variable one by one as split variables and calculate the Gini Index for each variable.

- ***Temperature / Outlook 'sunny' ($X_1|X_{21}$):***

  As given in dataset there is three possible values for 'Temperature X_1': Hot X_{11}, Cool X_{12}, Mild X_{13}. And to calculate the Gini Index of X_1, first we count each value:

  $n(X_{11}|X_{21}) = 2\,,\qquad n(X_{12}|X_{21}) = 1,\qquad n(X_{13}|X_{21}) = 2$

  $n(y_1|X_{11}, X_{21}) = 0,\qquad n(y_1|X_{12}, X_{21}) = 1,\qquad n(y_1|X_{13}, X_{21}) = 1$

  $n(y_2|X_{11}, X_{21}) = 2,\qquad n(y_2|X_{12}, X_{21}) = 0,\qquad n(y_2|X_{13}, X_{21}) = 1$

  And:

$$G(X_{11}|X_{21}) = 1 - \left(\frac{n(y_1|X_{11}, X_{21})}{n(X_{11}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{11}, X_{21})}{n(X_{11}|X_{21})}\right)^2$$

$$G(X_{11}|X_{21}) = 1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2$$

$$G(X_{11}|X_{21}) = 0$$

$$G(X_{12}|X_{21}) = 1 - \left(\frac{n(y_1|X_{12}, X_{21})}{n(X_{12}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{12}, X_{21})}{n(X_{12}|X_{21})}\right)^2$$

$$G(X_{12}|X_{21}) = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2$$

$$G(X_{12}|X_{21}) = 0$$

$$G(X_{13}|X_{21}) = 1 - \left(\frac{n(y_1|X_{13}, X_{21})}{n(X_{13}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{13}, X_{21})}{n(X_{13}|X_{21})}\right)^2$$

$$G(X_{13}|X_{21}) = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2$$

$$G(X_{13}|X_{21}) = 0.5$$

The total Gini Index for ‘Temperature’ when ‘Outlook’ is ‘Sunny’:

$$G(X_1|X_{21}) = P(X_{11}|X_{21}) * G(X_{11}|X_{21}) + P(X_{12}|X_{21}) * G(X_{12}|X_{21}) + P(X_{13}|X_{21}) * G(X_{13}|X_{21})$$

$$G(X_1|X_{21}) = \frac{2}{5} * 0 + \frac{1}{5} * 0 + \frac{2}{5} * 0.5$$

$$G(X_1|X_{21}) = 0.2$$

- ***Humidity / Outlook ‘sunny’ ($X_3|X_{21}$):***

As given in dataset there is two possible values for ‘Humidity X_3’: High X_{31}, Normal X_{32}. And to calculate the Gini Index of X_3, first we count each value:

$n(X_{31}|X_{21}) = 3\,,$ $\quad n(X_{32}|X_{21}) = 2$

$n(y_1|X_{31}, X_{21}) = 0,$ $\quad n(y_1|X_{32}, X_{21}) = 2$

$n(y_2|X_{31}, X_{21}) = 3,$ $\quad n(y_2|X_{32}, X_{21}) = 0$

And:

$$G(X_{31}|X_{21}) = 1 - \left(\frac{n(y_1|X_{31}, X_{21})}{n(X_{31}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{31}, X_{21})}{n(X_{31}|X_{21})}\right)^2$$

$$G(X_{31}|X_{21}) = 1 - \left(\frac{0}{3}\right)^2 - \left(\frac{3}{3}\right)^2$$

$$G(X_{31}|X_{21}) = 0$$

$$G(X_{32}|X_{21}) = 1 - \left(\frac{n(y_1|X_{32}, X_{21})}{n(X_{32}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{32}, X_{21})}{n(X_{32}|X_{21})}\right)^2$$

$$G(X_{32}|X_{21}) = 1 - \left(\frac{2}{2}\right)^2 - \left(\frac{0}{2}\right)^2$$

$$G(X_{32}|X_{21}) = 0$$

The total Gini Index for 'Humidity' when 'Outlook' is 'Sunny':

$$G(X_3|X_{21}) = P(X_{31}|X_{21}) * G(X_{31}|X_{21}) + P(X_{32}|X_{21}) * G(X_{32}|X_{21})$$

$$G(X_3|X_{21}) = \frac{3}{5} * 0 + \frac{2}{5} * 0$$

$$G(X_3|X_{21}) = 0$$

- ***Windy / Outlook 'sunny' ($X_4|X_{21}$):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the Gini Index of X_4, first we count each value:

