Bachelor thesis

Statistics and Machine Learning in Sports Injury Prevention

Melissa GENOUD

Summer semester 2024

Supervisor: Dr. Senthil Murugan NAGARAJAN

Co-Supervisor: Prof. Dr. Christophe LEY

Contents

1 General Introduction **1**

1.1 Injuries in team sports . . . 1

1.2 General concept of injury prediction using statistics and machine learning? . . . 1

1.3 Main challenges . . . 2

1.4 General idea of the project . . . 2

1.5 Motivation for this project . . . 3

1.5.1 Use of the project in high school . . . 3

2 Data cleaning and preparation **4**

2.1 Dataset description . . . 4

2.1.1 NBA Injuries from 2010 - 2018 . . . 4

2.1.2 NBA players info . . . 5

2.2 Data cleaning and preparation: Injury dataset . . . 6

2.2.1 Replacing null values . . . 6

2.2.2 Categorising the 'Notes' . . . 7

2.2.2.1 Body part . . . 7

2.2.2.2 Injury type . . . 8

2.2.2.3 Injury detail . . . 9

2.2.2.4 Other . . . 10

2.2.3 Code . . . 11

2.3 Data cleaning and preparation: Information dataset . . . 13

2.3.1 Transforming the dates into the right format . . . 13

2.4 Merging datasets . . . 14

2.5 Removing irrelevant information . . . 15

2.6 Removing duplicates . . . 16

3 Descriptive statistics **17**

3.1 Injuries per player . . . 17

3.2 Injuries per team . . . 18

3.3 Injuries per year . . . 20
3.4 Injuries per month . . . 21
3.5 Injuries per body part . . . 24
3.6 Injuries per type . . . 25
3.7 Injuries per detail . . . 26
3.8 Other reasons to miss a game . . . 27
3.9 Injuries by age range . . . 27
3.10 Injuries by weight range . . . 29

4 Inferential statistics 30
4.1 Relationship between position and injuries . . . 30
4.1.1 Chi-square test of independence . . . 30
4.1.2 Relationship between position and injured body part . . . 32
4.1.2.1 Results of the Chi-square test of independence . . . 33
4.1.2.2 Insights and interpretation by position . . . 33
4.1.3 Relationship between position and injury type . . . 37
4.1.3.1 Results of the Chi-square test of independence . . . 39
4.1.3.2 Important observable insights by injury type . . . 39
4.1.4 Relationship between position and injury detail . . . 41
4.1.4.1 Results of the chi-square test of independence . . . 42
4.1.4.2 Important deviations by position contributing most to the high chi-square value . . . 43
4.2 Influence of age on injuries . . . 44
4.3 Influence of weight on injuries . . . 45
4.4 Relationship between all player variables and their injuries . . . 45
4.4.1 Multiple class logistic regression . . . 45
4.4.2 Logistic regression to determine the injured body part . . . 49
4.4.3 Logistic regression to determine the injury type . . . 50
4.4.4 Logistic regression to determine the injury detail . . . 50

5 Teaching 52
5.1 Introduction to data . . . 52
5.2 Activity to introduce statistics and visualisation . . . 52
5.3 Introduction to proportion and percentages . . . 54

6 Conclusion 55

Chapter 1

General Introduction

1.1 Injuries in team sports

In all kinds of team sports, injury is one of the biggest obstacles each athlete is facing in their respective careers. They can have significant physical, psychological and financial consequences [2] on a team and will definitely have an impact on the athlete's sports career depending on the severity of the injury. The success of many teams often relies on the fitness and health of their key players. Coaches and team staffs are therefore more and more interested in including the use of rapidly evolving technologies to predict and prevent eventual future injuries to maximize the team's performance and keep their players in the greatest physical shape.

While Artificial Intelligence (AI) methods have already had success in the domain of sports when it is used for game analysis, tactics, performance analysis and outcome predictions, it has started to make its appearance in sports medicine in order to foresee inciting injuries and identify the injury risk for an athlete [4]. This allows coaches and team staff to have the necessary information to manage the workload, make individual training adjustments, adopt injury prevention strategies and adapt game strategies to the elevated injury risk [1].

1.2 General concept of injury prediction using statistics and machine learning?

Injury prediction using statistics and Machine Learning (ML) generally consists of collecting historical data (such as information on previous injuries), player characteristics (such as weight, height, age, sex, muscle strength and speed), environmental factors (such as weather, playing surface or training facilities) along with many other relevant variables. While statistics are often used to analyse patterns and connections between variables within an injury dataset, ML on the other

hand uses those patterns in order to predict future injuries based on the occurrence of previous injuries when similar patterns were observed. To use ML, we need to split ideally our dataset in train and test sets. Our training set is usually around 20% of the whole dataset and is used to find patterns and relationships within the data in order to learn how these patterns affect the result. After training the model, we use the other 80% of the dataset, where we know the result that we are supposed to get, in order to see how accurately the model can predict the result without knowing what the actual result will be.

1.3 Main challenges

Sports science and sports medicine has become an important field of research for statisticians, data scientists and computer scientists, but they are often facing difficult challenges:

- The lack of data related to injury incidence is one of the main reasons why risk estimations may decrease in accuracy [5]. Indeed, fortunately for the athletes, there are not many injuries occurring during a season leading to a small sample size of injured athletes compared to athletes without injuries.
- It is very difficult to access injury data as health related data needs to ensure privacy. If it is not publicly available data, consent to work with this type of data is hard to obtain because of ethical guidelines.
- As there are many interactions between intrinsic (individual) and extrinsic (environmental) risk factors and unpredictable factors that are added, such as contact, it is sometimes very difficult to be able to predict an injury [4].
- A high number of variables is required in order to make accurate injury predictions.

We therefore need to take a step back and understand the limitations injury-prediction methods can have and focus more on identifying an injury risk on an individual level [4].

1.4 General idea of the project

For this project, we will use a publicly available injury dataset from the National Basketball Association (NBA) including all types of reasons why an athlete has been excluded from playing a game. We will use statistical methods to find relations and patterns in the data from which we will draw conclusions. We will be working with the platform *Kaggle*, which allows us to manipulate

datasets found on this platform or elsewhere, using a code in Python language. In addition to this, we will be using *Excel*, which contains multiple functions to manipulate datasets and plot graphs.

1.5 Motivation for this project

Being a basketball player and coach, I have experienced many injuries myself but I have also seen a lot of athletes around me get injured, for some of them resulting in years of recovery and rehabilitation, which had big impacts on the team's success, but most importantly on the player's life and sports career. As a Bachelor in Mathematics student, I think it is very interesting that by using knowledge in statistics as well as learning about machine learning algorithms, I can be able to prevent some injuries and therefore have a positive impact on a team, and more importantly, a player's career.
Mathematical projects which are based on events that we experience in our life, can also show many young people that mathematics can be used in a domain that they have a high interest in, which might encourage more students to pursue their studies in the fields of mathematics, data science or computer science. This aspect is important to me, as my professional goal is to become a high school mathematics teacher.

1.5.1 Use of the project in high school

I will be able to use this project in my future teaching career, to teach students the basics of statistics, like mean, median, mode and range, or to analyse the frequency of injury types across teams and seasons. I will also be able to introduce graphs and visualisation techniques, to create and read histograms, bar graphs and scatter plots. I can encourage my future students to make connections between mathematics and other subjects, like biology or physics, but also to show them the possible applications.

Chapter 2

Data cleaning and preparation

2.1 Dataset description

Definition 1. [8] A *population* is the entire set of objects, measurements or individuals whose properties are being observed and analysed .

For our statistical analysis on injuries in sports, we have decided to focus on basketball and take datasets which have the players of the NBA as population.
We have worked with two different datasets:

2.1.1 NBA Injuries from 2010 - 2018

The dataset includes all injury reports from the NBA covering the period of the season 2010 - 2011 to the season 2017 - 2018. This data was scraped from the *Pro Sports Transactions* website [10].

Definition 2. [8] A *variable*, or also called a data item, is the metric of interest for each individual or object in a population .

The raw dataset is given under the format of a tabular and contains 9783 observations (rows) and 5 variables (columns):

- **Date**: This variable indicates the day a player has been added to the injury report or has returned to the lineup after having missed previous games. There are 9783 observations for this variable, of which 1581 are unique dates. The day with the most observations (54) is April 26, 2012. This variable falls under the category of ordinal qualitative data.

- **Team**: All 30 NBA teams are present in our dataset. The team 'Charlotte Bobcats' also features in the dataset, but have become the 'Charlotte Hornets' in the 2014-2015 season. There are 9780 observations in the dataset, meaning that 3 players that have been injured, were not playing for any team. All three were reported as placed on injury leave due to

surgery. The 'Milwaukee Bucks' is the team represented the most in the dataset with 462 observations. This variable falls under the category of nominal qualitative data.

- **Acquired**: The 'Acquired' variable tells us when a player has returned or been added to a team. This can help us calculate the time a player has been placed on injury leave for a specific injury. There are 1589 observations, of which 486 are unique. Kevin Love is the player that has been acquired most times (18). This variable falls under the category of nominal qualitative data.

- **Relinquished**: 8194 players have been relinquished from a game, of which 778 are unique. Kevin Love is also the player that was relinquished most times, with 78 appearances in injury reports. This variable falls under the category of nominal qualitative data.

- **Notes**: The 'Notes' indicate the reason a player has been relinquished or acquired. This variable falls under the category of nominal qualitative data.

	Date	Team	Acquired	Relinquised	Notes
count	9783	9780	1589	8194	9783
unique	1581	31	486	778	1790
top	2012-04-26	Bucks	Kevin Love	Kevin Love	returned to lineup
freq	54	462	18	78	1580

Figure 2.1: Description of the injury dataset by using the data in [10].

2.1.2 NBA players info

In order to have more information on the injured players, we have added another dataset, that we will merge with the previous one, so that we can have a better understanding on the different injury reasons.
This dataset contains information on players that were active between the years 1947 and 2018. It has been collected from the website *Basketball-reference* [7].
This raw dataset is also given under the form of a tabular and contains a total of 4574 observations (rows) and 8 different features (columns):

- **Player:** This nominal qualitative variable indicates the NBA player of whom information is given.

- **From:** This variable indicates the first year the player in question has played in the NBA. It is an ordinal qualitative variable.
- **To:** The 'To' feature indicates the last year the player in question has played in the NBA. However, if the last year is given as '2018', in most cases it is not true, because for all the players that have been active after 2018, the last year is also given as '2018'. Since our injury dataset also stops at the year 2018, this is not a problem for our project. This data falls under the category of ordinal qualitative data.
- **Pos:** This nominal qualitative variable indicates the position of a player on the court. In basketball there are 5 different positions: Point guard (PG), Shooting guard (SG), Small forward (SF), Power forward (PF) and Center (C). In this dataset, they have regrouped PG and SG in the guard (G) category and the SF and PF in the Forward (F) category.
- **Ht:** This variable indicates the height of an NBA player (given in feet and inches) and falls under the category of continuous numerical variables.
- **Wt:** The 'Wt' feature is a continuous numerical variable indicating the weight of an NBA player (given in lbs [1]).
- **Birth date:** This feature gives us information on when a player was born, which allows us to calculate his age at the time of injury. This data can be categorised as an ordinal qualitative variable.
- **College:** The last variable is a nominal qualitative variable informing us in which college a player was active before entering the NBA.

2.2 Data cleaning and preparation: Injury dataset

2.2.1 Replacing null values

As observed in section 2.1, we only have a total number of 1589 observations for the 'Acquired' variable, meaning that there are $9783 - 1589 = 8194$ null values for this variable. The number 8194 corresponds to the amount of observations for the 'Relinquished' variable.
We will therefore replace all the missing values for the 'Acquired' variable and replace it with 'NA' (not acquired), as shown in Figure 2.2. We will then do the exact same thing for the variables 'Team' and 'Relinquished', by replacing their null values by 'NT', respectively 'NR'.

[1] lbs is the abbreviation for pounds: 1lb = 0.4536 kg

In [5]:
```
#Replacing missing values in 'Acquired' with NA
df['Acquired'] = df['Acquired'].fillna('NA')
df.head()
```

Out[5]:

	Date	Team	Acquired	Relinquised	Notes
0	2010-10-03	Bulls	NA	Carlos Boozer	fractured bone in right pinky finger (out inde...
1	2010-10-06	Pistons	NA	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)
2	2010-10-06	Pistons	NA	Terrico White	broken fifth metatarsal in right foot (out ind...
3	2010-10-08	Blazers	NA	Jeff Pendergraph / Jeff Ayres	torn ACL in right knee (out indefinitely)
4	2010-10-08	Nets	NA	Troy Murphy	strained lower back (out indefinitely)

Figure 2.2: Replacing null values by 'NA'.

Code in Figure 2.2:

- *df['Acquired'] = df['Acquired'].fillna('NA')*:
  - *df['Acquired']*: column of the 'Acquired' variable in the injury DataFrame (df).
  - *fillna('NA')*: replaces any null values, denoted by 'NaN', in the 'Acquired' column with 'NA'.
- *df.head()*: This function presents the first five rows of df after the transformation.

The same code can be used for the strings 'NR' and 'NT'.

2.2.2 Categorising the 'Notes'

With the notes given under this form, we cannot make any type of statistical analysis regarding injury types. We will create 4 more variables, which will illustrate what types of injuries NBA players suffer most from. In order to do this, we had to go through the 1790 unique observations in the notes and write down all the keywords we come across with, in order to be able to categorise the notes by recognising these specific keywords.

2.2.2.1 Body part

The 'body part' variable contains 9 different categories: upper body, hip/pelvis, leg, foot, arm, knee, hand, back, head/face. Here is the list of key words that have been chosen to be mapped to their respective body parts:

- upper body: 'abdominal', 'rib', 'chest', 'sternum', 'abdomen', 'oblique', 'collarbone', 'pectoral', 'core'

- hip/pelvis: 'hip', 'pelvis', 'abductor', 'tailbone', 'groin', 'pubic', 'SI'
- leg: 'leg', 'achilles', 'hamstring', 'calf', 'fibula', 'quadricep', 'shin', 'thigh', 'tibia', 'gastrocnemius'
- foot: 'foot', 'recovering from surgery on left foot (DNP)[2]', 'ankle', 'metatarsal', 'heel', 'toe', 'toenail', 'plantar', 'fascia'
- arm: 'arm', 'elbow', 'forearm', 'shoulder', 'flexor', 'biceps', 'axilary', 'rotator cuff', 'triceps'
- knee: 'knee', 'patella', 'kneecap', 'meniscus'
- hand: 'hand', 'wrist', 'finger', 'thumb'
- back: 'back', 'cervical', 'neck', 'spinal'
- head/face: 'head', 'face', 'facial', 'nose', 'eye', 'orbital', 'concussion', 'corneal', 'eyelid', 'cheekbone', 'jaw', 'mouth', 'oral', 'sinus'.

[14]:

	Date	Team	Acquired	Relinquised	Notes	body_part
0	2010-10-03	Bulls	NA	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand
1	2010-10-06	Pistons	NA	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg
2	2010-10-06	Pistons	NA	Terrico White	broken fifth metatarsal in right foot (out ind...	foot
3	2010-10-08	Blazers	NA	Jeff Pendergraph / Jeff Ayres	torn ACL in right knee (out indefinitely)	knee
4	2010-10-08	Nets	NA	Troy Murphy	strained lower back (out indefinitely)	back

Figure 2.3: Head of the dataset with the 'body part' variable.

When none of the above keywords are recognised, we chose to enter 'NBP' (No body part).

2.2.2.2 Injury type

The 'injury type' variable describes how the body part mentioned is injured. It is divided into 14 categories: hernia, strain, sore, pain, broken, sprain, bruise, swollen, dislocated, concussion, stress, tear, inflammation and abrasion. Here is the list of key words that have been chosen to be mapped to their respective injury types:

- hernia: 'hernia', 'herniated'

[2]This keyword has been added as a longer phrase, because the program was not recognising the keyword 'foot'

- strain: ‘strain’, ‘strained’
- sore: ‘spasm’, ‘sore’, ‘soreness’, ‘stiff’, ‘stiffness’, ‘tightness’
- pain: ‘pain’
- broken: ‘broken’, ‘fractured’, ‘splits’, ‘splints’, ‘spurs’, ‘damaged’
- sprain: ‘sprain’, ‘hyperextended’
- bruise: ‘bruise’
- swollen: ‘bulging’, ‘swelling’, ‘swollen’
- dislocated: ‘dislocated’, ‘separated’
- concussion: ‘concussion’
- stress: ‘fatigue’, ‘stress’
- tear: ‘tear’, ‘torn’, ‘rupture’
- inflammation: ‘infection’, ‘synovitis’, ‘inflammation’, ‘dermatitis’, ‘bursitis’, ‘tendinitis’
- abrasion: ‘abrasion’, ‘scratch’, ‘lacerated’.

