Using Injury-Risk Forecasting to Quantify Financial Impact in the NBA

Ethan Wang

Abstract

Injuries in the NBA have become consequential not only for team success but for the financial costs those teams suffer. This study develops a machine learning framework that predicts next-game injury risk using publicly available box-score data, player attributes, and injury history, then translates these probabilities into expected financial costs. Combining five datasets from 2010-2022, I derived sixteen workload and recency features and trained a Random Forest model optimized with five-fold cross-validation. At a 2% threshold for classification, the model predicts out-of-sample 69% of injuries while correctly ruling out 62% of healthy games, indicating better-than-chance predictive power is possible using solely public data. Feature-importance analysis identified workload shifts and rest as primary predictors. Extending beyond prediction, this study gives a new way to interpret the financial implications of injuries, looking at how strategic rest decisions can minimize financial loss. This study offers NBA organizations a data-driven tool linking injury prevention with financial optimization, bridging injury forecasting with economic decision-making.

1. Introduction

Injuries have been a longstanding problem in all of sports, but in the NBA their impact stands out dramatically. With smaller rosters and superstars carrying a tremendous share of responsibility, the loss of one player can upend an entire season. Unlike football or baseball, where depth and roster size provide cushion for absences, a single injury to a team can swing playoff odds, alter franchise direction, and dramatically weaken league ratings. In the 2024 NBA season, for instance, teams like the 76ers and Pelicans were hit particularly hard with injuries and saw their postseason hopes vanish. These losses don't just hurt on-the-court performance, but can wreck teams financially, with over $350 million being spent on injury-related costs throughout an NBA season (Smith, 2016). Over time, staying ahead of NBA injuries isn't just about player health and safety, it's the key to a competitive edge against others.

In the past, teams have approached injury prevention rigorously, using machine learning to analyze both publicly available data, like game statistics, along with data from wearable technologies, like heart rate or step count (Dowsett, 2022). While prior research on forecasting injuries using machine learning models have focused primarily on identifying injury odds, this study extends that work by translating predicted injury probabilities into expected financial costs, giving teams a quantitative framework to assess health and monetary risk. In doing so, this research connects performance analytics with financial optimization, an area largely unexplored in current sports injury-forecasting literature.

I set out to answer a practical question for NBA front offices: Can a machine learning model that combines regular box-score data, and player attributes effectively predict whether a player will miss the next game with an injury– and if so, how can these predictions be used to estimate the financial cost of injuries to NBA teams? To do this, I merged five public datasets (injury logs 2010-2022, game-level box scores with minutes played, season-level box score and player attribute descriptions, team-level box scores per game, and player salaries) into a dataset where each observation is unique to a game-player. I engineered and derived 16 workload and recency metrics, treated missing values, and trained a Random Forest classifier for predicting injuries. To optimize my model, I utilized five-fold cross-validation to guide hyper-parameter tuning and performance estimation.

To preview my results, I found that simple box score data can be useful for successfully predicting injuries. The model predicts 69% of injuries while correctly ruling out 62% of healthy games. In practice, this means I can generate early warnings for over two-thirds of forthcoming injuries, giving teams a powerful tool to minimize injury odds. Additionally, using the model's logit injury probabilities, I demonstrate a framework to give teams financial insight into the benefits of resting players who are at high risk of injury, and show that this can be used to save upwards of $5.7M across teams in the NBA if optimized.

1.1 Literature Review

In the past, several studies have researched injury prediction in professional basketball. Cohan, Schuster, and Fernandez (2021) forecasted injuries using a deep learning model with injury history and game activity logs. They found that their model can learn to create meaningful features as a combination of raw features to predict injuries. In doing so, their model achieved 93.4% accuracy, with a Receiver Operating Characteristic (ROC) Area Under the Curve (AUC) of 0.80. Their research highlights the severe class imbalance within injury datasets, noting that a model predicting every case as a non-injury would still achieve approximately 98% accuracy. Charest et al. (2021) studied the effect of distance and direction of back-to-back games in the NBA, ultimately finding that specific travel patterns worsen recovery and performance. Although my study doesn't include travel distance between games, I do consider related metrics, such as days in between games or back-to-back games–

measured by the rest variable. Lu et al. (2022) focused on analyzing lower-extremity muscle strains (LEMSs) within NBA injuries from 1999 to 2019. They compared performance across different classification models trained on NBA injury data, finding that the best predicting machine learning algorithm for predicting LEMs was XGBoost. They identified that pre-existing injury history helped best predict LEMs. Chan et. al (2024) conducted a systematic review on the relationship between workload spikes and injury risk. Accumulating evidence over 11 studies, they found that training load was correlated with injury risk, highlighting the importance of including workload variables inside ML prediction models.

While Charest et al. (2021) and Lu et al. (2022) looked at specific drivers of injury, my research utilizes a wide array of publicly available data for injury prediction, similar to Cohan et al. (2021). Unlike prior studies, however, my study extends beyond prediction to include a cost-related threshold evaluation that weighs the consequences between false positives and false negatives. Additionally, I use an expected cost framework to identify the financial burden of player injuries, giving new insights into the economic dimension of injury prediction.