$n(X_{41}|X_{21}) = 4$, $\quad n(X_{42}|X_{21}) = 1$

$n(y_1|X_{41}, X_{21}) = 1$, $\quad n(y_1|X_{42}, X_{21}) = 1$

$n(y_2|X_{41}, X_{21}) = 3$, $\quad n(y_2|X_{42}, X_{21}) = 0$

And:

$$G(X_{41}|X_{21}) = 1 - \left(\frac{n(y_1|X_{41}, X_{21})}{n(X_{41}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{41}, X_{21})}{n(X_{41}|X_{21})}\right)^2$$

$$G(X_{41}|X_{21}) = 1 - \left(\frac{1}{4}\right)^2 - \left(\frac{3}{4}\right)^2$$

$$G(X_{41}|X_{21}) = 0.375$$

$$G(X_{42}|X_{21}) = 1 - \left(\frac{n(y_1|X_{42}, X_{21})}{n(X_{42}|X_{21})}\right)^2 - \left(\frac{n(y_2|X_{42}, X_{21})}{n(X_{42}|X_{21})}\right)^2$$

$$G(X_{42}|X_{21}) = 1 - \left(\frac{1}{1}\right)^2 - \left(\frac{0}{1}\right)^2$$

$$G(X_{42}|X_{21}) = 0$$

The total Gini Index for 'Humidity' when 'Outlook' is 'Sunny':

$$G(X_4|X_{21}) = P(X_{41}|X_{21}) * G(X_{41}|X_{21}) + P(X_{42}|X_{21}) * G(X_{42}|X_{21})$$

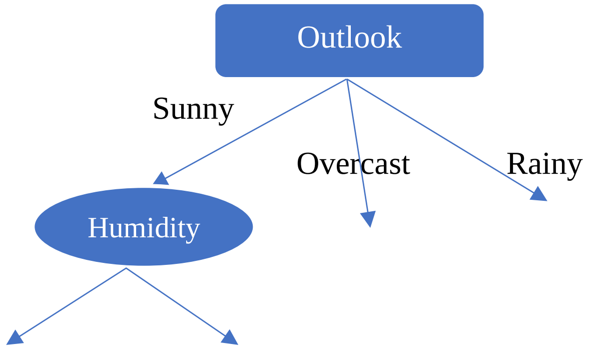
$$G(X_4|X_{21}) = \frac{4}{5} * 0.375 + \frac{1}{5} * 0$$

$G(X_4|X_{21}) = 0.3$

Now, the Gini Index can be computed:

Split Variable	Gini Index Before splitting	Gini Index After splitting	Gini Index
Temperature	0.48	0.2	0.28
Humidity	0.48	0	0.48
Windy	0.48	0.3	0.18

From the above table, it is clear that the largest Gini Index is provided by the variable 'Humidity' so it is used as the first Internal Node:



There are two possible values of humidity so data is split into two parts, i.e., humidity 'high' and humidity 'low' as shown below:

Dataset for Humidity 'High' ($X_3 = X_{31}$):

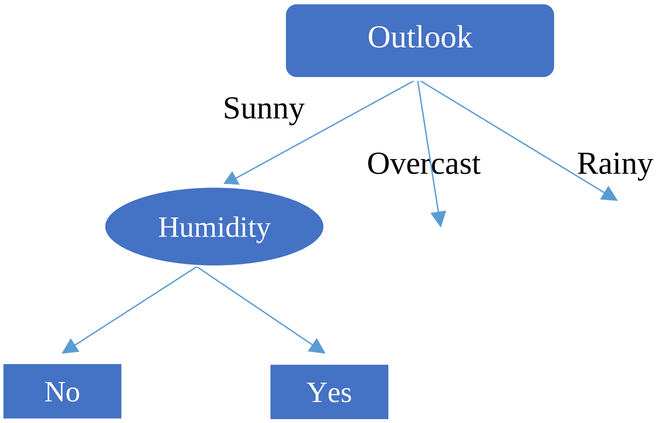
Temperature	Windy	Play
Hot	False	No
Hot	True	No
Mild	False	No

For this dataset, we have two independent variables Temperature X_1 and Windy X_4. As the dataset represents the same class 'No' for all the records, therefore for Humidity value 'High' ($X_3 = X_{31}$), the dependent variable is always 'No', $Y = y_2$.

Dataset for Humidity 'Normal' ($X_3 = X_{32}$):

Temperature	Windy	Play
Cool	False	Yes
Mild	True	Yes

When Humidity's value is 'Normal', ($X_3 = X_{32}$), the output class is always 'Yes', in other words the dependent variable is always 'Yes', $Y = y_1$. Thus, decision tree will look like as:

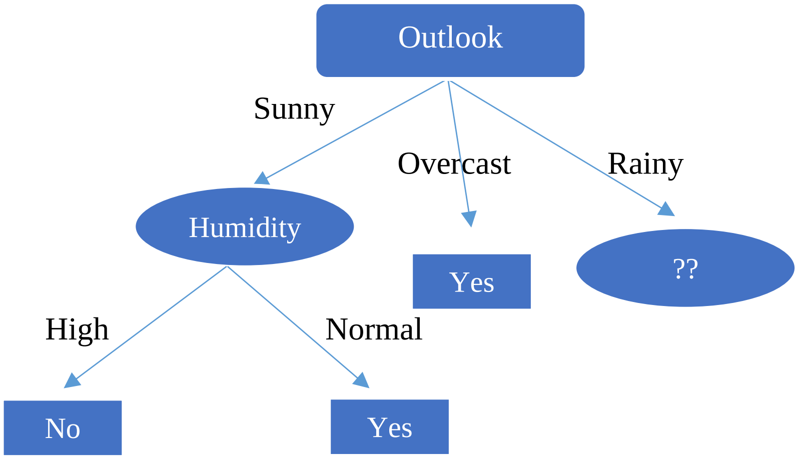


Now, the analysis of Sunny dataset is over. Let us take next subset which has Outlook as 'Overcast' for further analysis.