When none of the above keywords are recognised, we chose to enter ‘NIT’ (No injury type).

[15]:

	Date	Team	Acquired	Relinquised	Notes	body_part	injury_type
0	2010-10-03	Bulls	NA	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand	broken
1	2010-10-06	Pistons	NA	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg	tear
2	2010-10-06	Pistons	NA	Terrico White	broken fifth metatarsal in right foot (out ind...	foot	broken
3	2010-10-08	Blazers	NA	Jeff Pendergraph / Jeff Ayres	torn ACL in right knee (out indefinitely)	knee	tear
4	2010-10-08	Nets	NA	Troy Murphy	strained lower back (out indefinitely)	back	strain

Figure 2.4: Head of the dataset with the ‘injury type’ variable.

2.2.2.3 Injury detail

We have named ‘injury detail’ the variable that describes which part of the musculoskeletal system, in each of the injured body part, is affected. This variable is divided into 8 different categories: tissue, tendon, nerve, muscle, joint, cartilage, bone and ligament. Here is the list of key words that have been chosen to be mapped to their respective injury details:

- tissue: 'fascia', 'tissue', 'fascitis'
- tendon: 'Achilles', 'tendon', 'tendinitis', 'rotator cuff',
- nerve: 'nerve'
- muscle: 'calf', 'hamstring', 'flexor', 'muscle', 'thigh', 'quadricep', 'abdomen', 'gastrocnemius', 'oblique', 'groin', 'abductor', 'biceps', 'spasms', 'cramp', 'soreness'
- joint: 'joint', 'synovitis', 'shoulder', 'hip', 'bursae'
- cartilage: 'cartilage', 'disc', 'labrum', 'meniscus'
- bone: 'bone', 'heel', 'tibia', 'rib', 'pelvis', 'fibula', 'shin', 'metatarsal', 'cervical', 'dislocated', 'broken', 'kneecap', 'fracture'
- ligament: 'ligament', 'MCL', 'ACL', 'sprain'.

When none of the above keywords are recognised, we chose to enter 'NID' (No injury detail).

[16]:

	Date	Team	Acquired	Relinquised	Notes	body_part	injury_type	injury_detail
0	2010-10-03	Bulls	NA	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand	broken	bone
1	2010-10-06	Pistons	NA	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg	tear	tendon
2	2010-10-06	Pistons	NA	Terrico White	broken fifth metatarsal in right foot (out ind...	foot	broken	bone
3	2010-10-08	Blazers	NA	Jeff Pendergraph / Jeff Ayres	torn ACL in right knee (out indefinitely)	knee	tear	ligament
4	2010-10-08	Nets	NA	Troy Murphy	strained lower back (out indefinitely)	back	strain	NID

Figure 2.5: Head of the dataset with the 'injury detail' variable.

2.2.2.4 Other

Every other reason for which a player misses a game, which does not fit into the other 3 variables, has been placed under the variable 'other', which is divided into 8 categories: activated (describes when someone has returned to the lineup), personal, fined, illness, procedure (medical procedures that do not include surgery), rehab, rest and surgery.

- activated: 'activated', 'returned'
- personal: 'personal'
- fined: 'fined'

- illness: 'ill', 'vertigo', 'headache', 'flu', 'cold', 'chicken pox', 'food poisoning', 'gastroenteritis', 'stomach', 'throat', 'viral', 'virus', 'allergic', 'dehydration', 'blood', 'bronchitis', 'heart', 'pneumonia', 'dizziness', 'pulmonary', 'respiratory'
- procedure: 'procedure', 'treatment','dental','tooth'
- rehab: 'rehab'
- rest: 'rest', 'resting'
- surgery: 'surgery','appendectomy'.

When none of the above keywords are recognised, we chose to enter 'NO' (No other).

[80]:

	Date	Team	Acquired	Relinquised	Notes	body_part	injury_type	injury_detail	other
32	2010-11-03	Pistons	NA	Will Bynum	sore right hamstring (DNP)	leg	sore	muscle	NO
33	2010-11-03	Warriors	NA	Brandan Wright	stomach virus (DTD)	NBP	NIT	NID	illness
34	2010-11-04	Blazers	NA	Elliot Williams	dislocated patella in right knee (surgery, out...	knee	dislocated	bone	surgery
35	2010-11-05	Bucks	NA	Andrew Bogut	migraine headache (DNP)	head/face	NIT	NID	illness
36	2010-11-05	Hawks	NA	Marvin Williams	bone bruise in right knee (DNP)	knee	bruise	bone	NO

Figure 2.6: Extract of the dataset with the 'other' variable.

2.2.3 Code

To do this categorisation of the notes, we have used the same kind of code for every new feature. We will describe the code for one of the variables, say 'injury type':

- *def categorise_injury_type(notes)*: We define a function, which will categorise the injuries of the dataset, based on the text written in the 'Notes' variable.
- *injury_types*: We define a dictionary which contains injury type categories as keys and a list of keywords, which will be mapped to their designated category.
- The function we defined will go through the list of keywords and calculate the partial ratio score between the notes and the different keywords.
- *best_matches*: We define a new dictionary, which for each category contains the best matching keyword recognised in the notes along with its matching score. We set the initial score to -1, to ensure that during the process of comparison, any score obtained will be greater than -1, as they usually lie between 0 and 100.

```python
def categorise_injury_type(notes):
    injury_types = {
        'hernia': ['hernia', 'herniated'],
        'strain': ['strain', 'strained'],
        'sore': ['spasm', 'sore', 'soreness', 'stiff', 'stiffness', 'tightness'],
        'pain': ['pain'],
        'broken': ['broken', 'fractured', 'splits', 'splints', 'spurs', 'damaged'],
        'sprain': ['sprain', 'hyperextended'],
        'bruise': ['bruise'],
        'swollen': ['bulging', 'swelling', 'swollen'],
        'dislocated': ['dislocated', 'separated'],
        'concussion': ['concussion'],
        'stress': ['fatigue', 'stress'],
        'tear': ['tear', 'torn', 'rupture'],
        'inflammation': ['infection', 'synovitis', 'inflammation', 'dermatitis', 'bursitis', 'tendinitis'],
        'abrasion': ['abrasion', 'scratch', 'lacerated']
    }
    best_matches = {category: {'keyword': None, 'score': -1} for category in injury_types.keys()}
    for category, keywords in injury_types.items():
        for keyword in keywords:
            score = fuzz.partial_ratio(keyword.lower(), notes.lower())
            if score > best_matches[category]['score']:
                best_matches[category] = {'keyword': keyword, 'score': score}
    best_match_category = max(best_matches, key=lambda x: best_matches[x]['score'])
    best_match_keyword = best_matches[best_match_category]['keyword']
    threshold = 85
    if best_matches[best_match_category]['score'] < threshold:
        return 'NIT'
    return best_match_category
df['injury_type'] = df['Notes'].apply(categorise_injury_type)
df.head()
```

Figure 2.7: Code to create the 'injury type' variable.

- Outer loop: Iterates over the different injury type categories.
- Inner loop: Iterates over the different keywords in their respective category.
- *fuzz.partial_ratio()*: This function provided by the *'fuzzywuzzy'* library, is given to calculate the matching score between the lower case version of the notes and the lower case version of the keywords, in order to ensure that case differences don't play a role in the matching score.
- Update of the best match: If using the previous function we find a keyword with a higher score than the keyword found in *best_matches* for a category, we update this dictionary with the better keyword.
- After numerous iterations through categories and keywords, looking at the *best_matches* dictionary, we find the category with highest matching score and name it *best_match_category*, as well as the keyword with the best matching score and name it *best_match_keyword*.

- If there is no keyword found with a higher matching score than the threshold set to 85 for the best matching category, we return the string 'NIT'.
- Otherwise, we return the best matching category.
- *df['injury_type']= df['Notes'].apply(categorise_injury_type)*: We define as *'injury_type'* the result of the application of the function *categorise_injury_type* to every element in the notes.

2.3 Data cleaning and preparation: Information dataset

2.3.1 Transforming the dates into the right format

[12]:

	Player	From	To	Pos	Ht	Wt	Birth Date	Colleges
0	alaa abdelnaby	1991	1995	F-C	06-Oct	240.0	June 24, 1968	Duke University
1	zaid abdul aziz	1969	1978	C-F	06-Sep	235.0	April 7, 1946	Iowa State University
2	kareem abdul jabbar	1970	1989	C	07-Feb	225.0	April 16, 1947	University of California, Los Angeles
3	mahmoud abdul rauf	1991	2001	G	06-Jan	162.0	March 9, 1969	Louisiana State University
4	tariq abdul wahad	1998	2003	F	06-Jun	223.0	November 3, 1974	University of Michigan, San Jose State University

Figure 2.8: Head of the information dataset before transforming the birth dates.

As we can observe in Figure 2.8, the birth date is given by the format *Month D, Yr*, which does not correspond to the format we have for the date of injury for our dates at the time of injury *YYYY-MM-DD*. We will change both date variables to the format *DD/MM/YYYY*, so that we can calculate the age of a player when he suffers an injury.

[13]:
```
df['Birth Date'] = pd.to_datetime(df['Birth Date'])
df['Birth Date'] = df['Birth Date'].dt.strftime('%d/%m/%Y')
df.head()
```

[13]:

	Player	From	To	Pos	Ht	Wt	Birth Date	Colleges
0	alaa abdelnaby	1991	1995	F-C	06-Oct	240.0	24/06/1968	Duke University
1	zaid abdul aziz	1969	1978	C-F	06-Sep	235.0	07/04/1946	Iowa State University
2	kareem abdul jabbar	1970	1989	C	07-Feb	225.0	16/04/1947	University of California, Los Angeles
3	mahmoud abdul rauf	1991	2001	G	06-Jan	162.0	09/03/1969	Louisiana State University
4	tariq abdul wahad	1998	2003	F	06-Jun	223.0	03/11/1974	University of Michigan, San Jose State University

Figure 2.9: Head of the information dataset after transforming the birth dates.

2.4 Merging datasets

We have merged the dataset containing all injury reports with the dataset containing information about the players, to add more information to our initial dataset and be able to draw conclusions on the reasons players get injured. We have chosen to add the features 'Birth Date', 'Position' and 'Weight' from the information dataset to our injury dataset and to calculate the age at the time of injury for a relinquished or acquired player using *Excel* and its *DATEDIF()* formula.
To be able to merge both datasets, the name of the player on the information dataset had to match the name of the acquired or relinquished player, which was not always the case. Some players on the injury dataset had two names written down (real name and nickname), which caused problems to merge. We have therefore manually changed all the names that didn't match in both datasets and have added all the missing positions, birth dates and weights that were missing, in order to have the same amount of information on every injured player.

	Date	Team	Acquired	Relinquished	Notes	body_part	injury_type	injury_detail	other	Birth Date	Age	Position	Weight
0	03/10/2010	Bulls	NaN	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand	broken	bone	NO	20/11/1981	28	F-C	258
1	06/10/2010	Pistons	NaN	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg	tear	tendon	NO	02/03/1987	23	F	231
2	06/10/2010	Pistons	NaN	Terrico White	broken fifth metatarsal in right foot (out ind...	foot	broken	bone	NO	07/03/1990	20	G	213
3	08/10/2010	Blazers	NaN	Jeff Ayres	torn ACL in right knee (out indefinitely)	knee	tear	ligament	NO	29/04/1987	23	F	240
4	08/10/2010	Nets	NaN	Troy Murphy	strained lower back (out indefinitely)	back	strain	NID	NO	02/05/1980	30	F-C	245

Figure 2.10: Merged datasets.

2.5 Removing irrelevant information

We have removed all the data corresponding to coaches who have been acquired or relinquished, as they are irrelevant for determining injury risks on players. After doing so, we are left with a total of 9770 observations. This has also been done using *Excel* by filtering out all notes containing the word 'coach'.
We have also removed the 'Acquired' column, as we are interested in who is getting injured, rather than who is coming back from an injury. The 'Acquired' variable would have been useful, if for every injury a player has suffered, the date of his return was mentioned in the dataset, which would have allowed us to calculate the time he was out due to his injury. Unfortunately, we did not have this kind of information on many injuries, which is why we have removed this variable from the dataset using the *.drop()* function. After doing this, we have also removed all the the rows containing 'NR' values in the column 'Relinquished', as these were the rows which had information on acquired players. We have then renamed the variable 'Relinquished' by 'Player', which indicates the player who has suffered an injury.
At this point, our dataset contained 8185 observations.

```
df
```

	Date	Team	Player	Notes	body_part	injury_type	injury_detail	other	Birth Date	Age	Position	Weight
0	03/10/2010	Bulls	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand	broken	bone	NO	20/11/1981	28	F-C	258
1	06/10/2010	Pistons	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg	tear	tendon	NO	02/03/1987	23	F	231
2	06/10/2010	Pistons	Terrico White	broken fifth metatarsal in right foot (out ind...	foot	broken	bone	NO	07/03/1990	20	G	213
3	08/10/2010	Blazers	Jeff Ayres	torn ACL in right knee (out indefinitely)	knee	tear	ligament	NO	29/04/1987	23	F	240
4	08/10/2010	Nets	Troy Murphy	strained lower back (out indefinitely)	back	strain	NID	NO	02/05/1980	30	F-C	245
...	...	...	...	...	...	...	...	...	...	...	...	...
9763	09/05/2018	Celtics	Shane Larkin	sprained left shoulder (out for season)	arm	sprain	joint	NO	02/10/1992	25	G	175
9764	09/05/2018	Thunder	Paul George	arthroscopic surgery on right knee (out indefi...	knee	NIT	NID	surgery	02/05/1990	28	F	220
9765	22/05/2018	Warriors	Andre Iguodala	bruised left leg (DTD)	leg	bruise	NID	NO	28/01/1984	34	G-F	215
9766	25/05/2018	Rockets	Chris Paul	strained right hamstring (out for season)	leg	strain	muscle	NO	06/05/1985	33	G	175
9767	26/05/2018	Cavaliers	Kevin Love	concussion (DTD)	head/face	concussion	NID	NO	07/09/1988	29	F	251

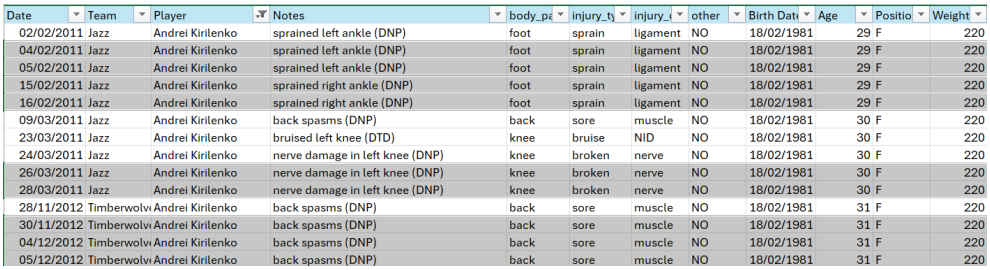
8185 rows × 12 columns

Figure 2.11: Dataset after removing irrelevant information.

We have also removed all the rows containing 'activated', 'personal', 'fined', 'rehab', as they do not give us information about injuries and there were only very few observations.

2.6 Removing duplicates

As our dataset contains information on when a player has not played a game, instead of the day a player has been injured, there were a lot of cases where the same injury was mentioned several times in the dataset, as some have missed multiple games due to the same injury. This dataset therefore did not allow us to analyse the different injury occurrences. It also did not allow us to figure out how long a player has been missing games, because not all injuries were repeated in the dataset, even if they had to miss multiple games. Players who have been missing for a long time, just the few first games are mentioned.
This led us to take the decision to remove all the duplicate injuries. We have done this manually using *Excel* to ensure that the right duplicates were removed, as we want to keep the same injuries which happened at a later time.