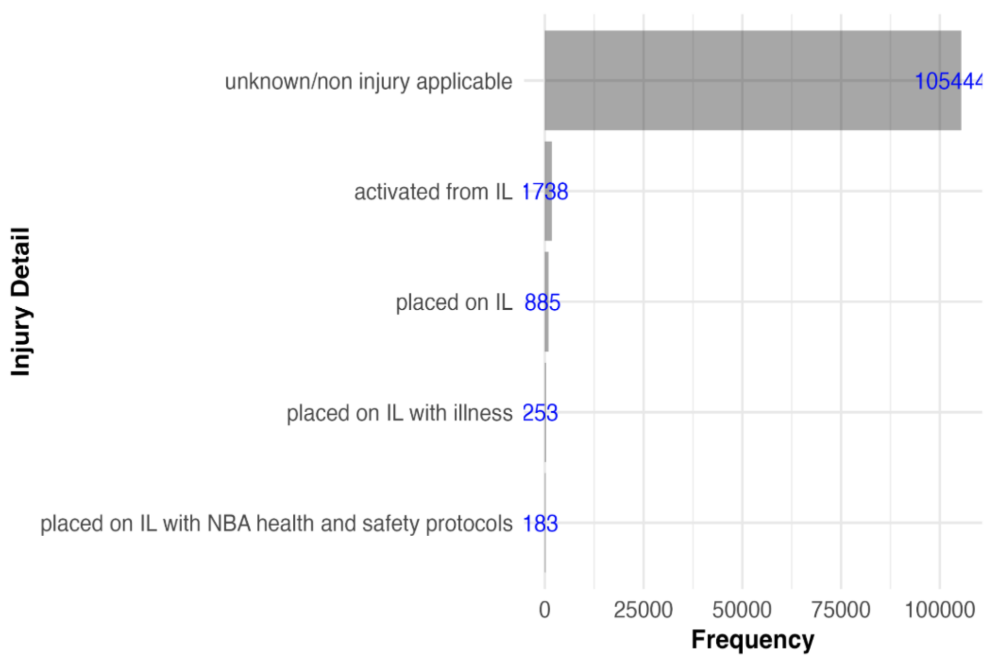
2. Material

This study utilizes multiple publicly available datasets from Kaggle to conduct the analysis. Together, these sources provide injury history, player-level workload, anthropometric information, and team-level game context.

2.1 Injuries Dataset

This study uses the nba-injuries dataset from Kaggle (Hopkins, 2018). The dataset consists of public injury reports and game summaries, covering detailed information about player injuries across ten NBA seasons (2010–2020). The dataset includes fields like the date of the injury, the player who got injured, and the type of injury. This dataset is the foundation for the injury prediction variable in the study. Using this dataset, I identified who was injured and the type of injury that was suffered. In Figure 1, I show the five most frequent injuries reported in the dataset. Because the majority of these injuries are reported as "unknown" type, I constructed a binary injury label that groups all injuries together (i.e. "injury next game: yes/no"). For more details about "known" injury types, see Figure S1 where I sort known injuries by frequency and severity.

Figure 1: Top 5 most frequent injuries within dataset



Certain types of injury labels are outside the scope of my prediction model: illness and infection, health and safety protocols, load-management and conditioning, personal, legal, and administrative (considered to be injuries because they are still logged on the IL). Therefore I do not include them as injuries in forecasting. If a player has an injury detail that corresponds to the following values, the injury indicator will be counted as 0, instead of 1. I decided against dropping them from the dataset, because they still provide useful game-level information and help preserve the continuity of player records. The study acknowledges that this means there will be significantly more non-injuries than injuries in the dataset, and will talk about this in the limitations section.

2.2 Team Statistics by Game and Season Dataset

The NBA Traditional Stats dataset from Kaggle compiles team-level box score statistics across multiple NBA seasons (Jóźwiak, 2024). For this study, I used the final team scores for each game to craft close game indicators in each game. The purpose of this feature was to capture game intensity, under the hypothesis that players who regularly play in tightly competitive games may have higher physical stress and therefore a higher injury risk.

2.3 Player Attributes

The NBA players dataset from Kaggle contains biometric, biographic, and basic box score data from 1996 to 2022 (Cirtautas, 2023). I use variables such as *height*, *weight*, and season averages per player to look at whether player attributes change injury likelihood odds.

2.4 Player Stats

The NBA Game Details dataset is a player-level dataset that contains useful box-score related metrics (Lauga, 2020). From this dataset, I use the “minutes played” variable, which is the minutes and seconds that a player plays per NBA game. I used the minutes played to construct the following features: (1) *avg. minutes (last 5 games)*, (2) *change in minutes since last game*, (3) *avg. high minute streak (last 20)* and (4) *high minute games in the last 20. Avg. minutes (last 5 games)* measures the mean amount of games in a player’s last five games, which provides a short term glimpse of a player’s recent playing time. *Change in minutes since the last game* provides an understanding of how a player’s current game compares to the last game, with sudden workload changes having a dramatic impact on injury odds. *Avg. high minute streak (last 20)* measures the average length of consecutive-game stretches, within a player’s last 20 games, where they played heavy minutes (above 35 minutes). In other words, it provides an understanding of how often and how long a player sustains extended workloads without a break, highlighting patterns of accumulated risk. *High minute games in the last 20* reflects how much games in a player’s 20 most recent games are of heavy minutes (above 35 minutes). All of these variables potentially signal workload spikes which may impact risk of injury.

2.5 Player Salary

The NBA Player Stats and Salaries 2010-2025 dataset is a player-level dataset that contains both box-score data and details on a player’s salary (Ratin21, 2025). From this dataset, I will be extracting the salaries for an understanding of the financial cost of injuries. The distribution of player salaries in the NBA is skewed right, with the league minimum being the lowest possible salary and super-max contracts being some of the highest. Throughout the years, contracts have progressively climbed because of the increase in salary cap and inflation.

3. Preprocessing the Datasets

3.1 Merging the Datasets for Modeling

To construct the dataset that my model uses, I merged across all previous datasets by player-game.

3.2 Feature Engineering in The Merged Dataset

An indicator for close basketball games is included to measure how game intensity may affect injury odds. If a game is closer, is the player playing harder? Could this put a higher demand on their body? To add an indicator for close games, I have to consider multiple factors. The NBA considers a close game as a game where the point differential is confined within a 10 point margin before the start of the fourth quarter and narrows down to a 5 point or less disparity at the end of the game. For the sake of simplicity and because I don’t have access to the score of the game at the start of the fourth quarter, I will be considering close games as games with a point differential of 5 points or less by the end.

I used minutes played and prior player performance statistics to engineer a series of workload and recency variables. First, I created binary indicators for high-minute games (>30 minutes) and mid-minute games (>23 minutes), and then calculated streaks of consecutive occurrences in each respective category. From there, I calculated rolling metrics over a player's last 20 and last 5 games, including the number and average length of high- and mid-minute streaks within trailing games. I also added short-term features that capture workload and recovery such as *change in minutes since last game, days of rest,* and *days since last injury*. To observe a given player's injury history, I included season-to-date injury counts and total career injuries. Finally, I incorporated previous-season averages (rebounds, 3-point attempts, free-throw attempts, and minutes) to provide an understanding of player tendencies.