Dataset for Outlook 'Overcast' ($X_2 = X_{22}$):

Temperature	Humidity	Windy	Play
Hot	High	False	yes
Cool	Normal	True	Yes
Mild	High	True	Yes
Hot	Normal	False	yes

For outlook Overcast, the output class is always 'Yes'. in other words, the dependent variable is always 'Yes', $Y = y_1$. Thus, decision tree will look like as:



Now, we have to select another split variable for the outlook rainy and subset of the dataset for this is given below.

Temperature	Humidity	Windy	Play
Cool	Normal	False	Yes
Mild	Normal	False	Yes
Mild	High	True	No
Cool	Normal	True	No
Mild	High	False	Yes

In this case, the new dataset has 5 records ($N = 5$), and three independent variables:
Temperature X_1, Humidity X_3 and Windy X_4.

Again, we need first to calculate Gini Index of the whole dataset when ‘Outlook’ is ‘Rain’ ($X_2 = X_{23}$), on the basis of whether (Play) is held or not:

While $n(y_1) = 3$, and $n(y_2) = 2$, Then:

$$G = 1 - \left(\frac{n(y_1)}{N}\right)^2 - \left(\frac{n(y_2)}{N}\right)^2$$

$$G = 1 - \left(\frac{3}{5}\right)^2 - \left(\frac{2}{5}\right)^2$$

$$G = 0.48$$

Let us consider each variable one by one as split variables and calculate the Gini Index for each variable.

- ***Temperature / Outlook ‘Rain’ ($X_1|X_{23}$):***

As given in dataset there is two possible values for ‘Temperature X_1’: Cool X_{12}, Mild X_{13}.

And to calculate the Gini Index of X_1, first we count each value:

$$n(X_{12}|X_{23}) = 2, \qquad n(X_{13}|X_{23}) = 3$$

$$n(y_1|X_{12}, X_{23}) = 1, \qquad n(y_1|X_{13}, X_{23}) = 2$$

$$n(y_2|X_{12}, X_{23}) = 1, \qquad n(y_2|X_{13}, X_{23}) = 1$$

And:

$$G(X_{12}|X_{23}) = 1 - \left(\frac{n(y_1|X_{12}, X_{23})}{n(X_{12}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{12}, X_{23})}{n(X_{12}|X_{23})}\right)^2$$

$$G(X_{12}|X_{23}) = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2$$

$$G(X_{12}|X_{23}) = 0.5$$

$$G(X_{13}|X_{23}) = 1 - \left(\frac{n(y_1|X_{13}, X_{23})}{n(X_{13}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{13}, X_{23})}{n(X_{13}|X_{23})}\right)^2$$

$$G(X_{13}|X_{23}) = 1 - \left(\frac{2}{3}\right)^2 - \left(\frac{1}{3}\right)^2$$

$$G(X_{13}|X_{23}) = 0.4444$$

The total Gini Index for ‘Temperature’ when ‘Outlook’ is ‘Rain’:

$$G(X_1|X_{23}) = P(X_{12}|X_{23}) * G(X_{12}|X_{23}) + P(X_{13}|X_{23}) * G(X_{13}|X_{23})$$

$$G(X_1|X_{23}) = \frac{2}{5} * 0.5 + \frac{3}{5} * 0.4444$$

$$G(X_1|X_{23}) = 0.4666$$

- ***Humidity / Outlook ‘Rain’ ($X_3|X_{23}$):***

As given in dataset there is two possible values for ‘Humidity X_3’: High X_{31}, Normal X_{32}.

And to calculate the Gini Index of X_1, first we count each value:

$n(X_{31}|X_{23}) = 2,$ $n(X_{32}|X_{23}) = 3$

$n(y_1|X_{31}, X_{23}) = 1,$ $n(y_1|X_{32}, X_{23}) = 2$

$n(y_2|X_{31}, X_{23}) = 1,$ $n(y_2|X_{32}, X_{23}) = 1$

And:

$$G(X_{31}|X_{23}) = 1 - \left(\frac{n(y_1|X_{31}, X_{23})}{n(X_{31}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{31}, X_{23})}{n(X_{31}|X_{23})}\right)^2$$

$$G(X_{31}|X_{23}) = 1 - \left(\frac{1}{2}\right)^2 - \left(\frac{1}{2}\right)^2$$

$$G(X_{31}|X_{23}) = 0.5$$

$$G(X_{32}|X_{23}) = 1 - \left(\frac{n(y_1|X_{32}, X_{23})}{n(X_{32}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{32}, X_{23})}{n(X_{32}|X_{23})}\right)^2$$

$$G(X_{32}|X_{23}) = 1 - \left(\frac{2}{3}\right)^2 - \left(\frac{1}{3}\right)^2$$

$$G(X_{32}|X_{23}) = 0.4444$$

The total Gini Index for ‘Humidity’ when ‘Outlook’ is ‘Rain’:

$$G(X_3|X_{23}) = P(X_{31}|X_{23}) * G(X_{31}|X_{23}) + P(X_{32}|X_{23}) * G(X_{32}|X_{23})$$

$$G(X_3|X_{23}) = \frac{2}{5} * 0.5 + \frac{3}{5} * 0.4444$$

$$G(X_3|X_{23}) = 0.4666$$

- ***Windy / Outlook 'Rain' ($X_4|X_{23}$):***

As given in dataset there is two possible values for 'Windy X_4': False X_{41}, True X_{42}. And to calculate the Gini Index of X_4, first we count each value:

$$n(X_{41}|X_{23}) = 3, \qquad n(X_{42}|X_{23}) = 2$$
$$n(y_1|X_{41}, X_{23}) = 3, \qquad n(y_1|X_{42}, X_{23}) = 0$$
$$n(y_2|X_{41}, X_{23}) = 0, \qquad n(y_2|X_{42}, X_{23}) = 2$$