Date	Team	Player	Notes	body_pa	injury_ty	injury_d	other	Birth Dat	Age	Positio	Weight
02/02/2011	Jazz	Andrei Kirilenko	sprained left ankle (DNP)	foot	sprain	ligament	NO	18/02/1981	29	F	220
04/02/2011	Jazz	Andrei Kirilenko	sprained left ankle (DNP)	foot	sprain	ligament	NO	18/02/1981	29	F	220
05/02/2011	Jazz	Andrei Kirilenko	sprained left ankle (DNP)	foot	sprain	ligament	NO	18/02/1981	29	F	220
15/02/2011	Jazz	Andrei Kirilenko	sprained right ankle (DNP)	foot	sprain	ligament	NO	18/02/1981	29	F	220
16/02/2011	Jazz	Andrei Kirilenko	sprained right ankle (DNP)	foot	sprain	ligament	NO	18/02/1981	29	F	220
09/03/2011	Jazz	Andrei Kirilenko	back spasms (DNP)	back	sore	muscle	NO	18/02/1981	30	F	220
23/03/2011	Jazz	Andrei Kirilenko	bruised left knee (DTD)	knee	bruise	NID	NO	18/02/1981	30	F	220
24/03/2011	Jazz	Andrei Kirilenko	nerve damage in left knee (DNP)	knee	broken	nerve	NO	18/02/1981	30	F	220
26/03/2011	Jazz	Andrei Kirilenko	nerve damage in left knee (DNP)	knee	broken	nerve	NO	18/02/1981	30	F	220
28/03/2011	Jazz	Andrei Kirilenko	nerve damage in left knee (DNP)	knee	broken	nerve	NO	18/02/1981	30	F	220
28/11/2012	Timberwolv	Andrei Kirilenko	back spasms (DNP)	back	sore	muscle	NO	18/02/1981	31	F	220
30/11/2012	Timberwolv	Andrei Kirilenko	back spasms (DNP)	back	sore	muscle	NO	18/02/1981	31	F	220
04/12/2012	Timberwolv	Andrei Kirilenko	back spasms (DNP)	back	sore	muscle	NO	18/02/1981	31	F	220
05/12/2012	Timberwolv	Andrei Kirilenko	back spasms (DNP)	back	sore	muscle	NO	18/02/1981	31	F	220

Figure 2.12: Removing duplicate information from the dataset.

On the Figure 2.12 above, the grey rows are the ones which will be removed from the dataset. After removing all irrelevant information and all the duplicates, we are left with a dataset of 4760 observations.

```
df
```

	Date	Team	Player	Notes	body_part	injury_type	injury_detail	other	Birth Date	Age	Position	Weight
0	03/10/2010	Bulls	Carlos Boozer	fractured bone in right pinky finger (out inde...	hand	broken	bone	NO	20/11/1981	28	F-C	258
1	06/10/2010	Pistons	Jonas Jerebko	torn right Acchilles tendon (out indefinitely)	leg	tear	tendon	NO	02/03/1987	23	F	231
2	06/10/2010	Pistons	Terrico White	broken fifth metatarsal in right foot (out ind...	foot	broken	bone	NO	07/03/1990	20	G	213
3	08/10/2010	Blazers	Jeff Ayres	torn ACL in right knee (out indefinitely)	knee	tear	ligament	NO	29/04/1987	23	F	240
4	08/10/2010	Nets	Troy Murphy	strained lower back (out indefinitely)	back	strain	NID	NO	02/05/1980	30	F-C	245
...	...	...	...	...	...	...	...	...	...	...	...	...
4755	07/05/2018	Jazz	Dante Exum	strained left hamstring (out for season)	leg	strain	muscle	NO	13/07/1995	22	G	190
4756	09/05/2018	Celtics	Shane Larkin	sprained left shoulder (out for season)	arm	sprain	joint	NO	02/10/1992	25	G	175
4757	09/05/2018	Thunder	Paul George	arthroscopic surgery on right knee (out indefi...	knee	NIT	NID	surgery	02/05/1990	28	F	220
4758	22/05/2018	Warriors	Andre Iguodala	bruised left leg (DTD)	leg	bruise	NID	NO	28/01/1984	34	G-F	215
4759	25/05/2018	Rockets	Chris Paul	strained right hamstring (out for season)	leg	strain	muscle	NO	06/05/1985	33	G	175

4760 rows × 12 columns

Figure 2.13: Dataset after removing duplicates.

Chapter 3

Descriptive statistics

3.1 Injuries per player

```
#Number of injuries per player
df_injuries = df[df['other'] == 'NO'].copy()
player_injury_count = df_injuries['Player'].value_counts()
sorted_players = player_injury_count.index.tolist()
total_injury_count = player_injury_count.sum()
print("Total Injury Count", total_injury_count)
player_injury_count[0:10]
```

```
Total Injury Count 3436
Player
Anthony Davis     26
Nene Hilario      24
Tyreke Evans      24
Dwyane Wade       23
Tony Parker       22
Jameer Nelson     22
Kevin Love        22
Luol Deng         21
Rodney Stuckey    20
Andrew Bogut      20
Name: count, dtype: int64
```

Figure 3.1: Top 10 most relinquished players.

Observations: The player that has been injured the most, is Anthony Davis, with a total number of 26 injuries. He has played for the New Orleans Pelicans from the 2012-2013 season to the 2018-2019 season. The number of injuries per player, ranges between 0 and 26.
A total of 3436 injuries have been reported from 2010-2018.

Code in Figure 3.1:

- We have only included all the observations containing 'NO' in the 'other' variable, as this excludes all the players that were sick and only includes all the different injuries. We will denote this DataFrame by *df_injuries*.
- The function *.value_counts()* counts the number of observations for each unique injured player.
- The function *.index_tolist()* sorts all the players in a list from most occurrences to least.
- The two lines underneath calculate the total number of injuries by summing the injuries of all players in the dataset.

3.2 Injuries per team

Using the same code as in Figure 3.1, by replacing 'Player' with 'Team' in the code, we can calculate the number of injured players per team.
We can then plot the results in a bar graph using the following code:

```
#Bar graph of injuries per team
df_teams = df_injuries.sort_values(by=['Team', 'Date'])
df_teams = df_injuries.set_index('Team').loc[sorted_teams].reset_index()

plt.figure(figsize=(14, 6))
bars= plt.bar(sorted_teams, team_injury_count[sorted_teams], color='skyblue')
plt.xlabel('NBA Team')
plt.ylabel('Injury Count')
plt.title('Injury Count by Team')
plt.xticks(rotation=90)
for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, height, str(height), ha='center', va='bottom')
plt.show()
```

Figure 3.2: Code to plot the bar graph of injuries per team.

Code in Figure 3.2

- The first two lines sort the *df_injuries* DataFrame by team and date, which is then filtered to include teams out of the *sorted_teams* list, containing all the NBA teams ordered from most affected to least affected.

- We then plot the Figure, which is a bar graph, where one bar represents the count of injuries per NBA team.
- We customise the plot by naming the x and y axis, giving a title and rotating the names of the teams for better readability.
- *height = bar.get_height()*: we recover the height of each bar.
- We add text to each bar by calculating the coordinates on where the label should be *((x,y)=(bar.get_x() + bar.get_width() / 2, height))*, by converting the height to a string *(str(height))* and by specifying the horizontal alignment *(ha)* and the vertical alignment *(va)*.
- The last step is to display the plot.

We get the following results:

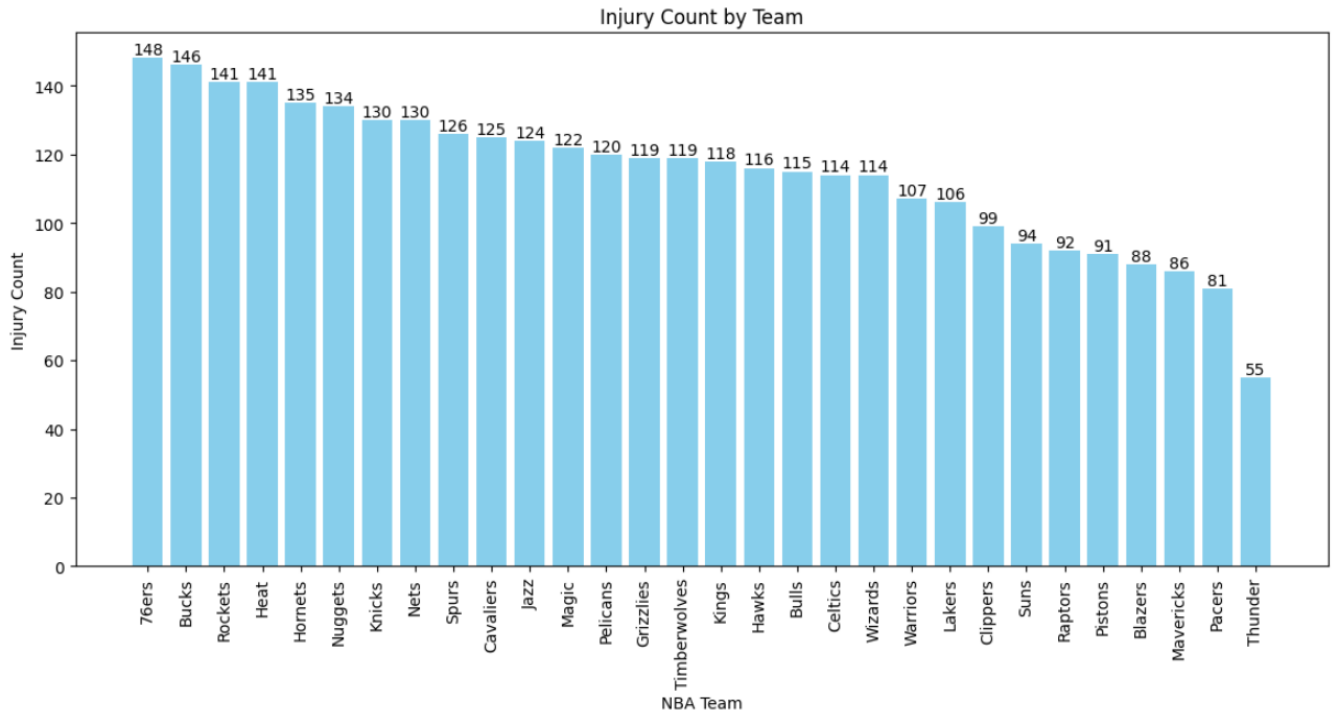


Figure 3.3: Bar graph of injuries per team.

Observations: We observe that the 'Philadelphia 76ers' have suffered most from injuries, with a total of 148 observations. The 'Oklahoma City Thunder' are clearly the team with the least amount of injuries, as they count 26 injuries less than the team with the second least amount. The number of relinquished players within a team ranges from 55 to 148.

3.3 Injuries per year

We have also analysed the amount of injuries per year. To do this we have used the following code:

```
#Injuries per year bar graph
df_injuries['Date'] = pd.to_datetime(df_injuries['Date'])
year_injury_count = df_injuries['Date'].dt.year.value_counts()
year_injury_count_df = year_injury_count.reset_index()
year_injury_count_df.columns = ['Date', 'Injury_Count']
sorted_years = year_injury_count.index.tolist()

plt.figure(figsize=(14, 6))
bars= plt.bar(sorted_years, year_injury_count[sorted_years], color='skyblue')
plt.xlabel('Year')
plt.ylabel('Injury Count')
plt.title('Injury Count by Year')
plt.xticks(rotation=90)
for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, height, str(height), ha='center', va='bottom')
plt.show()
```

Figure 3.4: Code for Figure 3.5.

Code in Figure 3.4

- The first part of this code is again similar to the code in Figure 3.1, with some modifications because we are working with dates. The *pd.to_datetime()* function from pandas converts the 'Date' column into datetime fromat, which is essential when working with dates. Another difference observed to the code in Figure 3.1 is *.dt.year.value_counts()* instead of just *.value_counts()*, as we want the number of occurrences of each year, and not each single date.
- The second part of the code is the same as the code in Figure 3.2, but instead of injuries per team, we have injuries per year.

We get the plot in Figure 3.5.

Observations: The bar graph in Figure 3.5 illustrates the distribution of the injuries over the years 2010 to 2018. One can observe that the year 2010 has a significant amount of fewer injuries compared to the other years. This observation can easily be explained with the fact that the dataset contains injuries starting in the season 2010-2011, meaning in October 2010, so only 3 out of the 12 months are analysed in 2010.

2016 is the year with the most injuries with 531 observations, followed closely behind by the year 2012 with 507 observations.

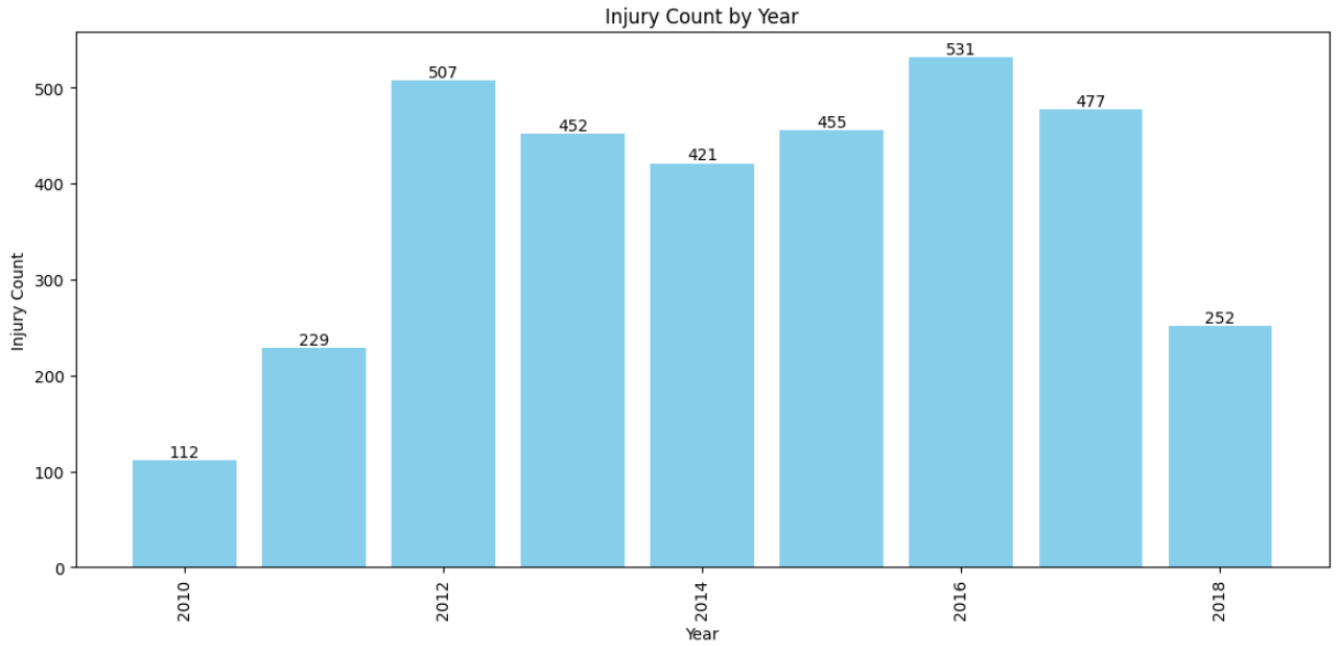


Figure 3.5: Bar graph of injuries per year.