3.3 Cleaning the Datasets for NA and Filling in Values

Some features contained missing values, which could interfere with the quality of my modeling fits. I resolved these missing values in the following ways:

1. Categorical fields.
   The final dataset contains a variable called *Relinquished*. In the context of my study, this is a team transferring a player to the injured list. In games where no player is transferred, *Relinquished* cannot be meaningfully interpreted; therefore I replaced NA entries in *Relinquished* with the string "unknown".
2. Numerical box-score statistics and recovery metrics (mean imputation).
   16% (22631/140879) of my observations had NA values in box-score related statistics., because omitting NA values in the dataset for box-score statistics leads to significant data loss, for conventional game-level performance figures–e.g., three-point and free-throw counts and percentages, rebounds, and the *rest* variable–I first used each player's own seasonal mean wherever at least one non-missing value existed. If an athlete had no

observed data in a given column, I substituted the league-wide mean. Following common practice in sports workload analysis (see Benson et al., 2021), I used each player's seasonal mean wherever a non-missing value existed; if none existed, I substituted the league-wide mean. For missing *age* values, I first looked for if the player had any previous existing *age* in other years, and attempted to use the difference in seasons as either an addition or subtraction to calculate a missing *age* value. If the player didn't have any preexisting *age* values in the dataset, I used the overall mean.

3. Streak, recency, and workload indicators (median imputation). Variables that are inherently skewed–such as streak magnitudes *(Last high-minute streak length*, *Avg. high-minute streak (last 20),* etc.), workload counts (*High-minute games in last 20*, *Avg. minutes (last 5 games)*), and recency measures (*Days since last injury*)–were imputed with the within-player median to mitigate the influence of outliers. This approach is described as appropriate for skewed data (Mohammed et al., 2021). As with the mean strategy above, I fell back on the overall-sample median only when a player was missing all previous values.

3.4 Final Dataset

The final dataset contains 21 total variables, where each observation is identified by a unique game-player. See Table 1 for details. The data spans from 2012 to 2023 and includes 8253 unique games and 1211 unique players. On average, players played 20.14 minutes per game (SD = 12.60), with an injury rate of 0.03 (SD = 0.17). Players typically had around 16 games of *rest* between games (SD= 56.34), with a total of 9151 games where a player played on a back-to-back (one day of rest). While the dataset's mean *rest* time is 16 days, this value is skewed by the significant number of low-minute or inactive players (as seen by the median of 4). The typical number of days separating a player and his last injury is 254.49 (SD = 312.23).

Table 1: Descriptive Statistics

variable	mean	# sd	# min	# median	# max
Identifiers and Context					
Gameid	22,627,242.79	4,222,721.43	21,100,001.00	21,701,180.50	52,100,131.00
Season	2,017.96	3.15	2,012.00	2,018.00	2,023.00
Injury Outcomes and History					
Rest	16.37	56.34	1.00	4.00	2,540.00
Days since last injury	254.49	312.23	1.00	113.00	3,105.00
Injuries so far this season	0.70	1.21	0.00	0.00	14.00
Total career injuries before today	5.70	5.87	0.00	4.00	48.00
Injury next game (yes / no)	0.03	0.17	0.00	0.00	1.00
Minutes and Workload Variables					
Minutes Played	20.14	12.60	0.00	21.68	60.12
Close games in last 20	5.29	2.13	0.00	5.00	15.00
Avg. high minute streak (last 20)	0.89	1.79	0.00	0.15	27.50
High minute games in last 20	5.30	5.80	0.00	3.00	20.00
Avg. minutes (last 5 games)	20.25	10.55	0.00	21.30	45.66
Change in minutes since last game	-0.06	10.09	-48.00	0.00	48.62
Player Characteristics					
Age	27.57	4.18	19.00	27.00	43.00
Player height (cm)	200.40	8.74	165.10	200.66	228.60
Player weight (kg)	100.11	11.55	60.33	99.79	141.07
Game Statistics (Previous Season Averages)					
Prev. season rebounds / game average	4.13	2.43	0.00	3.58	17.42
Prev. season 3-point attempts / game average	2.57	2.12	0.00	2.32	13.18
Prev. season free-throw attempts / game average	2.14	1.75	0.00	1.63	11.98
Player Salary					
Salary	7,786,954.00	8,720,888.00	5,767.00	4,160,000.00	52,938,707.00
Games missed from injury (within same season)	5.70	6.77	0.00	4.00	79.00

4. Method

I modeled injury risk using a Random Forest classifier (RF) coupled with 5-fold cross-validation. The RF method was preferred for three reasons. First, it can predict injuries with a non-linear interaction that is highly dependent on multiple complex factors, such as workloads, player playstyle tendencies from previous season averages, and player attributes. Second, because each tree only considers a random subset of variables in each split, the model helps lessen the impact of highly correlated variables and reduces over-fitting. Third, the model allows for easy post-hoc interpretability. More specifically, the algorithm enables the computation of Gini-based importance scores allowing us to identify pertinent metrics.

Model assessment and parameter tuning were done using 5-fold cross-validation. The data was stratified into five subsets, with four out of the five subsets being used as training data, and one out of five subsets used as testing data. I repeated this five times for five different subset combinations and checked confusion matrix results to ensure that my results are robust and generalize to different test samples.

The Random Forest Model is a machine learning model that makes predictions by combining many small decision trees. Each tree looks at a random portion of the data and different player statistics, adopting its own pattern of when injuries occur. The model then averages all the tree's predictions to make one overall injury probability. This approach is useful as it helps capture complex patterns while avoiding overfitting to any single part of the data.

This study implements the RF model using the “randomForest” package in R. Hyperparameters within the function include the following: ntree, mtry, nodesize, maxnodes, replace, sampsize, and classwt among others.