And:

$$G(X_{41}|X_{23}) = 1 - \left(\frac{n(y_1|X_{41}, X_{23})}{n(X_{41}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{41}, X_{23})}{n(X_{41}|X_{23})}\right)^2$$

$$G(X_{41}|X_{23}) = 1 - \left(\frac{0}{3}\right)^2 - \left(\frac{3}{3}\right)^2$$

$$G(X_{41}|X_{23}) = 0$$

$$G(X_{42}|X_{23}) = 1 - \left(\frac{n(y_1|X_{42}, X_{23})}{n(X_{42}|X_{23})}\right)^2 - \left(\frac{n(y_2|X_{42}, X_{23})}{n(X_{42}|X_{23})}\right)^2$$

$$G(X_{42}|X_{23}) = 1 - \left(\frac{2}{2}\right)^2 - \left(\frac{0}{0}\right)^2$$

$$G(X_{42}|X_{23}) = 0$$

The total Gini Index for 'Windy' when 'Outlook' is 'Rain':

$$G(X_4|X_{23}) = P(X_{41}|X_{23}) * G(X_{41}|X_{23}) + P(X_{42}|X_{23}) * G(X_{42}|X_{23})$$

$$G(X_4|X_{23}) = \frac{3}{5} * 0 + \frac{2}{5} * 0$$

$$G(X_4|X_{23}) = 0$$

Now, the Gini Index can be computed:

Split Variable	Gini Index Before splitting	Gini Index After splitting	Gini Index
Temperature	0.48	0.4666	0.0356
Humidity	0.48	0.4666	0.0356
Windy	0.48	0	0.48

From the above table, it is clear that the largest Gini Index is provided by the variable 'Windy' so it is used as the last Internal Node.

Dataset for Windy 'False' / Outlook 'Rain' ($X_4 = X_{41} | X_2 = X_{23}$):

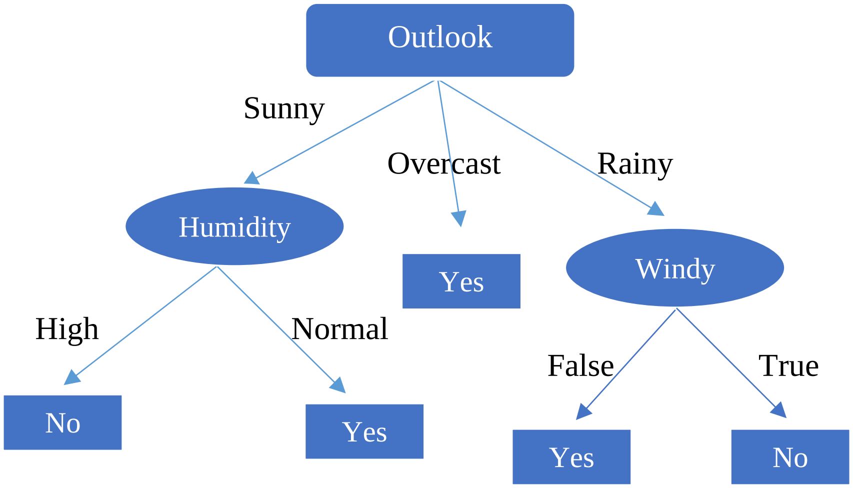
Temperature	Humidity	Play
Cool	Normal	Yes
Mild	Normal	Yes
Mild	High	yes

When Windy's value is 'False', ($X_4 = X_{41}$), the output class is always 'Yes', in other words the dependent variable is always 'Yes', $Y = y_1$.

Dataset for Windy 'True' / Outlook 'Rain' ($X_4 = X_{42} | X_2 = X_{23}$):

Temperature	Humidity	Play
Mild	High	No
Cool	Normal	No

When Windy's value is 'True', ($X_4 = X_{42}$), the output class is always 'No', in other words the dependent variable is always 'No', $Y = y_2$. Thus, decision tree will look like as:



Appendix [3]