3.4 Injuries per month

Code in Figure 3.6

- The first two lines are similar to previous codes.
- *month_injury_count = df_injuries.groupby('Month').size().reset_index(name='Injury Count')*: This line regroups all the data by month and calculates the size of each group with the *.size()* function. We reset the index and rename the column 'Injury Count'.
- We calculate the percentage per month by dividing by the total amount of injuries and multiplying by 100.
- The next line creates a DataFrame containing all the months from 1 to 12.
- We merge the *'month_injury_count'* with the *'all_months'* DataFrame on the 'Month' column.
- We plot the bar graph similarly as in Figure 3.4, with some differences: we manually name the different bars with the names of the months. We also use 'zip' to iterate through two different lists simultaneously and we pair each bar in the bar graph with its corresponding

```python
#Injuries per month
df_injuries['Date'] = pd.to_datetime(df_injuries['Date'])
df_injuries['Month'] = df_injuries['Date'].dt.month
month_injury_count = df_injuries.groupby('Month').size().reset_index(name='Injury Count')
total_injuries = month_injury_count['Injury Count'].sum()
month_injury_count['Injury Percentage']=(month_injury_count['Injury Count'] / total_injuries) * 100
all_months = pd.DataFrame({'Month': range(1, 13)})
month_injury_count = all_months.merge(month_injury_count, on='Month', how='left')

plt.figure(figsize=(12, 6))
bars=plt.bar(month_injury_count['Month'], month_injury_count['Injury Count'], color='skyblue')
plt.xlabel('Month')
plt.ylabel('Number of injuries')
plt.title('Number of Injuries per month')
plt.xticks(range(1, 13), ['January', 'February', 'March', 'April', 'May',
                          'June', 'July', 'August', 'September', 'October', 'November', 'December'], rotation=45)
plt.tight_layout()
for bar, percentage in zip(bars, month_injury_count['Injury Percentage']):
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, height, f"{percentage:.2f}%", ha='center', va='bottom')
plt.show()
```

Figure 3.6: Code for Figure 3.7

percentage value in the 'Injury Percentage'. Instead of putting as text label over the bars 'str(height)', we put *f"percentage:.2f%"*, which gives the percentage value with two decimal places.

Observations: The bar graph in Figure 3.7 describes the injury percentage distribution over the twelve months of the year. It is immediately observable, that there is a very low percentage of injuries in the months May to September.

This distribution is clearly explainable: the regular season of the NBA starts in mid October and is played until the end of April. October only has a low 5.27% of total injuries, because there are only a few pre-season games in the first half of October, where not many injuries occur. When the season starts, players usually have recovered from previous injuries through the summer break, have physically been prepared for the restart of the season and should be in a good shape, meaning they have good physical conditions, which should also be a reason for this low percentage in injuries.

From November to January, the amount of injuries rises from 12.19% (446 injuries) to 17.67% (607 injuries), before dropping again in February to 13.15% (452 injuries). An explanation for this drop could be the 'All Star Break' which usually lasts for almost a week in February, as the best players perform in a weekend of competitions and entertainment games, which are of very low intensity. There are therefore less games in February, compared to the other months. However, the month of March is the month with the most occurred injuries. Almost a fifth of all injuries occur in March

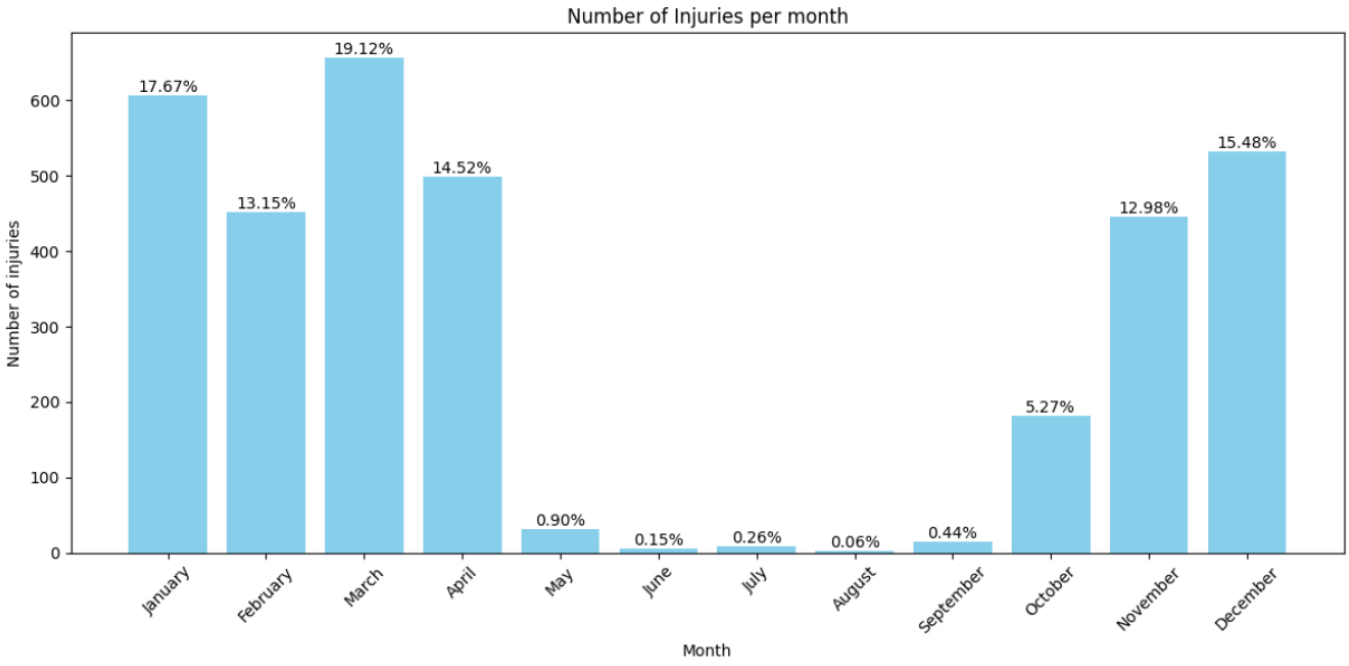


Figure 3.7: Bar graph of injuries per month.

(657 injuries). This leads us to think, that by the end of the regular season, players are physically tired and therefore at a higher risk of getting injured.
The regular season of the NBA season ends mid April, meaning that half of the teams finish their season in April, which is probably the reason for the drop in injuries. From the end of April to beginning of June are played the playoff series, in which only the top 16 teams out of the 30 teams play in elimination series, meaning that there are less active teams, less games and therefore less injuries. It is also not uncommon for NBA players to play through minor injuries in the playoffs, as this part of the season is really important and missing players can be the determining factor of who wins the championship title.

3.5 Injuries per body part

```python
#Bar graph of injuries per body part (percent)
body_part_counts = df_injuries['body_part'].value_counts().reset_index()
body_part_counts.columns = ['body_part', 'Injury Count']
total_injuries = body_part_counts['Injury Count'].sum()
body_part_counts['Injury Percentage'] = (body_part_counts['Injury Count'] / total_injuries) * 100

plt.figure(figsize=(12, 6))
bars = plt.bar(body_part_counts['body_part'], body_part_counts['Injury Count'], color='skyblue')
plt.xlabel('Body Part Injured')
plt.ylabel('Count of Injuries')
plt.title('Percentage of injuries per body part')
plt.xticks(rotation=45)
for bar, percentage in zip(bars, body_part_counts['Injury Percentage']):
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, height, f"{percentage:.2f}%", ha='center', va='bottom')
plt.show()
```

Figure 3.8: Code for Figure 3.9.

Code in Figure 3.8 This code only contains functions already used in previous codes and is the same kind of plot as in the code of Figure 3.6.

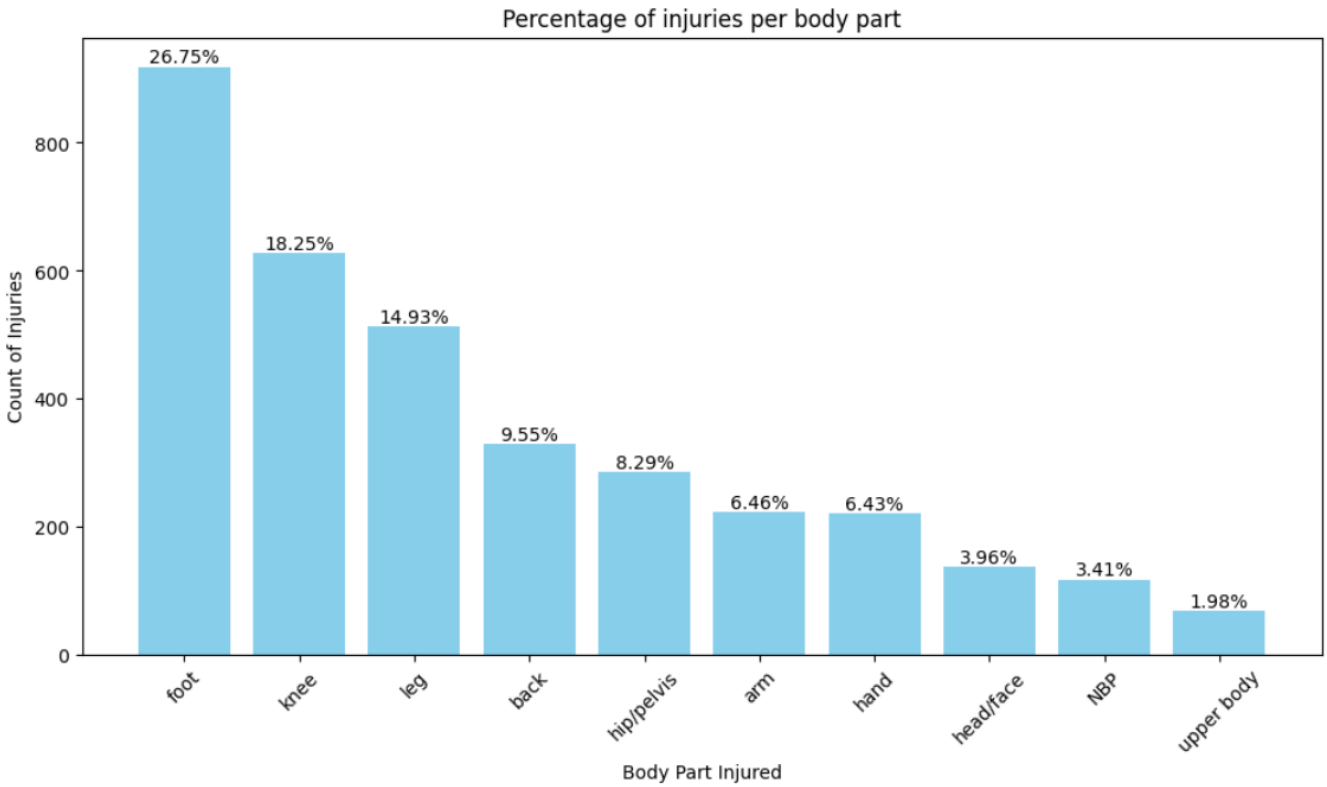


Figure 3.9: Bar graph of injuries per body part.

Observations: The bar graph in Figure 3.9 shows which parts of the body are most affected. The body part which is the most frequently injured is the foot which represents 26.75% of all injuries, as there have been 919 foot injuries, such as sprained ankles or fractures. Knee injuries are also frequent, with 627 injuries which represents almost a fifth of all injuries. Leg injuries, such as hamstring strains, is the third most injured body part as 14.93% of injuries are suffered on the leg. In 3.41% of injuries observed, no body part was specified. These injuries are described in the dataset with the note 'placed on injury leave' for example. The least common type of injuries are injuries to the upper body, with only 1.98% of our 3436 observations affecting the upper body.

3.6 Injuries per type

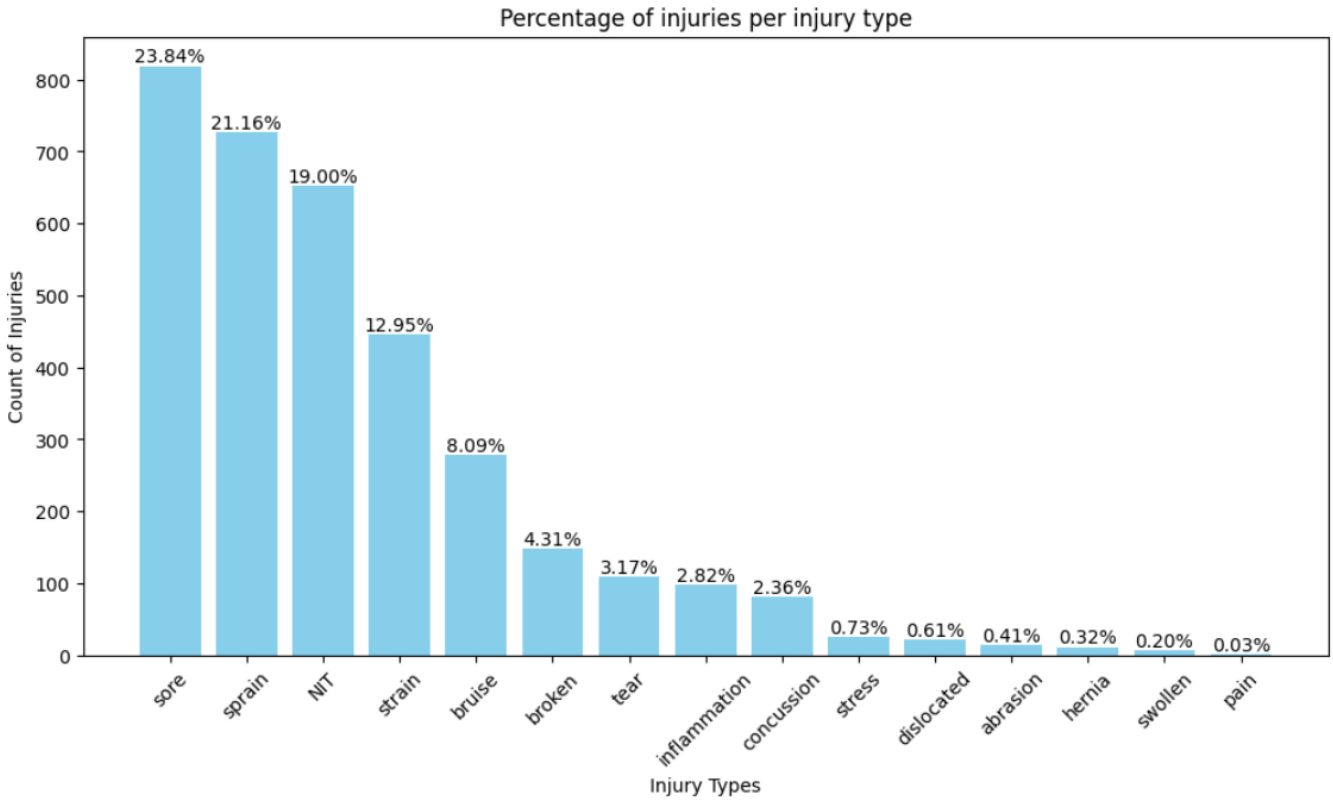


Figure 3.10: Bar graph of injuries per type.

Code for Figure 3.10: The code to create the bar graph in Figure 3.10 is the exact same code as in Figure 3.8, with the exception of the *'body_part'* feature being replaced by the *'injury_type'* feature.

Observations:
The bar graph in Figure 3.10 shows what types of injuries are most frequent in the dataset. Soreness is the main reason for players to miss a game, making up 23.84% of the 3436 injuries. Closely behind soreness are sprains, which are very common in basketball and represent 21.16% of our dataset.
Unfortunately, we can observe that 19% of injuries in the dataset do not have a specified type of injury. These notes are presented under the form 'placed on injury leave', 'injured left foot', etc. As this is a huge percentage, we need to take a step back and understand that we are limited to take conclusions from the type of injury.