The number of trees (ntree) was fixed at 500, consistent with the default in R’s randomForest package. As noted by Breiman (2001), the generalization error of a random forest converges as the number of trees increases, and Liaw & Wiener (2002) observe that the out-of-bag error stabilises once ‘enough trees’ are grown. Thus, 500 trees was chosen because it provides model stability without excessive computational cost.

The mtry parameter controls the amount of variables that are considered at each split. A smaller value increases the diversity among the trees but weakens the individual trees, while a larger value reduces bias but risks high correlation. In section 5.1, mtry is fitted by maximizing the area under the ROC curve.

I chose the default values (nodesize= 1; maxnodes= NULL) for the trees, allowing them to be grown to full depth. This setting minimizes bias and allows trees within the RF model to capture complex interactions. Higher nodesize or lower maxnodes values would have restricted tree depth, leading to higher bias but lower variance among the trees.

Bootstrap sampling parameters were also set to their default values (replace = TRUE; samp size = n). This allows for each tree in the RF to be trained on a more diverse dataset created by random sampling with replacement. The result lowers variance and reduces overfitting once the trees are averaged.

Class weights in the Random Forest model were set to the default value (classwt = NULL), which weighed both injuries and non-injuries as equal. While class weighting is useful in addressing imbalanced outcomes by penalizing misclassification of injuries more heavily, I chose to handle imbalance through threshold tuning. Doing this allowed me to directly control the trade-off between false-positives and false-negatives to reflect the practical costs of missed injuries vs false alarms.

Threshold is the final classification layer of the RF model. Prior to this layer, my RF model generates a logit (or "probability") of getting injured in the next game. The threshold converts this logit into a binary classification ("yes/no"). Lower thresholds (close to 0) mean that most logits will be classified as "yes", while higher thresholds (close to 1) classify most logits as "no". Section 5.2 details the process for which the threshold is fitted.

5. Results

The goal of this study is to develop a machine learning framework that predicts next-game injury risk using publicly available data, then translate these probabilities into expected financial costs. To accomplish this, the result section follows four steps: (1) tune and validate the model to make sure it works beyond chance, (2) pick a decision threshold that balances the false positives and false negatives based on cost, (3) show the out-of-sample performance of the model at that optimal threshold, (4) identify which features matter most for predicting injuries, and (5) use injury probabilities to generate an understanding of financial risk.

5.1 Parameter Tuning

To tune the random forest's mtry hyperparameter, I fixed mtry to values between 1 and 16 (the total number of variables that are used for prediction) and computed ROC points across a grid of decision thresholds for each fold of a 5-fold cross-validation. The thresholds were 0.9, 0.5, 0.15, 0.1, 0.09, 0.08, 0.07, 0.06, 0.05, 0.04, 0.03, 0.02, 0.01, 0.007, 0.005, 0.003, and 0.001. As the threshold decreases toward 0, both the true-positive rate (TPR) and false-positive rate (FPR) increase, ultimately tracing out the ROC relationship (see Figure 2A).

For each fold, I recorded FPR and TPR at every threshold, then averaged these across the five-folds to obtain an average ROC curve (Figure 2A, black line). I included a 45° reference line to represent random chance. Because the average ROC curve sits well above this chance line, the model performs better than random classification of injuries.

To identify the optimal mtry value for the random forest model, I approximated the area under the average ROC curve (AUC) for every value of mtry and selected the value that produced the highest value (Figure 2B, red line). I estimated the AUC by summing the true positive rates (TPR) across all thresholds, as the AUC represents the model's overall ability to distinguish between injured and non-injured players. A higher AUC indicates stronger class separation (injuries from non-injuries). Among all possible configurations, the model with mtry = 16 achieved the highest approximate AUC of 8.087, slightly outperforming other values of mtry (2nd best mtry = 14, AUC = 8.078; 3rd best mtry = 13, AUC = 8.077).

5.2 Threshold Testing

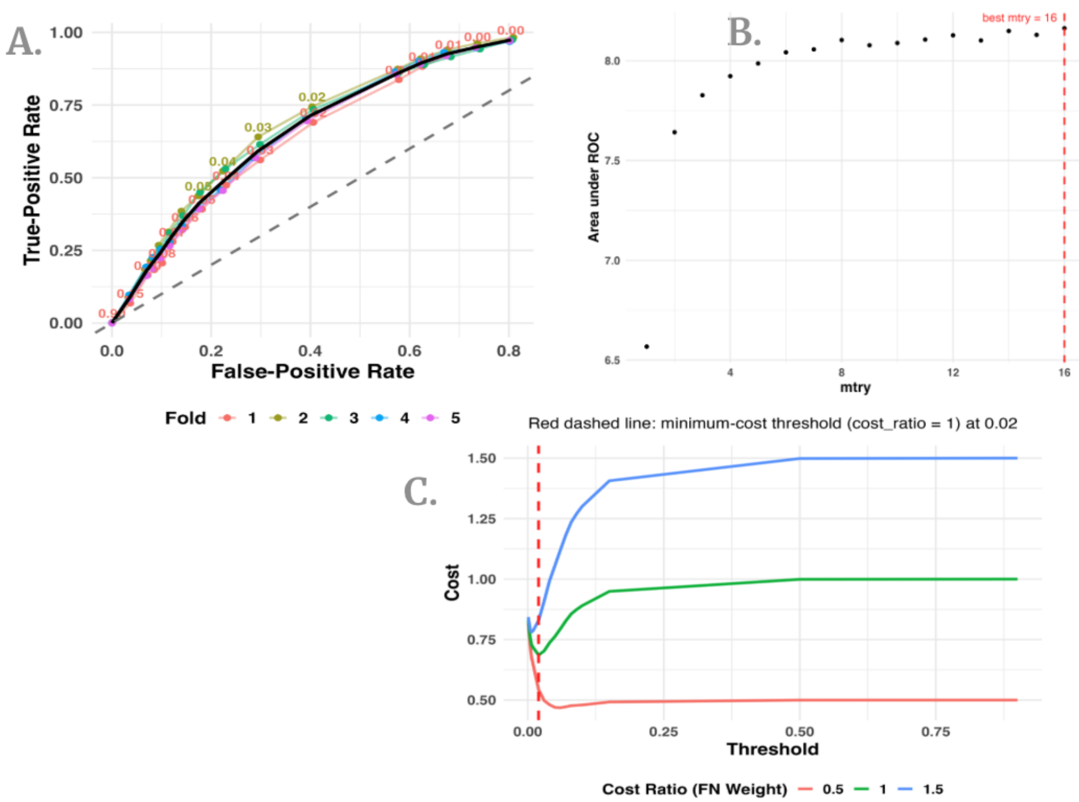
To determine which threshold minimizes cost, I first define a cost ratio between false negatives and false positives. Then, I plot an estimated cost score based on three cost ratios (0.5, 1, 1.5) in three different colors against different thresholds from 0 to 0.9 (see Figure 2C). Cost ratio, c, is defined as