<u>The codes of whole process of building the decision tree only:</u>

```
library(tidyr)
library(dplyr)
library(lubridate)
library(ggplot2)
library(rpart)
library(rpart.plot)
#First preparing the Match, Player's attributes, and Team's attributes tables:
#Teams
teams<-Team_Attributes[,-(1:2)]
teams[,2]<-as.Date(teams[,2])
teams<-mutate(teams,year=year(teams[,2]))


###Transforming class variables:
teams[,4]<-factor(teams[,4],levels = c("Slow", "Balanced", "Fast"))
teams[,6]<-factor(teams[,6],levels = c("Little", "Normal", "Lots"))
teams[,8]<-factor(teams[,8],levels = c("Long", "Mixed", "Short"))
teams[,9]<-factor(teams[,9],levels = c("Free Form", "Organised"))
teams[,11]<-factor(teams[,11],levels = c( "Risky", "Normal","Safe"))
teams[,13]<-factor(teams[,13],levels = c("Little", "Normal", "Lots"))
teams[,15]<-factor(teams[,15],levels = c("Little", "Normal", "Lots"))
teams[,16]<-factor(teams[,16],levels = c("Free Form", "Organised"))
teams[,18]<-factor(teams[,18],levels = c("Deep", "Medium","High"))
teams[,20]<-factor(teams[,20],levels = c("Double", "Press","Contain"))
teams[,22]<-factor(teams[,22],levels = c("Narrow", "Normal","Wide"))
teams[,23]<-factor(teams[,23],levels = c("Cover", "Offsid Trap"))
```

```
###Calculating means and medians over years:

teams<-group_by(teams,team_api_id,year) %>% summarise
(buildUpPlaySpeed=mean(buildUpPlaySpeed, na.rm = TRUE),

buildUpPlaySpeedClass= median(as.numeric(buildUpPlaySpeedClass), na.rm =
TRUE),

buildUpPlayDribbling= mean(buildUpPlayDribbling, na.rm = TRUE),

buildUpPlayDribblingClass= median(as.numeric(buildUpPlayDribblingClass),
na.rm = TRUE),

buildUpPlayPassing= mean(buildUpPlayPassing, na.rm = TRUE),

buildUpPlayPassingClass= median(as.numeric(buildUpPlayPassingClass), na.rm =
TRUE),

buildUpPlayPositioningClass= median(as.numeric(buildUpPlayPositioningClass),
na.rm = TRUE),

chanceCreationPassing= mean(chanceCreationPassing, na.rm = TRUE),

chanceCreationPassingClass= median(as.numeric(chanceCreationPassingClass),
na.rm = TRUE),

chanceCreationCrossing= mean(chanceCreationCrossing, na.rm = TRUE),

chanceCreationCrossingClass= median(as.numeric(chanceCreationCrossingClass),
na.rm = TRUE),

chanceCreationShooting= mean(chanceCreationShooting, na.rm = TRUE),

chanceCreationShootingClass= median(as.numeric(chanceCreationShootingClass),
na.rm = TRUE),

chanceCreationPositioningClass=
median(as.numeric(chanceCreationPositioningClass), na.rm = TRUE),

defencePressure= mean(defencePressure, na.rm = TRUE),

defencePressureClass= median(as.numeric(defencePressureClass), na.rm = TRUE),

defenceAggression= mean(defenceAggression , na.rm = TRUE),

defenceAggressionClass= median(as.numeric(defenceAggressionClass), na.rm =
TRUE),

defenceTeamWidth= mean(defenceTeamWidth , na.rm = TRUE),

defenceTeamWidthClass= median(as.numeric(defenceTeamWidthClass), na.rm =
TRUE),

defenceDefenderLineClass= median(as.numeric(defenceDefenderLineClass), na.rm
= TRUE))

teams<-ungroup(teams)

teams<-as.data.frame(teams)
```

```
###Transforming class variables:
teams[,4]<-factor(teams[,4],labels = c("Slow", "Balanced", "Fast"))
teams[,6]<-factor(teams[,6],labels = c("Little", "Normal", "Lots"))
teams[,8]<-factor(teams[,8],labels = c("Long", "Mixed", "Short"))
teams[,9]<-factor(teams[,9],labels = c("Free Form", "Organised"))
teams[,11]<-factor(teams[,11],labels = c( "Risky", "Normal","Safe"))
teams[,13]<-factor(teams[,13],labels = c("Little", "Normal", "Lots"))
teams[,15]<-factor(teams[,15],labels = c("Little", "Normal", "Lots"))
teams[,16]<-factor(teams[,16],labels = c("Free Form", "Organised"))
teams[,18]<-factor(teams[,18],labels = c("Deep", "Medium","High"))
teams[,20]<-factor(teams[,20],labels = c("Double", "Press","Contain"))
teams[,22]<-factor(teams[,22],labels = c("Narrow", "Normal","Wide"))
teams[,23]<-factor(teams[,23],labels = c("Cover", "Offsid Trap"))
teams_t<-select(teams,c(1,2,4,6,8,9,11,13,15,16,18,20,22,23))

#Players
players<-player_Attributes[,-(1:2)]
players[,2]<-as.Date(players[,2])
players<-mutate(players,year= year(players[,2]))
###Transforming class variables:
players[,5]<-factor(players[,5],levels = c("right","left"))
players[,6]<-factor(players[,6],levels = c("low", "medium","high"))
players[,7]<-factor(players[,7],levels = c("low", "medium","high"))
###Calculating means and medians over years:
# Create the mode function.
getmode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
```

```
# Calculating means.
players<-group_by(players,player_api_id,year)%>%