3.7 Injuries per detail

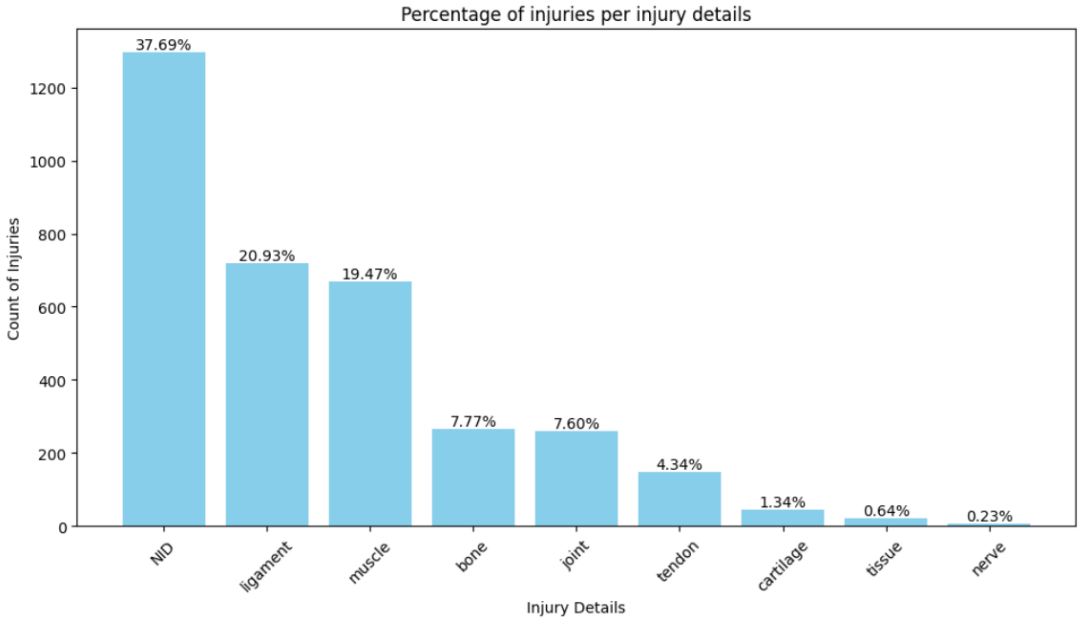


Figure 3.11: Bar graph of injuries per detail.

Code for Figure 3.11: The code to create the bar graph in Figure 3.11 is the exact same code as in Figure 3.8, with the exception of the *'body_part'* feature being replaced by the *'injury_detail'* feature.
Observations: The bar graph in Figure 3.11 shows what part of the musculoskeletal is most affected in the dataset. In 37.69% of the injuries described in the dataset, no part of the musculoskeletal system was mentioned and there was no information to be drawn from the notes to

include into this feature. When looking at the other 2141 injuries which included details on the injury, ligaments and muscles were the most injured parts of the system and represented 20.93% and 19.47% of the total amount of injuries. When excluding all the injuries without given details, injuries to the ligaments represent as much as 33.58% (719 injuries) and muscle injuries following right behind with 31.25% (669 injuries).

3.8 Other reasons to miss a game

```
#Other
other_counts = df['other'].value_counts().reset_index()
other_counts.columns = ['other', 'Injury Count']
total_other_counts = other_counts.sum()
print(other_counts)
```

```
       other  Injury Count
0         NO          3436
1    illness           482
2    surgery           434
3       rest           400
4  procedure             8
```

Figure 3.12: Count of the 'other' variable.

Observations: In 482 cases, players have missed a game because of an illness. 434 players have undergone surgery to repair an injury or for health related reasons and have therefore missed games for a larger period of time. There is also a big amount of players (400), who sit out games to rest in order to keep their body fit and prevent injuries.

3.9 Injuries by age range

Code in Figure 3.13:

- We define the ranges by checking the youngest and oldest player we have in the dataset. As there are 24 years between the youngest and the oldest, we take the ranges to be of 3 years, in order to have 8 different ranges.
- We then plot the histogram with the *plot.hist()* function, where we want to plot the age of injured players of *df_ injuries* with the bins being 8 age ranges.
- We customize the plot like we did for all the bar graphs.

```
#Injuries by age range
age_start = 18
age_end = 42
age_step = 3
age_ranges = range(age_start, age_end + 1, age_step)

plt.hist(df_injuries['Age'], bins=age_ranges, color='skyblue', edgecolor='black', alpha=0.7)
plt.title('Injuries by Age Ranges')
plt.xlabel('Age Ranges')
plt.ylabel('Number of Injuries')
plt.xticks(age_ranges)
plt.grid(axis='y', linestyle='--', alpha=0.7)

plt.show()
```

Figure 3.13: Code for Figure 3.14.

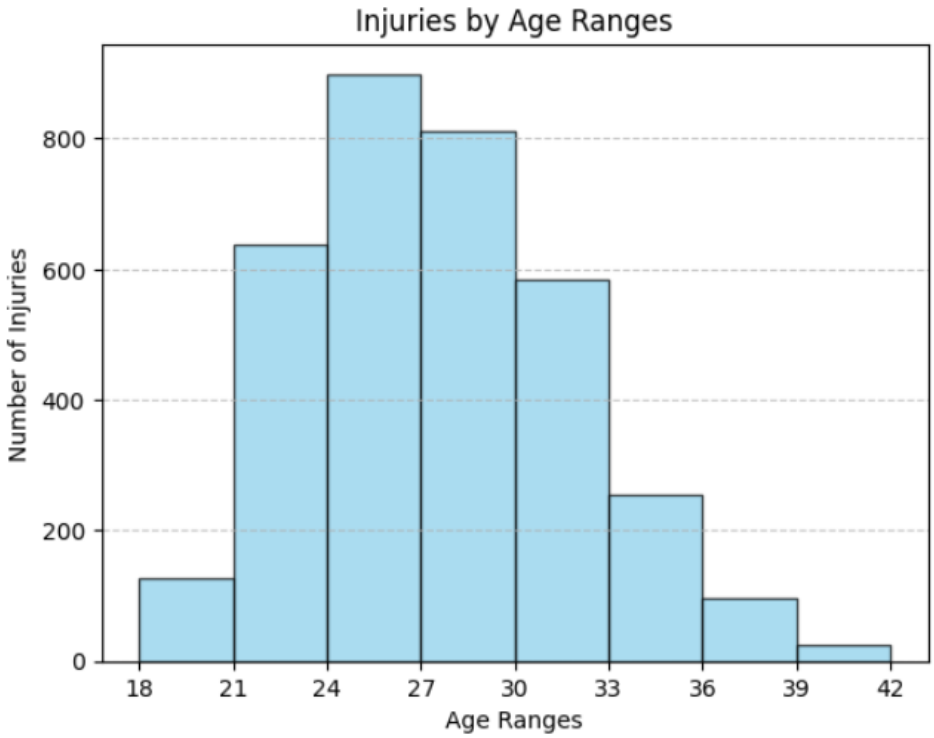


Figure 3.14: Histogram of injuries per age range.

Observations: The Figure 3.14 is a histogram illustrating the age at which players have suffered their respective injuries. We can observe that most injuries occur between 21 and 33 years old with its peak between 24 and 27. Most players get drafted when they are between 18 and 21 years old and need to find their role in the team, meaning that in their first years most players do not get as many minutes as more experienced players and are therefore less likely to get injured. They

are also young and usually fitter than older players, which decreases their injury risk. The reason for the drop after 33 years old is probably because most NBA players stop playing in their mid 30's, meaning there are less players playing at that age.
The mean of the age at which most players get injured is 27.06 and the median is found at the age of 27, while the mode is equal to 24.

3.10 Injuries by weight range

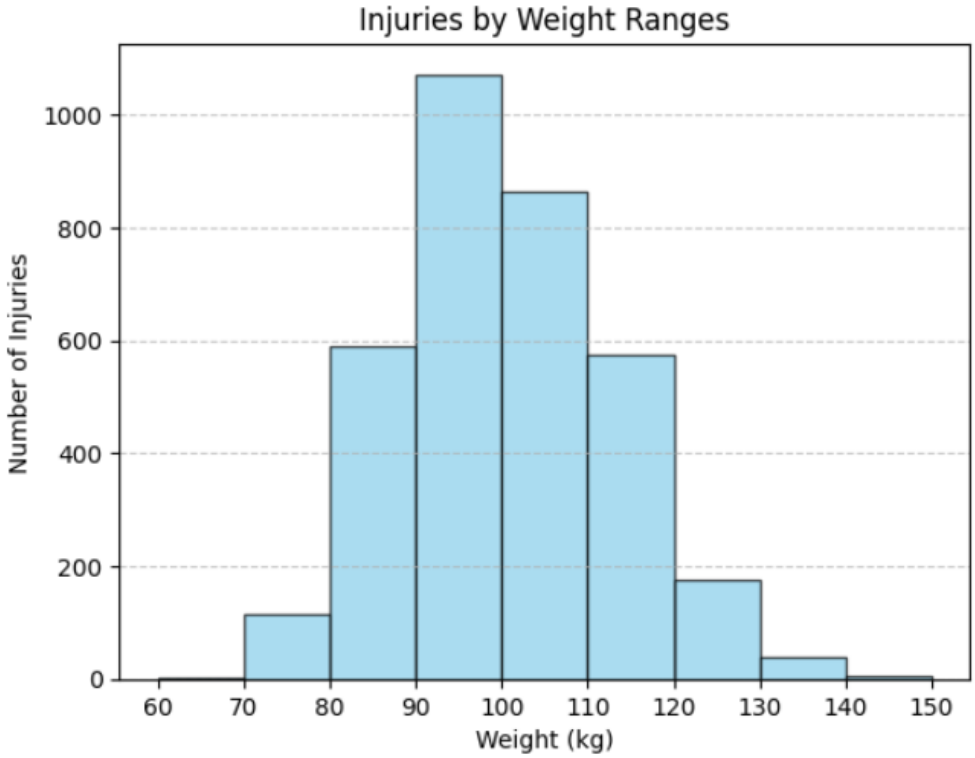


Figure 3.15: Histogram of injuries per weight range.

Code for Figure 3.15: We have used the same code as in Figure 3.13 and have replaced 'Age' by 'Weight'. We have also converted the weight, which was given in pounds, to kilograms by multiplying the initial weight values by 0.45359237.
Observations: The Figure 3.15 is a histogram which illustrates the weight of an injured player. The average weight of NBA players lies around 95kg, which explains the fact that the most injuries are counted in players which weigh between 90 and 100kg. The mean of weight of injured players is 100.73kg, while both the median and the mode are equal to 99.79kg.

Chapter 4

Inferential statistics

4.1 Relationship between position and injuries

4.1.1 Chi-square test of independence

The chi-square (χ^2) of independence is a hypothesis test in which we check if two qualitative variables depend on each other. This test observes frequencies of observations in all the combined groups and compares them to the frequencies one would expect if the variables were independent. The more similar the observed and expected values are, the higher the probability that the variables are unrelated [9]. To do so, this test requires a contingency table to analyse the data.

Definition 3. [3] A *contingency table* is an arrangement, where the categories of one categorical variable are represented in the rows and the categories of the other categorical variable are represented in the columns. A cell inside this arrangement contains the count of observations for this pair of categories .

Like every hypothesis test, the chi-square test checks two mutually exclusive hypothesis:

Definition 4. The *null hypothesis*, denoted as H_0, states that both variables are not related.

Definition 5. The *alternative hypothesis*, denoted as H_a or H_1, states that both variables are related.

We will calculate the test statistic for the chi-squared test in order to determine whether or not we should reject the null hypothesis.

Definition 6. [3] The *test statistic for the Chi-Square Test of Independence* is denoted by χ^2 and computed

$$\chi^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

where

- R is the amount of categories for the first variable,
- C is the amount of categories for the second variable,
- o_{ij} is the observed cell count in the i^{th} row and j^{th} column of the table,
- e_{ij} is the expected cell count in the i^{th} row and j^{th} column of the table, where every e_{ij} is computed as follows:

$$e_{ij} = \frac{(\text{total of } i^{th} \text{ row}) \cdot (\text{total of } j^{th} \text{ column})}{\text{grand total}}.$$

The value of $(o_{ij} - e_{ij})$ is called the *residual* of the cell (i, j) and is denoted by r_{ij}.

Definition 7. The *degree of freedom* is the amount of independent values that are free to vary and is calculated with the following formula:

$$df = (R - 1) \cdot (C - 1).$$

Definition 8. In statistical hypothesis testing, the *p value* is the probability of obtaining a result at least as extreme as that obtained, assuming the truth of the null hypothesis that the finding was the result of chance alone.

Let $\alpha = 0.05$ be our significance level, then:

- if the $p < \alpha$ we can conclude that we have a statistically significant test result and reject the null hypothesis. In this case, the results are only 5% likely to be as extreme as observed, given that the null hypothesis is true.
- If the $p \geq \alpha$ we can conclude that there is not necessarily a significant dependence between the variables and we cannot reject the null hypothesis.

Data requirements: [3]

- We need two categorical variables with at least 2 categories in each variable.
- There is no relationship between the subjects of each group, meaning that a subject of one group cannot be in another group.
- Expected frequencies in each cell need to be at least equal to 1.
- 80% of the expected frequencies in each cell need to be higher than 5.

4.1.2 Relationship between position and injured body part

In order to determine whether the position of a player has an impact on which body part is most likely to be injured, we have used the chi-square test of independence.

Check for data requirements:

- The 'Position' variable is a categorical variable containing 5 different categories and the 'body_part' is a categorical variable with 9 different categories.
- No player is mentioned in more than one position category and no player has more than one body part injured mentioned at the same time in the dataset.
- The lowest expected frequency is equal to 8.62 (centers with an upper body injury), meaning that all frequencies are above 1 and 100% of the frequencies are above 5.

We have all the data requirements necessary to perform the Chi-square test.

Injured body part per position										
Observed	leg	knee	foot	arm	back	head/face	hand	hip/pelvis	upper body	**Total**
G	229	255	380	110	109	49	127	116	25	1400
G-F	74	99	136	21	42	21	30	36	9	468
F	119	239	227	62	81	54	62	91	19	954
F-C	70	115	140	45	71	17	38	43	12	551
C	53	76	124	27	42	28	23	26	14	413
Total	545	784	1007	265	345	169	280	312	79	3786
Expected	leg	knee	foot	arm	back	head/face	hand	hip/pelvis	upper body	**Total**
G	201.53	289.91	372.37	97.99	127.58	62.49	103.54	115.37	29.21	1400
G-F	67.37	96.91	124.48	32.76	42.65	20.89	34.61	38.57	9.77	468
F	137.33	197.55	253.74	66.77	86.93	42.58	70.55	78.62	19.91	954
F-C	79.32	114.10	146.55	38.57	50.21	24.60	40.75	45.41	11.50	551
C	59.45	85.52	109.85	28.91	37.63	18.44	30.54	34.03	8.62	413
Total	545	784	1007	265	345	169	280	312	79	3786
Deviations	leg	knee	foot	arm	back	head/face	hand	hip/pelvis	upper body	
G	3.74	4.20	0.16	1.47	2.70	2.91	5.32	0.00	0.61	
G-F	0.65	0.04	1.07	4.22	0.01	0.00	0.61	0.17	0.06	
F	2.45	8.70	2.82	0.34	0.40	3.06	1.04	1.95	0.04	
F-C	1.09	0.01	0.29	1.07	8.61	2.35	0.19	0.13	0.02	
C	0.70	1.06	1.82	0.13	0.51	4.96	1.86	1.90	3.36	
Chi square	78.81									
p-value	0.000008									

Figure 4.1: Relationship between injured body part and position.

Hypotheses:

- H_0: The player's position is independent from the injured body part.
- H_a: The player's position is not independent from the injured body part.

4.1.2.1 Results of the Chi-square test of independence

Chi-square (χ^2):
In this case we have a Chi-square statistic equal to:

$$\chi^2 = 78.81.$$

Degrees of freedom:
There are

$$df = (5-1) \cdot (9-1) = 4 \cdot 8 = 32$$

degrees of freedom.
p **value:**
By using the *Excel* formula *CHISQ.DIST.RT(χ^2, df)*, we get a p value equal to

$$p = 0.000008.$$

Interpretation:

- The high χ^2 indicates that there is a big difference between the observed and expected values.
- Since we get $p = 0.000008$, which is clearly < 0.05 and indicates the probability of getting $\chi^2 = 78.81$ under H_0, we can reject the null hypothesis and accept that there is a significant association between the position of a player and his injured body part.

4.1.2.2 Insights and interpretation by position

By analysing both the graph in Figure 4.2 and the tables in Figure 4.1, we can get insights on how the two variables are associated.