$$c = \textit{cost of a FN (false negative) / cost of a FP (false positive).}$$

When c < 1, false negatives are being weighed as less than false positives. At a c = 1, false negatives and false positives are viewed equally, while c > 1 implies that a false negative is viewed as more costly than a false positive. The true cost ratio will vary by team, player, and contract. Therefore, I report results at c = 1, as a neutral expected-value baseline that does not include unverified cost-related assumptions, but my methodology is robust to any cost ratio. For a cost ratio of 1, the threshold that minimizes cost is 0.02 (minimum of green line in Figure 2C).

Figure 2: Model performance evaluation for the Random Forest classifier

(A) ROC curves averaged across folds for mtry = 16, (B) area under the ROC (AROC) across mtry values, and (C) cost–threshold curves illustrating false-negative/false-positive trade-offs.



5.3 Optimal Model and Feature Importance

In Table 2, I present the confusion matrix for my model. At a conservative 2% threshold, I predicted 69% of true, out-of-sample, injuries while correctly ruling out 63% of healthy games. In practice, this means I can generate early warnings for roughly two thirds of forthcoming injuries, giving teams a tool to minimize injury odds.

Table 2. Confusion matrix

Means and standard errors (in parentheses) across 5 folds. Accuracy and proportion correct in grey.

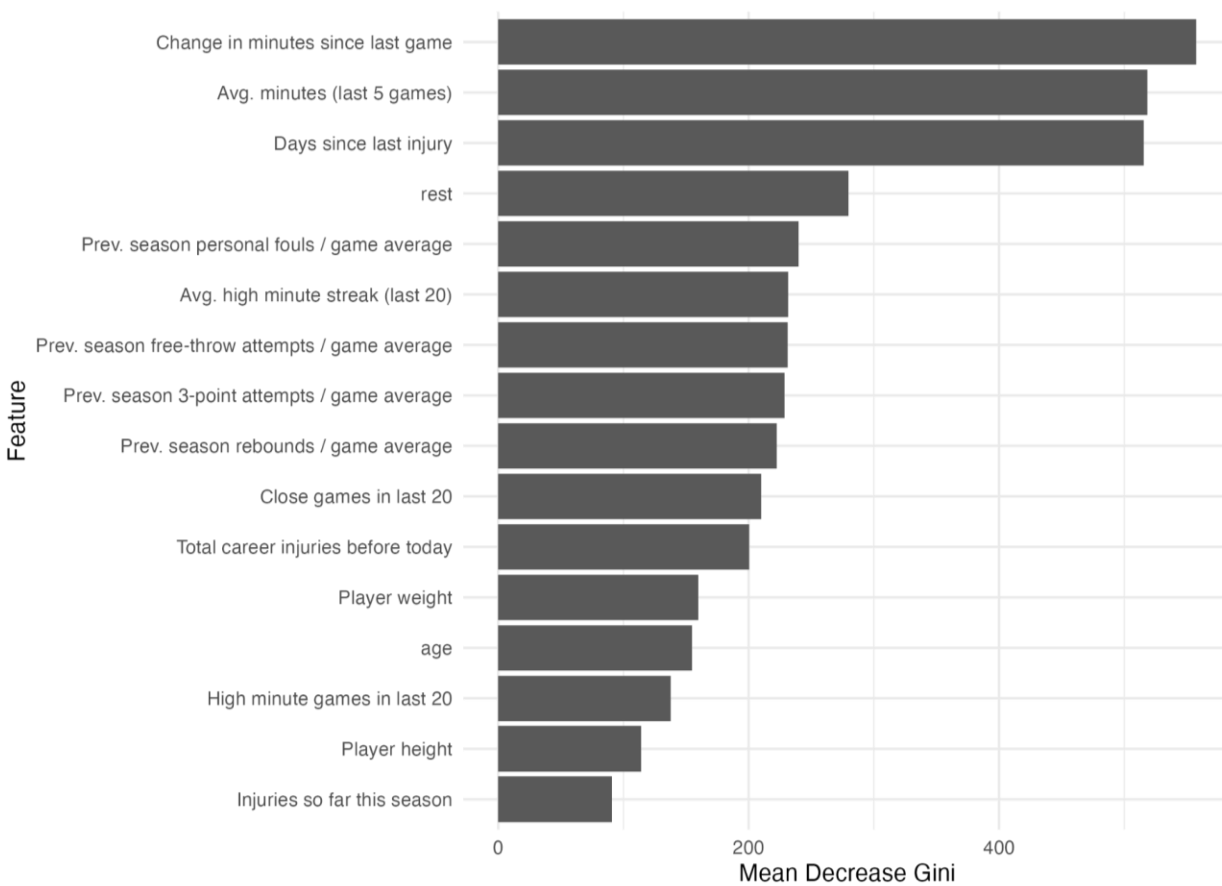
		Actual		
		0	1	
Predicted	0	13577.20 (41.32)	182.60 (8.89)	0.99
	1	8160.00 (47.13)	400.20 (7.55)	0.05
		0.62	0.69	Accuracy 0.63

After training the Random Forest model, I examined which variables most strongly influenced injury prediction through their feature importance score. A feature importance score measures how much a variable contributes to reducing impurity or how well a variable helps the model separate injured from non-injured players. High scores mean the feature was more useful for making cleaner splits between injuries vs non-injuries. Feature importance scores (Gini gain) ranked the following predictors as the most important predictors, in the following order: *change in minutes since last game*, *average minutes in the last 5 games*, *days since last injury*, and *rest*. The following variables were the least predictive of injury: *age*, *number of high minutes played (games above 30 minutes played) in the last 20 games*, *player height*, and *number of injuries previously in the season*.