summarise(overall_rating=mean(overall_rating,na.rm = TRUE),
potential= mean(potential,na.rm = TRUE),
preferred_foot= getmode(preferred_foot),
attacking_work_rate= median(as.numeric(attacking_work_rate),na.rm = TRUE),
defensive_work_rate= median(as.numeric(defensive_work_rate),na.rm = TRUE),
crossing= mean(crossing,na.rm = TRUE),
finishing= mean(finishing,na.rm = TRUE),
heading_accuracy= mean(heading_accuracy,na.rm = TRUE),
short_passing= mean(short_passing,na.rm = TRUE),
volleys= mean(volleys,na.rm = TRUE),
dribbling= mean(dribbling,na.rm = TRUE),
curve= mean(curve,na.rm = TRUE),
free_kick_accuracy= mean(free_kick_accuracy,na.rm = TRUE),
long_passing= mean(long_passing,na.rm = TRUE),
ball_control= mean(ball_control,na.rm = TRUE),
acceleration= mean(acceleration,na.rm = TRUE),
sprint_speed= mean(sprint_speed,na.rm = TRUE),
agility= mean(agility,na.rm = TRUE),
reactions= mean(reactions,na.rm = TRUE),
balance= mean(balance,na.rm = TRUE),
shot_power= mean(shot_power,na.rm = TRUE),
jumping= mean(jumping,na.rm = TRUE),
stamina= mean(stamina,na.rm = TRUE),
strength= mean(strength,na.rm = TRUE),
long_shots= mean(long_shots,na.rm = TRUE),
aggression= mean(aggression,na.rm = TRUE),
interceptions= mean(interceptions,na.rm = TRUE),
positioning= mean(positioning,na.rm = TRUE),
vision= mean(vision,na.rm = TRUE),
penalties= mean(penalties,na.rm = TRUE),
marking= mean(marking,na.rm = TRUE),
standing_tackle= mean(standing_tackle,na.rm = TRUE),
sliding_tackle= mean(sliding_tackle,na.rm = TRUE),
gk_diving= mean(gk_diving,na.rm = TRUE)
,
gk_handling= mean(gk_handling,na.rm = TRUE),
gk_kicking= mean(gk_kicking,na.rm = TRUE),
gk_positioning= mean(gk_positioning,na.rm = TRUE),
gk_reflexes= mean(gk_reflexes,na.rm = TRUE))

players<-ungroup(players)
players<-as.data.frame(players)

###Calculating skills, mental, physical and gk attributes
players<-
mutate(players,skills=c(1:73059),physical=c(1:73059),mental=c(1:73059),gk_att
=c(1:73059))
for (j in 1:73059){
  players$skills[j]<-
mean(as.vector(t(players[j,c(8:17,23,27,32,34,35)])),na.rm = TRUE)
}
for (j in 1:73059){
  players$physical[j]<-mean(as.vector(t(players[j,c(18:22,24:26)])),na.rm =
TRUE)
}
for (j in 1:73059){
  players$mental[j]<-mean(as.vector(t(players[j,c(28:31,33)])),na.rm = TRUE)
}
for (j in 1:73059){
  players$gk_att[j]<-mean(as.vector(t(players[j,c(36:40)])),na.rm = TRUE)
}
```

```
##Transform variables to categorical:
for (i in 41:44) {
  for (j in 1:73059) {
    if (players[j,i]>=0 & players[j,i]<40 ){players[j,i]<-"very poor"}
    if (players[j,i]>=40 & players[j,i]<50 ){players[j,i]<-"poor"}
    if (players[j,i]>=50 & players[j,i]<70 ){players[j,i]<-"fair"}
    if (players[j,i]>=70 & players[j,i]<80 ){players[j,i]<-"good"}
    if (players[j,i]>=80 & players[j,i]<90 ){players[j,i]<-"very good"}
    if (players[j,i]>=90 & players[j,i]<100 ){players[j,i]<-"excellent"}
  }
}
players[,41]<-factor(players[,41],levels = c("very
poor","poor","fair","good","very good","excellent"))
players[,42]<-factor(players[,42],levels = c("very
poor","poor","fair","good","very good","excellent"))
players[,43]<-factor(players[,43],levels = c("very
poor","poor","fair","good","very good","excellent"))
players[,44]<-factor(players[,44],levels = c("very
poor","poor","fair","good","very good","excellent"))


players_t<-select(players,c(1,2,41:44))


#Matches
matches<-Match[,-c(1,2,4,5,78:115)]
matches[,2]<-as.Date(matches[,2])
matches<-mutate(matches,year=year(matches[,2]))
##Reduction
matches<-gather(matches,player_stat,player_id,-c(1:7,74))
matches<-separate(matches,player_stat,into = c("state","x","y"))
matches<-matches[,-c(11,10)]
matches<-na.omit(matches)
```

```
##Merging with players:
match1<-select(matches,c(3,8,9,10))
match1<-merge(match1,players_t,by.x = c("year","player_id"),by.y =
c("year","player_api_id"))
match_p_home<-filter(match1,state=="home")
match_p_away<-filter(match1,state=="away")


matches<-merge(matches,match_p_home, by.x = c("match_api_id"),by.y =
c("match_api_id"))
matches<-merge(matches,match_p_away, by.x = c("match_api_id"),by.y =
c("match_api_id"))


matches<-merge(matches,teams_t,by.x = c("year","home_team_api_id"),by.y =
C("year","team_api_id"))
matches<-merge(matches,teams_t,by.x = c("year","away_team_api_id"),by.y =
C("year","team_api_id"))
##Match results:
for (j in 1:25979) {
  if (matches[j,6]> Match1[j,7]){
    matches$winner[j]<-"Home Team W0n"
  }
  if(Match1[j,6]< Match1[j,7]){
    matches$winner[j]<-"Away Team W0n"
  }
  if (Match1[j,6]== Match1[j,7]){
    matches$winner[j]<-"Draw"
  }
}
```