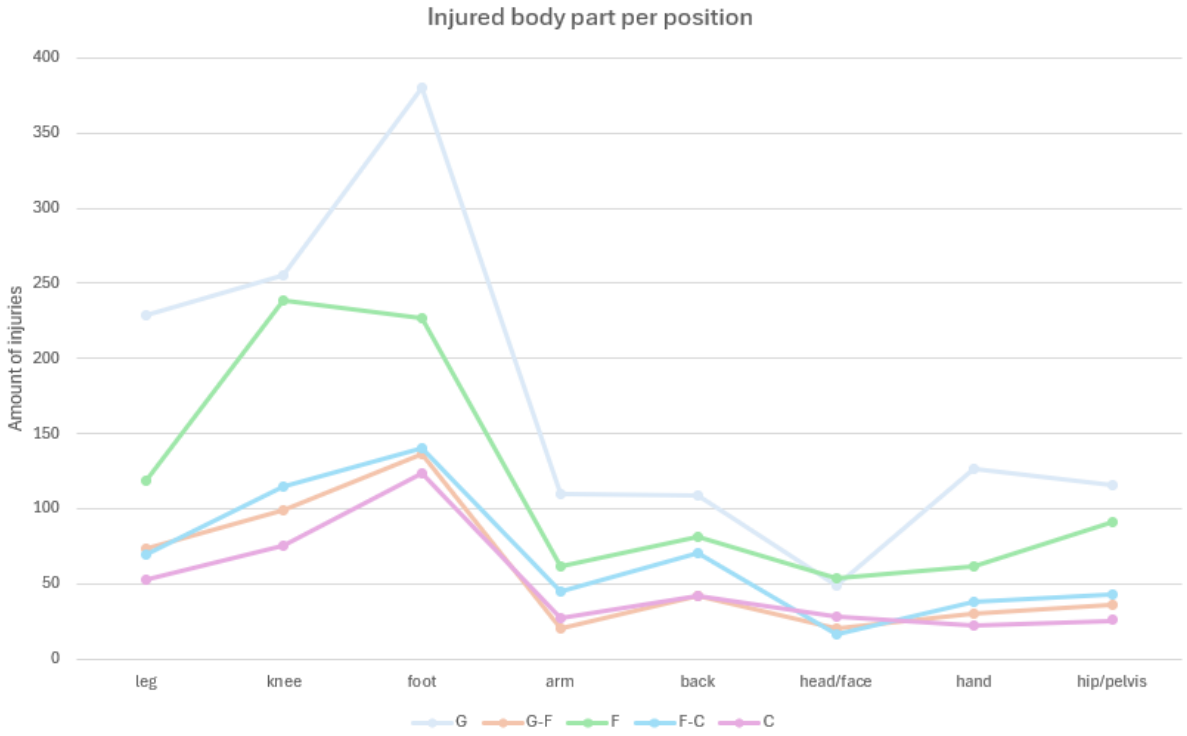


Figure 4.2: Injured body part by position.

Figure 4.2 is a combined line graph, which shows the amount of injuries per body part for each of the positions. These lines show us irregularities between positions in the increase and decrease of injuries when going from one body part to the other. As an example, one can see, that the proportion of knee injuries in Forwards is clearly higher than the proportions of knee injuries in other positions and is lower in guards.

1. **Guards (G):** Guards have significant deviations on the body parts knee (-4.20) and hand (+5.32), meaning that guards have a significant amount of less knee injuries than expected, which could be due to their generally better agility focusing specifically on the knee strength and flexibility. This agility comes from their main role of primarily attacking the basket and creating scoring opportunities by making quick movements, changes of direction and maintaining control when being defended aggressively and makes them being used to the type of movements which are the main reasons for knee injuries. Guards tend therefore more to using speed and agility training, like engaging in sprint intervals or using agility training ladders to enhance their acceleration and deceleration abilities, which are important in knee injury prevention. The elevated deviation in hand injuries is most probably related to their role as primary ball handler and passer, meaning they spend most time dribbling the ball and

making hard passes, which can make fingers and wrists vulnerable to injuries. Many hand injuries occur when not receiving the ball off a pass cleanly, leading to sprained or fractured fingers.

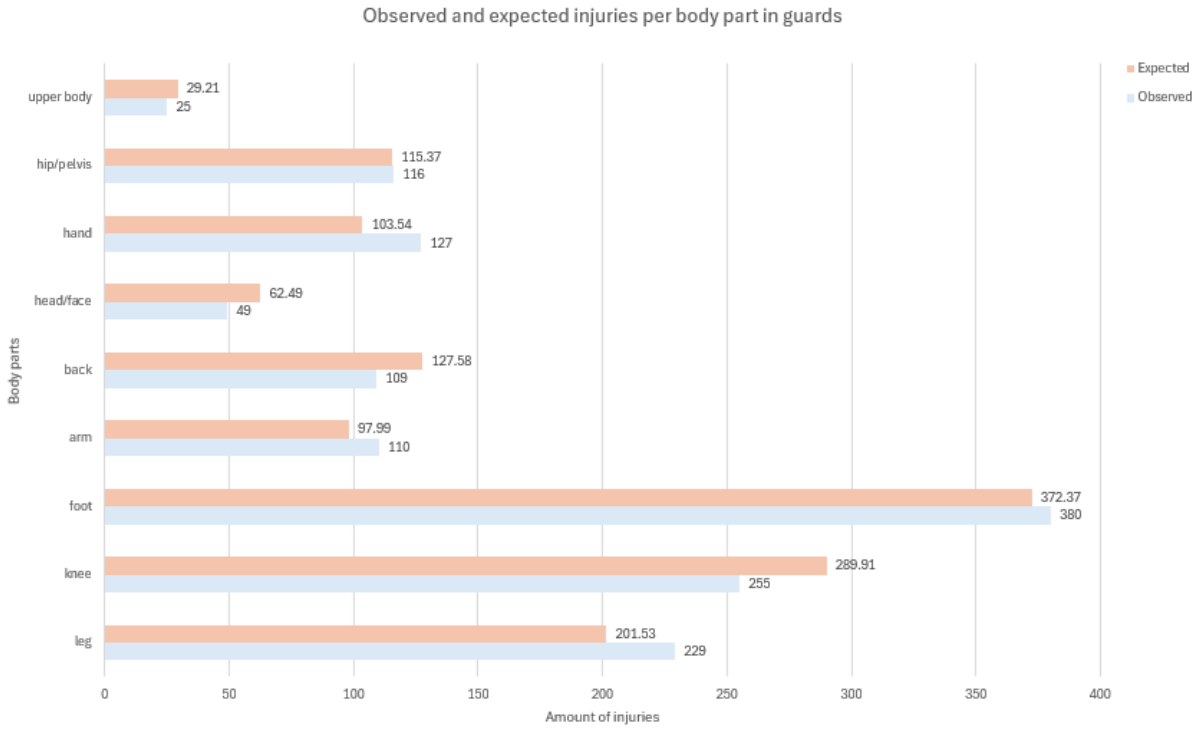


Figure 4.3: Compared observed and expected injury count per body part for guards.

Figure 4.3 is a bar graph comparing the amount of observed and expected injured body parts in guards, where the deviation between observed and expected injuries is recognisable.

2. **Forwards (F):** The main deviation across body parts for a forward is on the knees (+8.20), meaning that forwards are suffering much more from knee injuries than expected. This deviation can be explained with the frequent jumping and landing from forwards to get rebounds or with the fact that they are playing closer to the basket in a more crowded area. They often engage in physical battles in the paint which can have an impact on the knees through collisions and falls. In general, the forward position demands a high level of physicality, explosiveness and high jumps, which all put a lot of load on the knees. There is also a significant deviation for head and face injuries (+3.06), which is again explainable with the role of the forward position, engaging himself in crowded areas to fight for a rebound or

use physicality inside the paint to play defense or score closer to the basket, where there is a higher risk of getting hit in the head as there are many players around.

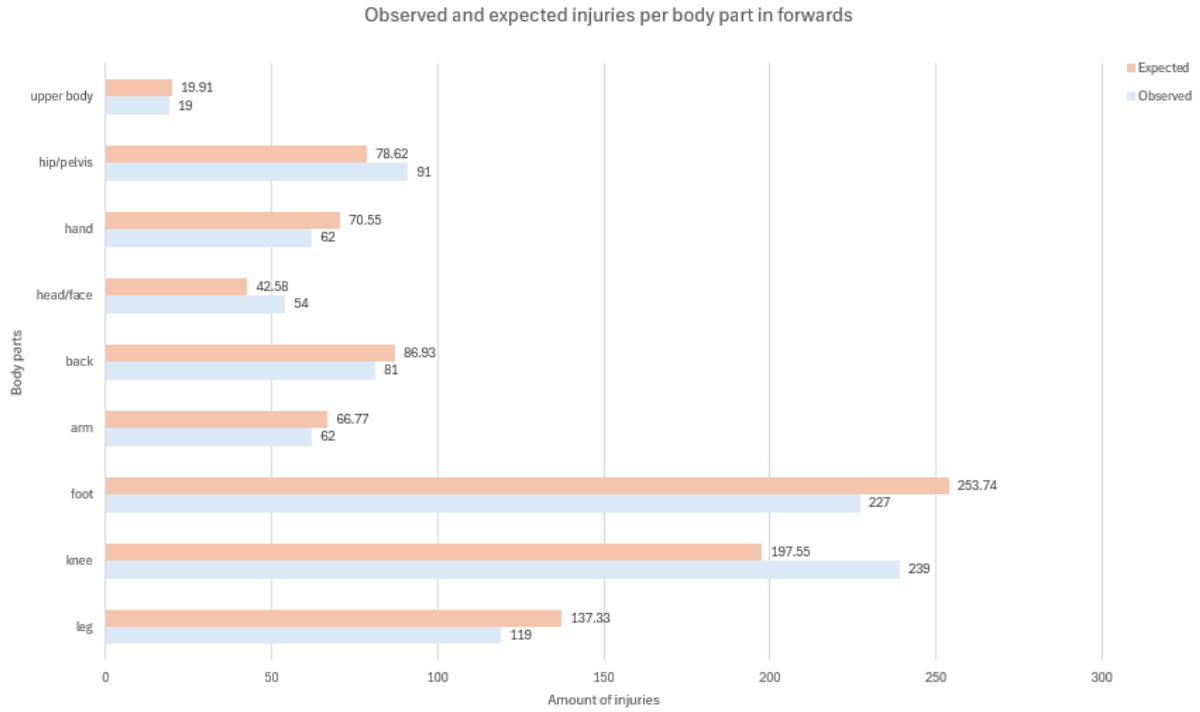


Figure 4.4: Compared observed and expected injury count per body part for forwards.

Figure 4.4 is a bar graph comparing the amount of observed and expected injured body parts in forwards, where the deviation between observed and expected injuries is recognisable.

3. **Centers (C):** Just like for the forward position, there is an elevated value in deviation for injuries to the head and face (+4.96) which can be explained with the same arguments as for the forward position and are even intensified for the centers. With a deviation of +3.36, centers are injured to the upper body more often than expected. This can be explained by post-play, where a center experiences a lot of contact by his defenders to gain advantageous position inside the paint to score under the basket. The defensive role of the centers puts an even bigger stress on the upper body, as they are often pushed and shoved by other centers inside, which constantly puts stress on the torso and the rest of the upper body.

   Figure 4.5 is a bar graph comparing the amount of observed and expected injured body parts in centers, where the deviation between observed and expected injuries is recognisable.

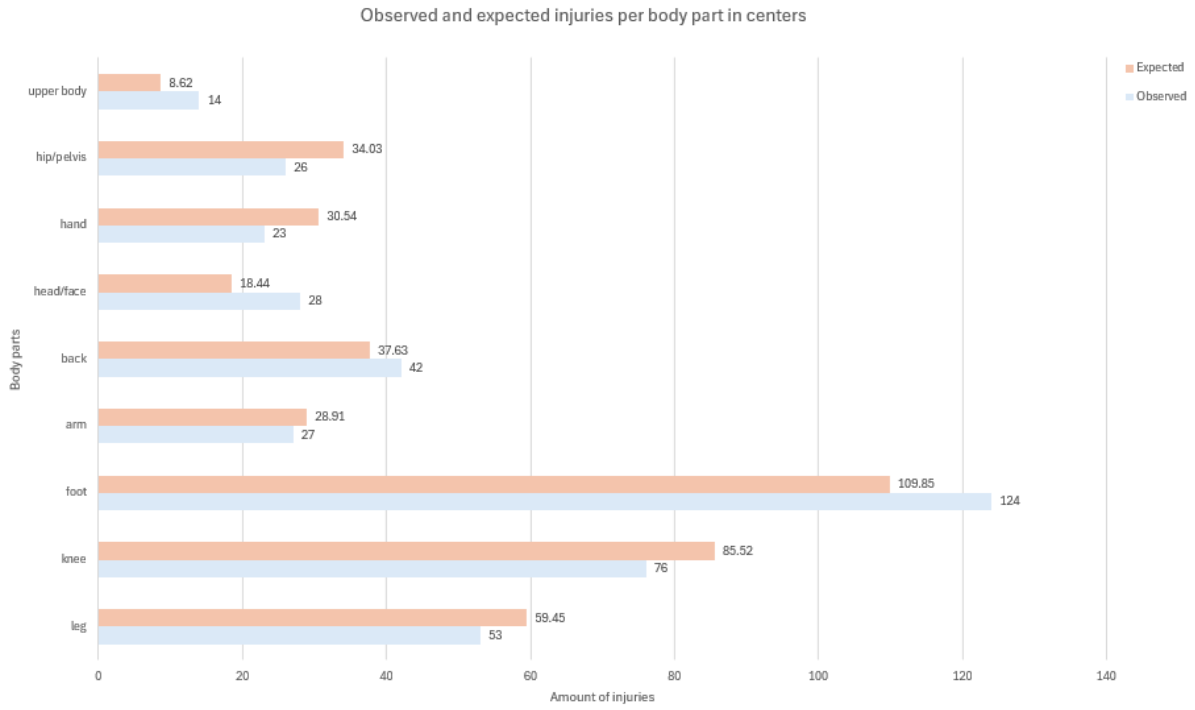


Figure 4.5: Compared observed and expected injury count per body part for centers.

4. **Guard-Forwards (G-F):** The reason there is a significant deviation in arm injuries (-4.22) for guard-forward is less straightforward compared to the deviation reasons for other positions. A possible reason for the fewer arm injuries than expected, could be the balanced workload, meaning that they balance between ball handling, shooting and driving, which leads to a variety of use of arms and reduces repetitive strains.

5. **Forward-Centers (F-C):** Forward-centers have more back injuries than expected, which could be due to their dual roles which require physicality in the paint but also explosiveness and could result in a too important physical workload to the back. However, the reason for this high deviation is not clear.

4.1.3 Relationship between position and injury type

In order to determine whether the position of a player has an impact on the injury type when a player gets injured, we have used the chi-square test of independence.

Check for data requirements:

- The 'Position' variable is a categorical variable containing 5 different categories and the

'injury_type' is a categorical variable with 14 different categories.

- No player is mentioned in more than one position category and no player has more than one injury type mentioned at the same time in the dataset.
- As we did not have a value of 1 in each cell for the 'abrasion', 'pain' and 'swollen' categories for every position, we have decided to remove those categories and remain with only 11 categories.
- After removing the categories mentioned above, we have 83.64% of expected values above 5 and all entries are at least equal to 2.72 (centers with a hernia), which is greater than 1.

We have all the data requirements necessary to perform the chi-square test.