The results are intuitive: players who have a sudden change in minutes compared to their last game, have suffered an injury recently, have a recency in the last 5 games of playing a significant amount of time, and aren't well rested have a higher injury risk. Specifically, players with more rest have significantly lower injury odds compared to those on 0-5 days of rest. Most injuries tend to happen on short amounts of rest, while long rests minimize injury chances, as expected. Contrary to my expectations, physical attributes, like *age* and a player's height, were not particularly useful to the model. This is inconsistent with previous findings such as Lu et al. (2022), where *age* was a driving factor in the predictions.

For days since the player's last injury, I find that players who re-injure tend to have had less time since their previous injury. I find that for a player's change in minutes since their previous game, a small increase in minutes is a mild risk amplifier, while extreme shifts either more or less are red flags. For the variable encoding the average number of minutes played in the last five games for a player, I found that the majority of injuries happened above the 20 minute zone, and injury risk slowly increased until it peaked at around 30 minutes. Like the rest of the top 4 predictors, injuries also happen frequently under 20 minutes, which suggests that *Avg. minutes (last 5 games)* is most powerful in combination with other predictors, and not a standalone predictor.

Figure 3: Feature importance score for variables within the model



5.4 Expected Cost and Salary Analysis

While previous studies have looked at forecasting injury probabilities using box-score related data, they have all stopped at predicting who is likely to get injured, without examining the financial consequences of those injuries. Utilizing salaries, and the predicted probabilities of injuries generated in the Random Forest Model, I construct a method for calculating expected cost:

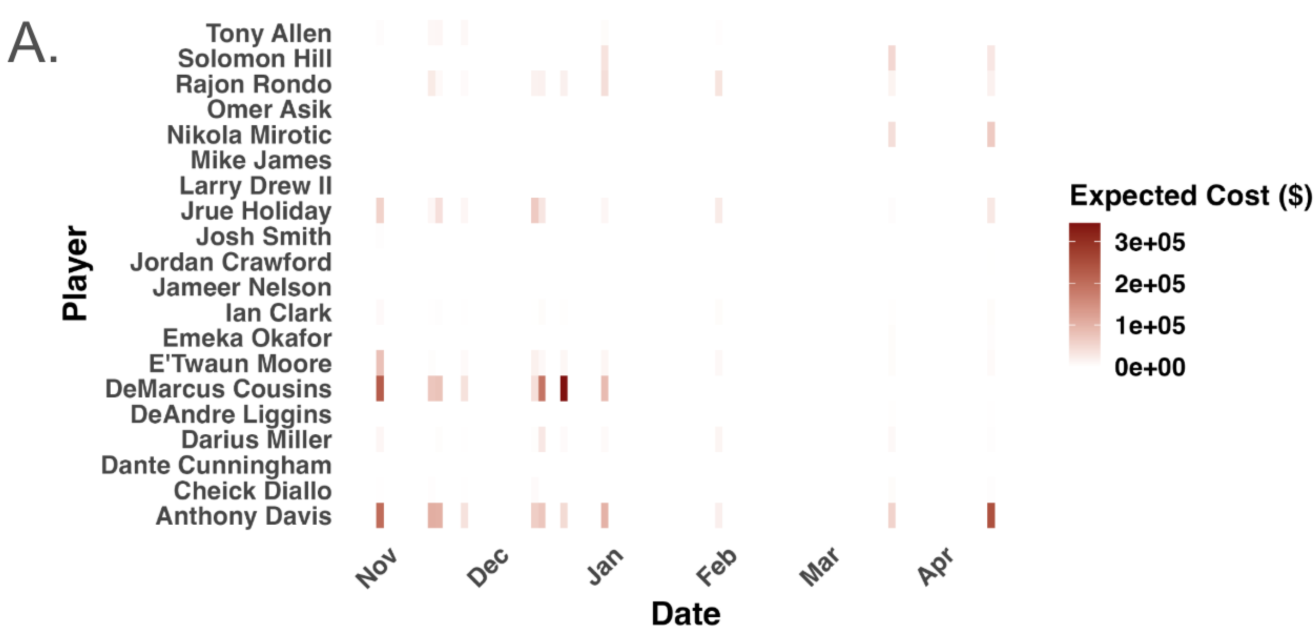
$$E[Cost] = P(injury) \times (salary \div 82) \times (average\ duration\ of\ injury)$$

Here *P(injury)* is the model's predicted probability of an injury, and salary/82 represents the player's per game salary, assuming an 82 game regular season. The average duration of injury is calculated as the mean number of games typically missed per injury. This calculation ultimately allows for a per-game estimation of a player's financial risk on the team. Figure 4A shows an example team (New Orleans, 2018), where each player's expected injury cost fluctuates throughout the season based on model predictions.