#Secound bulding the decision tree:

```
create_train_test <- function(data, size = 0.8, train = TRUE) {
  n_row = nrow(data)
  total_row = size * n_row
  train_sample < - 1: total_row
  if (train == TRUE) {
    return (data[train_sample, ])
  } else {
    return (data[-train_sample, ])
  }
}


fit <- rpart(Winner~., data = data_train, cp=0.0009)
tiff('test.tiff', units="in", width=10, height=8, res=600, compression =
'lzw')
rpart.plot(fit, type = 5, extra=8,branch.lty=3)
dev.off()
predict_unseen <-predict(fit, data_test, type = 'class')
table_mat <- table(data_test$Winner, predict_unseen)
accuracy_Test <- (sum(diag(table_mat)) / sum(table_mat))*100
print(paste('Accuracy for test', accuracy_Test, "%"))
```

References

1. Aggarwal, C. C. (2015). *Data mining: the textbook*. Springer.
2. Alonso-Fernandez, C., Calvo-Morata, A., Freire, M., Martinez-Ortiz, I., & Fernández-Manjón, B. (2019). Applications of data science to game learning analytics data: A systematic literature review. *Computers & Education*, *141*, 103612.
3. Berry, J. (1994). Database: A Potent New Tool for Selling. *Bus. Week*, *Sep*, *5*.
4. Bhatia, P. (2019). *Data mining and data warehousing: principles and practical techniques*. Cambridge University Press.
5. Blum, A., Hopcroft, J., & Kannan, R. (2016). Foundations of data science. *Vorabversion eines Lehrbuchs*, *5*, 5.
6. Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, *16*(3), 199-231.
7. Caffo, B. (2015). Regression models for data science in R. *Canada: Leanpub*.
8. Carmichael, I., & Marron, J. S. (2018). Data science vs. statistics: two cultures?. *Japanese Journal of Statistics and Data Science*, *1*(1), 117-138.
9. Cielen, D., Meysman, A. D., & Ali, M. (2016). Introducing Data Science: Big Data. *Machine Learning and More, Using Python Tools. Manning, Shelter Island, US*, *322*.
10. Clarke, B., Fokoue, E., & Zhang, H. H. (2009). *Principles and theory for data mining and machine learning*. Springer Science & Business Media.
11. Cleveland, W. S. (2001). Data science: an action plan for expanding the technical areas of the field of statistics. *International statistical review*, *69*(1), 21-26.
12. Davenport, T. H., & Patil, D. J. (2012). Data scientist. *Harvard business review*, *90*(5), 70-76.
13. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). *Mathematics for machine learning*. Cambridge University Press.
14. De Veaux, R. D., Agarwal, M., Averett, M., Baumer, B. S., Bray, A., Bressoud, T. C., ... & Ye, P. (2017). Curriculum guidelines for undergraduate programs in data science. *Annual Review of Statistics and Its Application*, *4*, 15-30.
15. Diggle, P. J. (2015). Statistics: a data science for the 21st century. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *178*(4), 793-813.

16. Donoho, D. (2017). 50 years of data science. *Journal of Computational and Graphical Statistics*, *26*(4), 745-766.

17. Engel, J. (2017). Statistical literacy for active citizenship: A call for data science education. *Statistics Education Research Journal*, *16*(1), 44-49.

18. Finzer, W. (2013). The data science education dilemma. *Technology Innovations in Statistics Education*, *7*(2).

19. Foote, K. D. (2016). A Brief History of Data Science. *Dataversity*, *December*, *14*.

20. Giudici, P. (2005). *Applied data mining: statistical methods for business and industry*. John Wiley & Sons.

21. Giordani, P., Ferraro, M. B., & Martella, F. (2020). *An Introduction to Clustering with R*. Springer Singapore.

22. Godsey, B. (2017). *Think like a data scientist*. Pearson Professional Computing.

23. Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

24. Härdle, W., Lu, H. H. S., & Shen, X. (Eds.). (2018). *Handbook of big data analytics*. Springer International Publishing.

25. Hardin, J., Hoerl, R., Horton, N. J., Nolan, D., Baumer, B., Hall-Holt, O., ... & Ward, M. D. (2015). Data science in statistics curricula: Preparing students to "think with data". *The American Statistician*, *69*(4), 343-353.

26. Hayashi, C. (1998). What is data science? Fundamental concepts and a heuristic example. In *Data science, classification, and related methods* (pp. 40-51). Springer, Tokyo.

27. Horton, N. J., Baumer, B. S., & Wickham, H. (2014). Teaching precursors to data science in introductory and second courses in statistics. *arXiv preprint arXiv:1401.3269*.

28. Hui, E. G. M. (2019). *Learn R for Applied Statistics*. Eric Goh Ming Hui.

29. Igual, L., & Seguí, S. (2017). Introduction to data science. In *Introduction to Data Science* (pp. 1-4). Springer, Cham.

30. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.

31. KDD-89: IJCAI-89 Workshop on Knowledge Discovery in Databases. August 20, 1989, Detroit MI, USA

32. Kroese, D. P., Botev, Z. I., Taimre, T., & Vaisman, R. (2019). *Data science and machine learning: Mathematical and statistical methods*. Chapman and Hall/CRC.