Injury types per position			**Chi squared test**									
	broken	tear	strain	sprain	stress	inflammation	sore	dislocated	bruise	concussion	hernia	**Total**
G	89	85	169	278	8	45	301	14	101	23	16	1129
G-F	33	25	56	84	3	23	106	1	40	9	3	383
F	57	67	105	180	5	36	204	6	78	27	5	770
F-C	20	36	78	102	4	17	143	4	29	9	1	443
C	39	15	39	87	6	7	76	4	31	14	1	319
Total	238	228	447	731	26	128	830	29	279	82	26	3044
Expected	broken	tear	strain	sprain	stress	inflammation	sore	dislocated	bruise	concussion	hernia	**Total**
G	88.27	84.56	165.79	271.12	9.64	47.47	307.84	10.76	103.48	30.41	9.64	1129
G-F	29.95	28.69	56.24	91.98	3.27	16.11	104.43	3.65	35.10	10.32	3.27	383
F	60.20	57.67	113.07	184.91	6.58	32.38	209.95	7.34	70.57	20.74	6.58	770
F-C	34.64	33.18	65.05	106.38	3.78	18.63	120.79	4.22	40.60	11.93	3.78	443
C	24.94	23.89	46.84	76.61	2.72	13.41	86.98	3.04	29.24	8.59	2.72	319
Total	238	228	447	731	26	128	830	29	279	82	26	3044
Deviations	broken	tear	strain	sprain	stress	inflammation	sore	dislocated	bruise	concussion	hernia	
G	0.006	0.002	0.062	0.174	0.280	0.129	0.152	0.978	0.059	1.807	4.190	
G-F	0.312	0.474	0.001	0.692	0.023	2.952	0.024	1.923	0.683	0.168	0.023	
F	0.170	1.508	0.576	0.130	0.378	0.405	0.169	0.243	0.781	1.888	0.378	
F-C	6.185	0.239	2.577	0.181	0.012	0.142	4.083	0.012	3.316	0.721	2.048	
C	7.924	3.310	1.313	1.410	3.937	3.067	1.386	0.304	0.106	3.402	1.092	
Chi squared	68.509											
p-value	0.003											

Figure 4.6: Relationship between injury type and position.

Hypotheses:

- H_0: The player's position is independent from the injury type.
- H_a: The player's position is not independent from the injury type.

4.1.3.1 Results of the Chi-square test of independence

Chi-square (χ^2):
In this case we have a chi-square statistic equal to:

$$\chi^2 = 68.51.$$

Degrees of freedom:
There are

$$df = (5-1) \cdot (11-1) = 4 \cdot 10 = 40$$

degrees of freedom.
p **value:**
By using the *Excel* formula *CHISQ.DIST.RT(χ^2, df)*, we get a p value equal to

$$p = 0.003.$$

Interpretation:

- The high χ^2 indicates, that there is a significant difference between the observed and expected values.
- Since we get $p = 0.003$, which is < 0.05 and indicates the probability of getting $\chi^2 = 68.51$ under H_0, we can reject the null hypothesis and accept that there is an association between the position of a player and his injured body part. However, this association seems to be weaker than the association between injured body part and position.

4.1.3.2 Important observable insights by injury type

By analysing both the graph in Figure 4.7 and the tables in Figure 4.6, we can get insights on how the two variables are associated.
Most significant deviations are given at the centers position, meaning that there is a difference between the proportions of injury types in centers compared to the rest of the team. We can observe on Figure 4.7, that the lines corresponding to injuries of centers behave slightly different than the lines of the other positions.

- The highest deviation for broken bones and concussions is found in centers, with deviations of $+7.92$ and $+3.4$, which is most likely explainable through their intense physical play inside the paint, where they come across a lot of contact and can easily be hit in the head leading to

Figure 4.7: Injury type by position.

possible concussions. Guards in comparison have a negative deviation (-1.81), meaning they have less concussions than expected, probably because they most frequently are standing around the perimeter, with less people around. This physicality in centers can also lead to fractures like broken noses or broken fingers, but can also cause hard falls, leading to possibly bad fractures in feet or knees for example.

- Centers also have a slightly elevated deviation in sprains (+1.41), which could be explained by the amount of rebounding resulting in more awkward landings possibilities, or because the risk of stepping on someone's foot in the paint is higher compared to around the perimeter. However, centers also experience less tears than expected (-3.31), which could maybe be explained by their robust muscle mass, which protects them against muscle and ligaments tears.

- Centers appear to suffer more from stress reactions/fractures than expected (+3.93), which is likely due to their larger body mass, putting greater stress on their bones on impacts combined with their frequent jumping, can increase the risk of stress fractures.

The following bar graph illustrates the differences in observed and expected observations among centers:

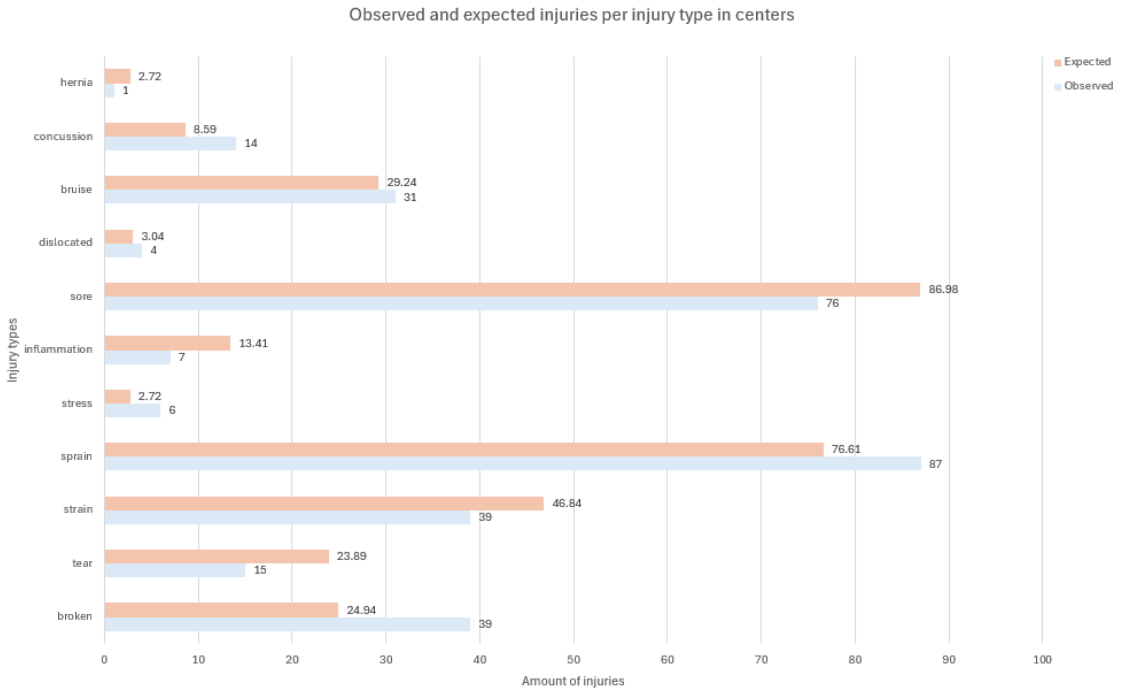


Figure 4.8: Compared observed and expected values injury count per injury type for centers.

4.1.4 Relationship between position and injury detail

In order to determine whether the position of a player has an impact on the injury detail when a player gets injured, we have used the chi-square test of independence.

Check for data requirements:

- The 'Position' variable is a categorical variable containing 5 different categories and the 'injury_detail' is a categorical variable with 8 different categories.
- No player is mentioned in more than one position category and no player has more than one injury detail mentioned at the same time in the dataset.
- As we did not have a value of 1 in each cell for the 'nerve' category for every position, we have decided to remove this category and remain with only 7 categories.
- After removing the categories mentioned above, we have 91.43% of expected values above 5 and all entries are at least equal to 3.35 (tissue injuries in centers), which is greater than 1.

We have all the data requirements necessary to perform the chi-square test.

Injury detail per position								
	bone	ligament	cartilage	muscle	tissue	joint	tendon	Total
G	126	295	31	275	7	127	66	927
G-F	54	87	9	92	4	22	31	299
F	82	183	40	173	6	78	37	599
F-C	31	107	10	97	9	56	20	330
C	69	86	8	60	4	26	18	271
Total	362	758	98	697	30	309	172	2426

Expected	bone	ligament	cartilage	muscle	tissue	joint	tendon	Total
G	138.32	289.64	37.45	266.33	11.46	118.07	65.72	927
G-F	44.62	93.42	12.08	85.90	3.70	38.08	21.20	299
F	89.38	187.16	24.20	172.10	7.41	76.29	42.47	599
F-C	49.24	103.11	13.33	94.81	4.08	42.03	23.40	330
C	40.44	84.67	10.95	77.86	3.35	34.52	19.21	271
Total	362	758	98	697	30	309	172	2426

Expected	bone	ligament	cartilage	muscle	tissue	joint	tendon
G	1.10	0.10	1.11	0.28	1.74	0.68	0.00
G-F	1.97	0.44	0.78	0.43	0.02	6.79	4.53
F	0.61	0.09	10.32	0.00	0.27	0.04	0.70
F-C	6.76	0.15	0.83	0.05	5.93	4.64	0.49
C	20.17	0.02	0.79	4.10	0.13	2.10	0.08
Chi squared	78.26						
p-value	0.0000001						

Figure 4.9: Relationship between injury detail and position.

Hypotheses:

- H_0: The player's position is independent from the injury detail.
- H_a: The player's position is not independent from the injury detail.

4.1.4.1 Results of the chi-square test of independence

Chi-square (χ^2):
In this case we have a chi-square statistic equal to:

$$\chi^2 = 78.26.$$

Degrees of freedom:
There are

$$df = (5-1) \cdot (7-1) = 4 \cdot 6 = 24$$

degrees of freedom.

p value:
By using the *Excel* formula *CHISQ.DIST.RT(χ^2, df)*, we get a p value equal to

$$p = 0.0000001.$$

Interpretation:

- The high χ^2 indicates, that there is a significant difference between the observed and expected values.
- Since we get $p = 0.0000001$, which is clearly < 0.05 and indicates the probability of getting $\chi^2 = 78.26$ under H_0, we can reject the null hypothesis and accept that there is a strong association between the position of a player and his injury detail.

4.1.4.2 Important deviations by position contributing most to the high chi-square value

The table in Figure 4.9 and the graph in Figure 4.10 suggest that the following observations contribute the most to the elevated chi-square. The combined line graph in Figure 4.10 illustrates these observations, as there are noticeable irregularities in some parts of the lines.

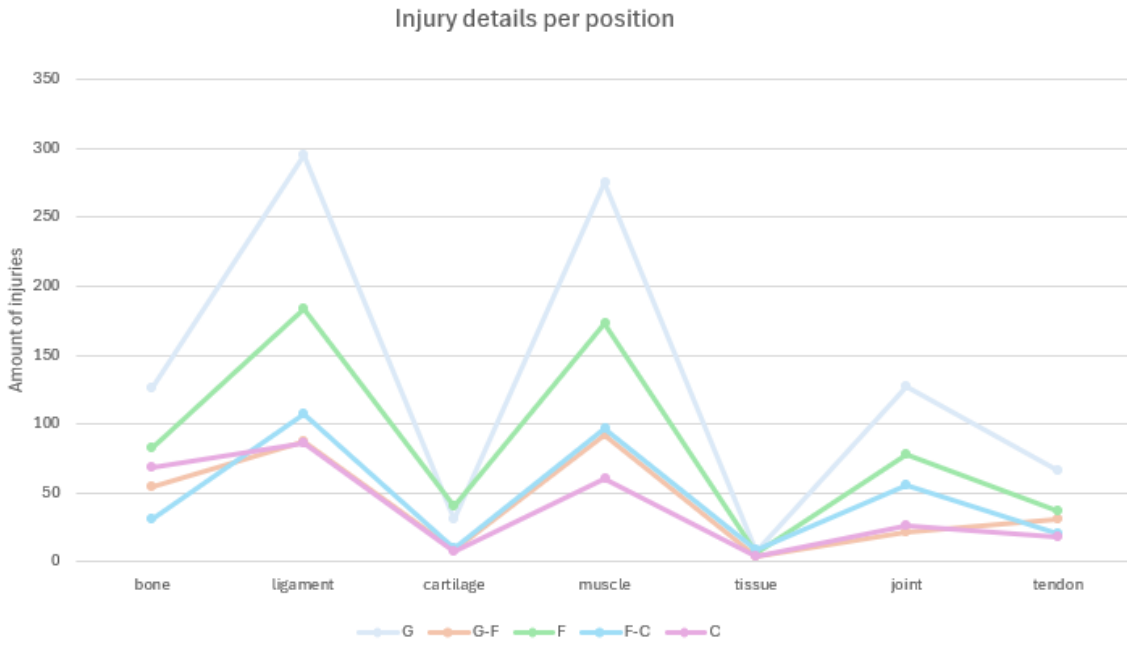


Figure 4.10: Injury detail by position

- The highest deviation is huge and observed in bones for centers (+20.17), which can be related and explained with the arguments mentioned for the injury type 'broken' in centers.
- The second highest deviation in the table is for cartilage injuries in forwards with a deviation of +10.32. As forwards already had a very high deviation in knee injuries, where cartilage injuries are common, such as meniscus injuries, we can interpret this deviation the same way we did for knee injuries in forwards.
- Muscle injuries occur less than expected in centers (-4.10), which is probably again relatable to their strong muscle mass.

4.2 Influence of age on injuries

```
#Correlation between age and injured body part/injury type/injury detail
df_injuries['body_part_encoded'] = LabelEncoder().fit_transform(df_injuries['body_part'])
df_injuries['injury_type_encoded'] = LabelEncoder().fit_transform(df_injuries['injury_type'])
df_injuries['injury_detail_encoded'] = LabelEncoder().fit_transform(df_injuries['injury_detail'])

# Calculate the correlation
correlation_1 = df_injuries['body_part_encoded'].corr(df_injuries['Age'])
correlation_2 = df_injuries['injury_type_encoded'].corr(df_injuries['Age'])
correlation_3 = df_injuries['injury_detail_encoded'].corr(df_injuries['Age'])

print("Correlation between 'body_part' and 'Age':", correlation_1)
print("Correlation between 'injury_type' and 'Age':", correlation_2)
print("Correlation between 'injury_detail' and 'Age':", correlation_3)
```

Figure 4.11: Code to calculate the correlations.

Code: The code in Figure 4.11 was created to calculate the correlation between the age of an injured player and his injured body part, injury type and injury detail. The code consists of the following steps:

- As the variables *'body_part'*, *'injury_type'* and *'injury_detail'* are categorical features, we needed to convert them into numerical features. To do so, we have used the *LabelEncoder()*, which converts every unique category into one unique number.
- As the variables are now all numerical, we can compute the correlation using the function *corr()*.

Results:

- **Correlation coefficient between 'body_part' and 'Age':** 0.05
- **Correlation coefficient between 'injury_type' and 'Age':** 0.01
- **Correlation coefficient between 'injury_detail' and 'Age':** 0.02

All of these correlation results are all < 0.1, meaning that there is only a negligible association between the age of an injured player and his affected body part, his injury type and his injury detail.

4.3 Influence of weight on injuries

Code: The code to calculate the correlation between the weight (in kg) of an injured player and his injured body part, injury detail and injury type, is the same as in Figure 4.11, but with 'Weight' instead of 'Age'.
Results:

- **Correlation coefficient between 'body_part' and 'Weight':** 0.01
- **Correlation coefficient between 'injury_type' and 'Weight':** -0.002
- **Correlation coefficient between 'injury_detail' and 'Weight':** -0.06

The absolute values of these correlation results are all < 0.1, meaning that there is also only a negligible association between the weight of an injured player and his affected body part, his injury type and his injury detail.

4.4 Relationship between all player variables and their injuries

4.4.1 Multiple class logistic regression

Definition 9. [6] *Multinominal logistic regression* is a classification model used to predict the probabilities of the different possible outcomes of a categorically distributed dependent variable with more than two categories, using values of a set of predictor variables, which can be of any form. It is an extension of the *logistic regression*, in which the outcome variable is restricted to two discrete categories.

Data: We have our independent variables $X_i = (x_{1,i}, ..., x_{M,i})$, where $x_{1,i}$ represents the first independent variable observed for the i^{th} player. If we have 5 different independent variables, $M = 5$. Our dependant categorical variable is given with the notation Y_i, which can take up to K different values.

Goal: The goal of multinominal logistic regression is to build a model, which recognises the relationship between the independent variables and the nominal outcome variable so it can predict the outcome when we collect new data of other players of whom we don‘t know the outcome yet.

Linear predictor: Multinominal logistic regression uses a linear predictor function $f(k, i)$ to predict that an observation i has an outcome $k \in \{1, ..., K\}$, given by

$$f(k, i) = \beta_{0,k} + \beta_{1,k}x_{1,i} + ... + \beta_{M,k}x_{M,i},$$

where $\beta_{m,k}$ with $m \in \{1, ...M\}$ is the regression coefficient, which represents the contribution of the associated independent variable to the likelihood of observing outcome k and $\beta_{0,k}$ is the intercept representing the mean value of the outcome variable when all of the predictor variables in the model are equal to zero. The regression coefficients and explanatory variables can be grouped into vectors of size $M + 1$ and we can rewrite the previous formula more compactly:

$$f(k, i) = \beta_k \cdot x_i,$$

where β_k is the coefficient vector associate with the outcome k, and x_i is the set of predictors for the i^{th} variable.