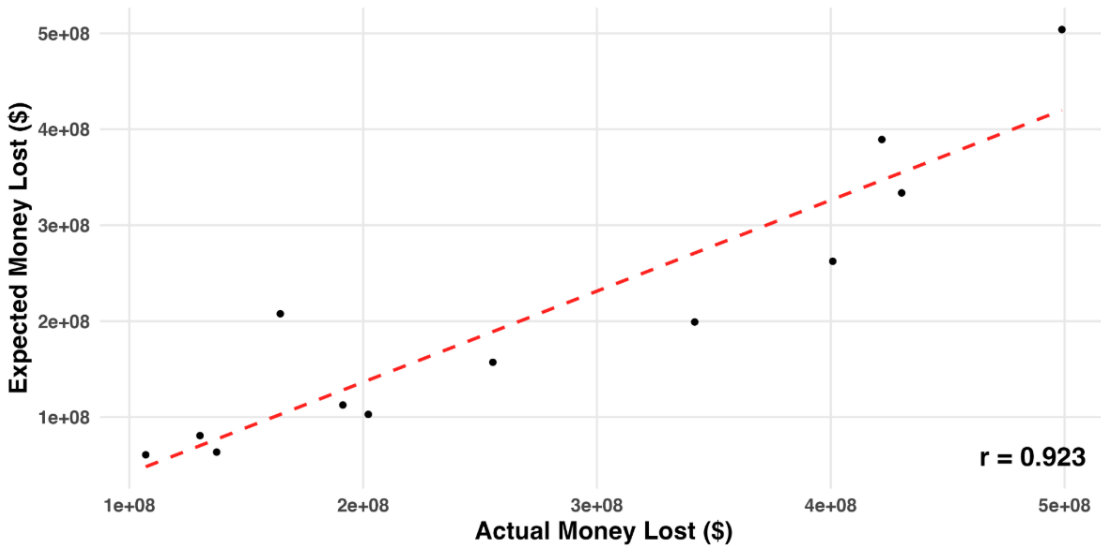
To complement this estimate, I calculate the actual financial cost of injury by multiplying the number of games missed after each injury by the player's per-game salary. This allows for a direct comparison between the expected and realized financial losses. Expected cost values were derived from the model's predicted probability of injury for each player, multiplied by their per-game salary and the average duration of injury. Figure 4B plots expected versus actual financial costs, showing a high degree of correlation between the model's expected cost and real financial outcomes ($r = 0.955$). Figure 4C shows the aggregated total expected costs by team and season. Expected cost has gone up throughout the years as a result of inflationary changes of salary. All together, these analyses demonstrate how injury prediction models can be used to estimate financial risk to a team.

Figure 4: Expected and realized injury-related financial costs

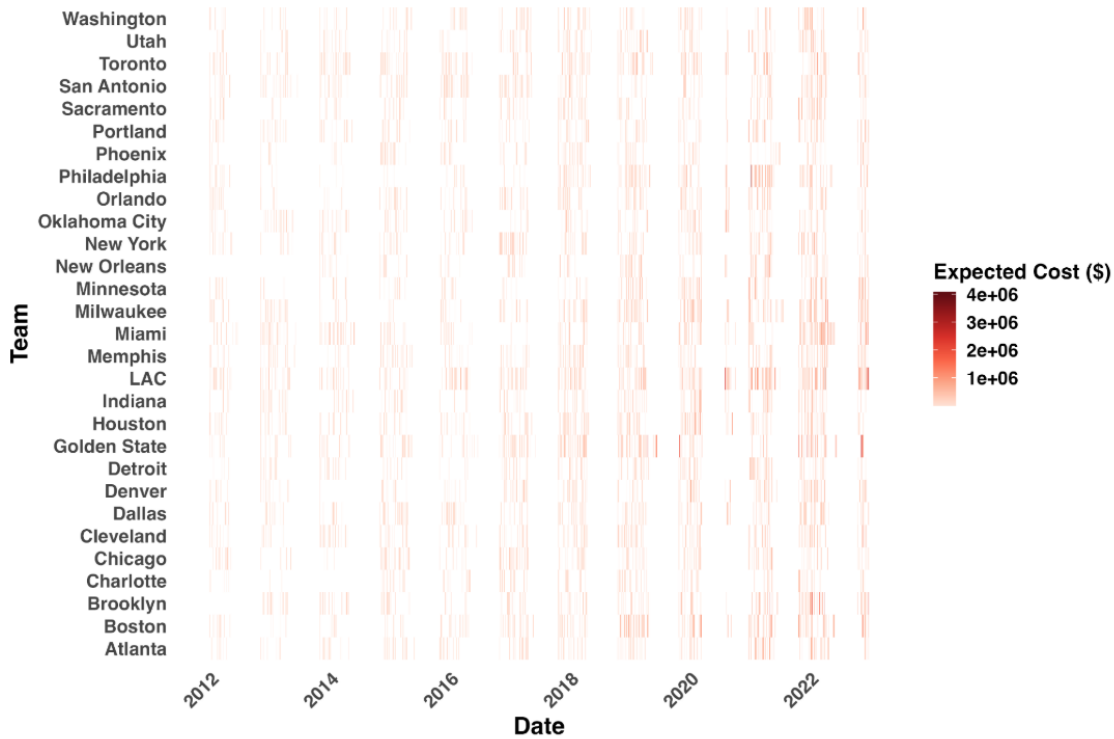
(A) Player-level expected cost heatmap for New Orleans (2018), (B) correlation between expected and actual team-level financial losses, and (C) league-wide expected injury costs over time (2012–2022).



B.


5e+08
4e+08
3e+08
2e+08
1e+08
Expected Money Lost ($)
r = 0.923
1e+08
2e+08
3e+08
4e+08
5e+08
Actual Money Lost ($)

C.


Washington
Utah
Toronto
San Antonio
Sacramento
Portland
Phoenix
Philadelphia
Orlando
Oklahoma City
New York
New Orleans
Minnesota
Milwaukee
Miami
Memphis
LAC
Indiana
Houston
Golden State
Detroit
Denver
Dallas
Cleveland
Chicago
Charlotte
Brooklyn
Boston
Atlanta
Team
2012
2014
2016
2018
2020
2022
Date
Expected Cost ($)
4e+06
3e+06
2e+06
1e+06

5.5 League-Wide Injury Cost Simulations

To explore the potential value of *rest*, I simulated a scenario in which high-risk players were rested before their next games. In this scenario, I selected a threshold that would be considered risky, and determined the days between the player's current game and the next team game that he could participate in (within the season). To demonstrate how to calculate league-wide costs, I selected a risky threshold at 0.10 and carried out cost-analysis. This selection is arbitrary, but it allows teams to take on a relatively high level of injury risk tolerance.

Building on this, I simulated the effect of resting players exceeding the threshold, rather than letting them play the next game. Specifically, I increased each player’s *rest* and *days since last injury* variables by the number of days between the current and next game.