33. Lawson, J. (2014). *Design and Analysis of Experiments with R* (Vol. 115). CRC press.

34. Larose, C. D., & Larose, D. T. (2019). *Data science using Python and R*. John Wiley & Sons.

35. Lee, S. J., & Siau, K. (2001). A review of data mining techniques. *Industrial Management & Data Systems*.

36. Loy, A., Kuiper, S., & Chihara, L. (2019). Supporting data science in the statistics curriculum. *Journal of Statistics Education*, *27*(1), 2-11.

37. Milborrow, S. (2016). Plotting rpart trees with the rpart. plot package.

38. Miller, J. D. (2017). *Statistics for Data Science: Leverage the power of statistics for Data Analysis, Classification, Regression, Machine Learning, and Neural Networks*. Packt Publishing Ltd.

39. Mueller, J. P., & Massaron, L. (2019). *Data Science Programming All-in-One For Dummies*. John Wiley & Sons.

40. Muller, M., Lange, I., Wang, D., Piorkowski, D., Tsay, J., Liao, Q. V., ... & Erickson, T. (2019, May). How data science workers work with data: Discovery, capture, curation, design, creation. In *Proceedings of the 2019 CHI conference on human factors in computing systems* (pp. 1-15).

41. Naur, P. (1974). *Concise survey of computer methods*. Petrocelli Books.

42. O'Neil, C., & Schutt, R. (2013). *Doing data science: Straight talk from the frontline*. " O'Reilly Media, Inc.".

43. Palumbo, F., Lauro, C. N., & Greenacre, M. J. (2010). Data analysis and classification. In *Proceedings of the 6th Conference of the Classification and Data Analysis Group of the Societ Italiana di Statistica", Springer*.

44. Peng, R. D., & Matsui, E. (2015). The art of data science. *A Guide for Anyone Who Works with Data. Skybrude Consulting, LLC*.

45. Peng, R. D. (2016). *R programming for data science* (pp. 86-181). Victoria, BC, Canada: Leanpub.

46. Pierson, L. (2015). *Data science for dummies*. 2nd edition. John Wiley & Sons.

47. Provost, F., & Fawcett, T. (2013). *Data Science for Business: What you need to know about data mining and data-analytic thinking*. " O'Reilly Media, Inc.".

48. Ratner, B. (2017). *Statistical and machine-learning data mining:: Techniques for better predictive modeling and analysis of big data*. CRC Press.

49. Rivera, R. (2020). *Principles of managerial statistics and data science*. John Wiley & Sons.

50. Robinson, E., & Nolis, J. (2020). *Build a Career in Data Science*. Manning Publications.

51. Rusu, O., Halcu, I., Grigoriu, O., Neculoiu, G., Sandulescu, V., Marinescu, M., & Marinescu, V. (2013, January). Converting unstructured and semi-structured data into knowledge. In *2013 11th RoEduNet International Conference* (pp. 1-4). IEEE.

52. Sarker, I. H., Kayes, A. S. M., Badsha, S., Alqahtani, H., Watters, P., & Ng, A. (2020). Cybersecurity data science: an overview from machine learning perspective. *Journal of Big data*, *7*(1), 1-29.

53. Schembera, B., & Durán, J. M. (2020). Dark data as the new challenge for big data science and the introduction of the scientific data officer. *Philosophy & Technology*, *33*(1), 93-115.

54. Seaman, D. P., Chaves, J. J., & Bugbee, K. S. (2017). Benchmarking Big Data Cloud-Based Infrastructures.

55. Severini, T. A. (2020). *Analytic methods in sports: Using mathematics and statistics to understand data from baseball, football, basketball, and other sports*. Crc Press.

56. Shmueli, G., Bruce, P. C., Yahav, I., Patel, N. R., & Lichtendahl Jr, K. C. (2017). *Data mining for business analytics: concepts, techniques, and applications in R*. John Wiley & Sons.

57. Spiegelhalter, D. (2019). *The art of statistics: Learning from data*. Penguin UK.

58. Tan, P. N., Steinbach, M., & Kumar, V. (2006). Data mining introduction.

59. Tewari, S. H. (2020). Data Science and its applications in Cyber Security (Cyber Security Data Science). *Available at SSRN 3687251*.

60. Therneau, T. M., & Atkinson, E. J. (2019). An introduction to recursive partitioning using the RPART routines. 2018. *URL: https://cran. r-project. org/web/packages/rpart/vignettes/longintro. pdf (date of access: 21.11. 2018)*.

61. Tukey, J. W. (1962). The future of data analysis. *The annals of mathematical statistics*, *33*(1), 1-67.

62. Van der Loo, M., & De Jonge, E. (2018). *Statistical data cleaning with applications in R*. John Wiley & Sons.

63. Wasserman, L. (2014). Rise of the machines. *Past, Present and Future of Statistical Science. Ed. by Xihong Lin, Christian Genest, David Banks, Geert Molenberghs, David Scott, and Jane*, 525-536.

64. Wickham, H. (2014). Tidy data. *Journal of statistical software, 59*(1), 1-23.

65. Wickham, H., & Grolemund, G. (2016). *R for data science: import, tidy, transform, visualize, and model data*. " O'Reilly Media, Inc.".