Equation: The multinomial logistic regression model to calculate the probability of an observation to be in the k^{th} category, is formulated as follows:

$$P(Y_i = k|X_i) = \frac{e^{\beta_k \cdot X_i}}{1 + \sum_{j=1}^{K-1} e^{\beta_j \cdot X_i}}.$$

Evaluation Metrics for Multinomial Logistic Regression:

1. **Accuracy:**

   Definition 10. *Accuracy* calculates the proportion of correctly classified observations out of the total observations:

   $$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = 1 - \frac{FP + FN}{TP + TN + FP + FN} = 1 - \text{ERR},$$

   where

- True positive (TP): observations correctly classified as belonging to a certain category;
- True negative (TN): observations correctly classified as not belonging to a certain category;
- False positive (FP): observations incorrectly classified as belonging to a certain category;
- False negative (FN): observations incorrectly classified as not belonging to a certain category.

The higher the accuracy, the better the model.

2. **Precision:**

   Definition 11. *Precision* calculates the proportion of correctly classified positive observations out of all observations classified as positive:

$$\text{Precision} = \frac{TP}{TP + FP}.$$

   A high precision value indicates few false positives and is useful to calculate if there are many false positives.

3. **Recall:**

   Definition 12. *Recall* calculates the proportion of correctly classified positive observations out of all actual positive observations:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

   A high precision value indicates few false negatives and is useful to calculate if there are many false negatives.

4. **F1 score:**

   Definition 13. The *F1 score* is the harmonic mean of precision and recall, providing a balance between the two metrics:

$$\text{F1 score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Code

```
#Logistic regression with 'body_part' as the dependent variable
df_LR_1= df_injuries[df_injuries['body_part'] != 'NBP'].copy()

X = df_LR_1[['Position ', 'Age', 'Weight', 'Team']]
y = df_LR_1['body_part']

preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), ['Age', 'Weight']),
        ('cat', OneHotEncoder(), ['Position ', 'Team'])
    ])

pipeline = Pipeline(steps=[
    ('preprocessor', preprocessor),
    ('classifier', LogisticRegression(max_iter=10000, multi_class='multinomial', solver='lbfgs'))
])
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=100)
pipeline.fit(X_train, y_train)

y_pred = pipeline.predict(X_test)
y_pred_proba = pipeline.predict_proba(X_test)

# Evaluation
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='weighted', zero_division=0)
recall = recall_score(y_test, y_pred, average='weighted', zero_division=0)
f1 = f1_score(y_test, y_pred, average='weighted', zero_division=0)
print("Accuracy:", accuracy)
print("Precision:", precision)
print("Recall:", recall)
print("F1 Score:", f1)
```

Figure 4.12: Multinominal logistic regression code.

The code given in Figure 4.12 is the code creating a multinominal logistic regression model to determine whether we can predict the injured body part, based on player information. This code can be reused with the injury types and details as dependent variables.

- We exclude the 'NBP' category as we do not have any information on injured body part for those observations. In the other two codes, we will exclude 'NIT' and 'NID' respectively, for the same reasons for which we delete 'NBP'.

- We define our independent variables X_i =('Position', 'Age', 'Weight', 'Team') and our dependent outcome variable as Y_i = ('body_part'). For the other two codes we would have the same predictor variables but would define our outcome variable as Y_i = ('injury_type') and Y_i = ('injury_detail').

- For the preprocessing of the features, we use a *'ColumnTransformer'*. Our numerical features, such as 'Age' and 'Weight' were standardized with the *'StandardScaler'* and our categorical

features 'Team' and 'Position' were encoded using the *'OneHotEncoder'*.

- We constructed a pipeline to ensure that the data transformations and the model are applied consistently and repeatedly. The pipeline includes the preprocessor defined in the previous step and the multinomial logistic regression classifier, with a maximum of 10,000 iterations and the *'lbfgs'* solver.
- We split our dataset into test (20%) and training (80%) sets.
- We fit the pipeline constructed to the training set.
- We make predictions on the test set and calculate the predicted probabilities.
- We evaluate the model created by computing the accuracy, precision, recall and F1 score for which we take the weighted average because of class imbalances. The *'zero_ division=0'* parameter ensures that 0 is used when there are no true samples, avoiding undefined metric warnings.

4.4.2 Logistic regression to determine the injured body part

We have used the multinominal logistic regression to determine how accurately we can predict an injured body part, based on information we have on a player, namely his team, position, weight and age. In order to understand this accuracy we have computed 4 different evaluation metrics:

- **Accuracy: 28.61%** The accuracy score means that 28.61% of predicted injured body parts, were correctly classified among all predictions made.
- **Precision: 19.27%** When the model predicts the injured body part, it is correct in 19.27% of the times, meaning that there are many false positives. In other words, the model often predicts a certain category to be injured, when in reality it isn't.
- **Recall: 28.61%** Due to the fact that we have no negatives in our dataset, as we only have players who have been injured and can therefore only have true and false positives, our recall score is calculated by the same formula as for the accuracy.
- **F1 Score: 20.24%** The F1 Score provides us with a measure, taking both the precision and recall into account to evaluate the model's performance. As our F1 Score is only 20.24%, we conclude that its performance is quite weak.

Even if the relationship is not strong enough to make predictions on player's future injured body parts, we can say that there is some association between our variables and the injured body part.

4.4.3 Logistic regression to determine the injury type

We have also used the multinominal logistic regression to determine how accurately we can predict an injury type, based on the same information we have on a player. In order to understand this accuracy we have computed the same 4 different evaluation metrics as for the injured body part:

- **Accuracy: 36.09%** The accuracy score means that 36.09% of predicted injury types were correctly classified among all predictions made.
- **Precision: 24.39%** When the model predicts the injury type, it is correct in 24.39% of the times, meaning that there are many false positives. In other words, the model often predicts a certain injury type to occur, when in reality it didn't.
- **Recall: 36.09%** Our recall value is the same as the accuracy for the same reasons mentioned for the body part variable.
- **F1 Score: 27.37%** The F1 Score takes both the precision and recall into account to measure and evaluate the model's performance. As our F1 Score is only 27.37%, we conclude that its performance is quite weak, but it stronger compared to the model we used to predict an injured body part.

Even if the model performs weakly to make predictions based on the variables we have, we can say that there is a stronger connection between our 4 predictor variables and the injury type than there was between the predictor variables and the injured body part. This observation was also made when we analysed the relationship between position of a player and his injured body part, injury type and detail, where the connection was stronger for the injury type compared to the injured body part.

4.4.4 Logistic regression to determine the injury detail

Lastly, we have used the multinominal logistic regression to determine how accurately we can predict an injury detail, based on the same information we have on a player. In order to understand this accuracy we have computed the same 4 different evaluation metrics as for the injured body part and the injury type:

- **Accuracy: 39.16%** The accuracy score means that 39.16% of predicted injury details were correctly classified among all predictions made.

- **Precision: 28.16%** When the model predicts the injury type, it is correct in 28.16% of the times, meaning that there are many false positives. In other words, the model often predicts a certain injury to the musculoskeletal system, when in reality another part of the system was injured.
- **Recall: 39.16%** Our recall value is the same as the accuracy for the same reasons mentioned for the body part variable.
- **F1 Score: 31.4%** Our F1 Score is 31.4%, meaning that the performance of the model is quite weak, but it stronger compared to the model we used to predict an injured body part and an injury type.

The model has a stronger performance when considering the *'injury_ detail'* variable as our outcome variable, however it is still low. The relationship between the predictor variables and the injury details could therefore be stronger than the relationship between the independent variables and the other outcome variables. This observation was also made when we analysed the relationship between position of a player and his injured body part, injury type and detail, where the connection was the strongest for the injury detail.

Conclusion

Possible reasons for the low performance score:

- We have an insufficient amount of features available to use to make predictions. Many more variables come into play, when analysing why players get injured and how. It would have been useful to have variables such as where a player got injured (practice, game,...), how many minutes he plays in games, rest days, lifestyle variables (sleep, nutrition, health,...), training load, bio-metrics, etc.
- Some body parts are significantly less injured compared to others, leading to class imbalances.
- Many injuries in basketball are contact injuries, which are very difficult to be predicted, as contact injuries can happen to every player.

Relationships: Our 4 independent variables have an impact on our 3 different categorical dependent outcome variables, but these associations are not strong enough to make predictions on the outcome variables based on only these 4 features.

Chapter 5

Teaching

As I mentioned in the introduction chapter, my professional goal is to become a high school mathematics teacher. I am therefore highly motivated in creating interesting activities, as well as answering to the frequent question of students 'Where will I need mathematics when I leave school?'. As most students like sports, this project is a good way to show them how mathematics can be used in a domain they like and how it can change the life, success and health of players. Some notions in mathematics can be introduced in an interesting way through this project and also proves their utility

5.1 Introduction to data

1. **Introducing the different types of data:** After defining the notions of 'discrete numerical variables', 'continuous numerical variables', 'nominal categorical variables' and 'ordinal categorical variables', a teacher could give his students a sample of our dataset and make them analyse the type of data for each feature.

2. **Collection of data:** In order to give an example to students on how we collect data, a survey could be made in the classroom to collect everyone's favourite sport.

3. **Frequency table:** By providing the students with a sample of the dataset, they could create a frequency table for a specific feature, which is the first step to descriptive statistics.

5.2 Activity to introduce statistics and visualisation

1. **Introduction to descriptive statistics:** In order to do this introduction, one could split the class in groups and give each group a different variable to calculate the mean, median and/or mode. For example, one could give to one group a frequency table of injuries per weight range and they would have to compute these three values and determine what they

think is the value representing the dataset best. To another group, we could give a frequency table of injuries per age range, with ages from 18 to 42. The same could be done by calculating the mode for injuries per team, injuries per year, injuries per month, injuries per position and so on.

2. **Data visualisation:** In the same groups with their same table, we could ask the students to make either a bar graph or a histogram, based on the type of data they have. In these graphs, they could add the mean, median and mode by drawing 3 different lines in 3 different colors.

3. **Presenting results:** The groups would then describe their results in front of the class, which would make them work on the way they present and explain their results and also improve their critical thinking, as they would have to try to interpret those results and understand if they make sense.

This type of activity could be done for students in the third year of secondary education (15-16 year olds).

Example of activity: Based on the following frequency table,

Age ranges	[18,21)	[21,24)	[24,27)	[27,30)	[30,33)	[33,36)	[36,39)	[39,42)	**Total**
Frequencies	127	638	899	811	585	255	96	25	3436

1. Compute the mean, median and mode for the injuries by age range.

2. Visualise the data using a convenient graph.

3. Interpret the results.

Reading graphs: Similarly, instead of giving them a frequency table, one could provide them with a graph, which they would have to try to read and create a table based on the graphs. Instead of calculating mean, median and mode based on the list they would have to calculate those values based on the graphs by creating a table first. This is a variation of the activity mentioned above and could be done for the same age group.

5.3 Introduction to proportion and percentages

In the same way, we could give them a frequency table and make them calculate the percentages of each category or range. For example, we could ask the students to calculate the percentages of injuries for each body part by providing them a frequency table of injuries per body part. We could do the same for injury types and details, for injuries per year and month, for injuries by team, injuries by position, but also injuries by age and weight ranges. Like the previous activity, students could visualise their results and present them in front of the class. This type of activity could be done for students in the first year of secondary education (13-14 year olds).

Example of activity: Based on the following frequency table,

Body parts	foot	knee	leg	back	hip/pelvis	arm	hand	head/face	upper body	NBP	**Total**
Frequency	919	627	513	328	285	222	221	136	117	68	3436

where 'NBP' (No body part) represents all the injuries for which we do not know which body part is injured.

1. Calculate the percentage of injuries by body part.
2. Visualise the percentages in a convenient graph.
3. Interpret your results.

Chapter 6

Conclusion

This thesis mainly focuses on analysing which injuries are most frequent in the NBA from 2010 - 2018, while also figuring out patterns and relationships between the players and their respective injuries, giving basketball coaches insights in how they can enhance player and team performances, while minimizing the injury risk for an individual player.

Key results: A notable observation was made that there is a peak in injuries at the end of the regular season, which is most likely related to the player's physical fatigue. The most common injured body parts are in the lower extremities, namely the feet, knees and legs, which together make up almost 60% of all injuries in the NBA. Soreness and sprains are the leading injury types in NBA players, while in 64,83% of injuries we know the type of, ligaments or muscles were injured. When analysing the relationships between player characteristics and their respective injuries, we have come to the conclusion that age and weight alone do not influence the injuries NBA players are suffering. However, when looking at the player's positions and the injuries they are suffering, we have noticed and proven through the chi-square test of independence, that there is a strong relationship between position and injuries. By examining injuries based on player positions, it was found that guards were more susceptible to hand injuries, forwards to knee injuries, and centers to upper body and head injuries, all of which can be attributed to their specific roles on the court. By applying multinominal logistic regression, we can come to the conclusion that all player characteristics together have an impact on our injuries, however these associations are not strong enough to be able to make injury predictions based on only the few features that we have.

Implications for basketball coaches: This distinction in injuries by position emphasizes the need for training and conditioning programs specific to each position. By customizing exercises and precautions based on the specific weaknesses of each player, coaches may be able to lower the frequency of these typical injuries. As a basketball coach, I would implement agility and strengthening exercises for ligaments and muscles for every player into my practice sessions, especially

around the knees and ankles, as these body parts are the most affected for every player, but I would pay an extra attention to strengthening the knees in forwards as they are more likely to injure knees than other positions. I would also focus on core and upper body strengthening in centers to stabilize the upper body and be more resistant in contacts and collisions, but would also implement post play techniques in order to have a proper positioning to help reduce their head injuries. Coaches should also teach players to be aware of their environment and surroundings while being on the court in order to prevent accidents.

To conclude, one can say that understanding the patterns and causes of different injuries through statistics and data analysis allows coaches and staff members to make informed decisions in training, prevention and player management, which will contribute to the health and success of players and basketball teams.

Bibliography

[1] "A deep learning approach to injury forecasting in NBA basketball". In: *Journal of Sports Analytics* 7.4 (2021). URL: https://content.iospress.com/articles/journal-of-sports-analytics/jsa200529.

[2] "Can we really predict injuries in team sports?" In: *Bulletin of the Spanish Society of Statistics and Operations Research (BEIO)* 38.3 (2022). URL: https://www.seio.es/beio/can-we-really-predict-injuries-in-team-sports/.

[3] *Chi-Square Test of Independence*. Kent State University Libraries. URL: https://libguides.library.kent.edu/SPSS/ChiSquare.

[4] Hans Eetvelde et al. "Machine learning methods in sport injury prediction and prevention: a systematic review". In: *Journal of Experimental Orthopaedics* 8 (Apr. 2021), p. 27. DOI: 10.1186/s40634-021-00346-x.

[5] Tim J Gabbett. "Incidence, severity, and cost of injuries to junior level middle-distance and long-distance runners: a prospective study". In: *Open Access Journal of Sports Medicine* 7 (2016), p. 117. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5094845/.

[6] *Multinomial logistic regression — Wikipedia, The Free Encyclopedia*. URL: https://en.wikipedia.org/wiki/Multinomial_logistic_regression.

[7] Apurva Singh. *NBA Players data (1950 to 2022)*. URL: https://www.kaggle.com/datasets/blitzapurv/nba-players-data-1950-to-2021.

[8] *Statistics for Machine Learning Tutorial*. Simplilearn. Nov. 2023. URL: https://www.simplilearn.com/tutorials/machine-learning-tutorial/statistics-for-machine-learning.

[9] Shaun Turney. *Chi-square test of independence*. Scribbr. 2023. URL: https://www.scribbr.com/statistics/chi-square-test-of-independence/.

[10] Jason Zivkovic. *Extensive NBA Injuries Deep Dive EDA*. URL: https://www.kaggle.com/code/jaseziv83/extensive-nba-injuries-deep-dive-eda.