I then applied the previously developed Random Forest model using these updated variables to generate new injury probabilities. Using these revised probabilities, I recalculated each player’s expected injury cost. Comparing the new expected costs to my original estimates allowed me to quantify the financial effect of resting high-risk players for one game.

Table 2 presents the financial outcomes of this simulation. Across multiple seasons, I show original estimates (*Expected Cost Before*), new estimates with simulated *rest* (*Expected Cost Updated*), and the total savings from simulated *rest* across all teams in the NBA (*League Wide Savings*). Positive *League Wide Savings* indicates cost savings from avoided injuries, while negative *League Wide Savings* would indicate losses due to unnecessary *rest*. Note that *League Wide Savings* is positive for every season between 2012 and 2023, and achieves a maximum of approximately $5.7M in 2021. This estimate is also conservative, as 9,551 observations in the dataset lack salary information, meaning true savings are likely even higher.

This framework can be extended to evaluate the effects of giving players multiple games of *rest*, allowing for estimation of optimal *rest* durations. Alternatively, it can be coupled with additional metrics related to the contribution of a player to each game, allowing teams to weigh the cost-benefit of resting a high-injury-risk player.

Table 2: Expected financial savings of simulated rest

Season	# Expected Cost Before ($)	# Expected Cost Updated ($)	# League Wide Savings ($)
2012	53,084,023.00	52,290,431.00	793,592.00
2013	58,416,697.00	57,701,799.00	714,898.00
2014	69,726,235.00	68,802,051.00	924,184.00
2015	89,525,765.00	88,680,847.00	844,918.00
2016	91,375,624.00	90,179,228.00	1,196,396.00
2017	122,890,987.00	122,032,816.00	858,171.00
2018	171,450,351.00	170,157,843.00	1,292,508.00
2019	267,101,203.00	264,294,772.00	2,806,431.00
2020	334,354,611.00	330,440,266.00	3,914,345.00
2021	373,074,240.00	367,327,041.00	5,747,199.00
2022	471,129,705.00	468,381,953.00	2,747,752.00
2023	181,735,706.00	180,382,552.00	1,353,154.00

Discussion

This study set out to answer a practical question for NBA front offices: Can a machine learning model that combines publicly available data be used to not only forecast player injuries, but also to estimate their financial impact?

To answer this, I merged five public datasets (injury logs 2010-2022, game-level box scores with minutes played, player salary details by season, season-level box score and player attribute descriptions, and team-level box scores per game) into a single game, individual player based dataset. I engineered and derived 16 workload and recency metrics, treated missing values, and trained a Random Forest classifier for predicting injuries. To optimize the model, I utilized 5-fold cross-validation to guide hyper-parameter tuning (mtry testing 1-16, threshold testing) and tested the model out-of-sample for performance estimation.

My results show that simple box score data can predict injuries with above-chance accuracy. At a 2% threshold, the model predicted around 69% of true injuries out-of-sample, while ruling out 62% of healthy games, suggesting that my model can offer teams early warnings for most upcoming injuries well above chance. Beyond predictive accuracy, the integration of financial risk modeling introduces a novel extension of injury forecasting. Across seasons, expected injury-related costs showed steady inflation from \$53.1M in 2012 to \$471.1M in 2022, closely aligning with the model's updated estimates after rest simulation. On average, resting players above the injury-risk threshold produced league-wide savings each year, peaking at \$5.75M in 2021 (Table 2). By translating injury probabilities into salary-adjusted costs, my analysis offers teams a quantitative framework for managing both player health and financial cost.

Feature Interpretation

Because the dataset only had a small proportion of injuries, the accuracy of the model exceeded my expectations. Since I decided on a lower injury threshold (0.02), I accepted a high number of false positives in exchange for identifying more true injuries. The average across five-folds yielded a 63% specificity (TN/(TN+FP)) and 69% sensitivity (TP/(TP+FN)). The top features in my model illustrate that changes in player workload and recovery dynamics are key drivers of injury likelihood. Any spikes in last game minute change are red flags for injury likelihood. The amount of days since the players' last injury is also a valuable metric, as if the time is shorter, there may be a chance that the player didn't fully recover from his previous injury. Average minutes of a players' last 5 games gives a representation of how much time a player has been playing recently (around the past two weeks). *Rest* is a powerful metric as it totals the number of days a player has in between games to potentially recover their body.

Practical Implications

My research is most impactful for NBA organizations. For load-management staff the model provides a flag for early-warnings toward injuries. Rather than simply proving the fact that longer *rest* lowers injury risk, the model quantifies when and for whom *rest* produces the greatest economic return. In particular, rest emerged as one of the strongest predictors in the feature-importance analysis, and the financial simulation demonstrated that players flagged as high-risk who were strategically rested generated expected savings for teams. For example, given the sheer amount of money allocated to star players, the expected savings from model-guided *rest* decisions remained positive throughout every season.

Future research can extend this framework by refining the optimal amount of days to look at alternative scenarios for player *rest*, and determining a truly optimal cost ratio between false positives and false negatives. By integrating salary and injury prediction, my research moves beyond just injury prediction, and gives outlets into what can be done with injury probabilities.

Limitations

There are three main limitations to my findings. First, when I decided to use previous season statistics as features in my model, I sacrificed 16% (22631/140879) of total rows of the final data. However, I decided that it was a reasonable compromise because previous season statistics such as rebounds and free-throw attempts give insight into certain players' physical playstyle. Especially with free-throw attempts, a player getting fouled is something that could greatly increase injury odds. Second, one of my engineered predictors (*number of close games in the last 20 games*) rely on end of game point differentials which is a simplification from the NBA's definition of close games. Additionally, throughout the injury dataset, the third most frequent type of injury is "Placed on IL" with no other description. With this description, I am unable to determine if "Placed on IL" is something that is an injury that could be predicted or something else like being placed on the IL for personal reasons or NBA health and safety protocols. Throughout my study, I will be treating "Placed on IL" as a predictable injury, which may increase the amount of injuries in the dataset. Furthermore, I do not consider illness and infections as injuries. This means the model treats many instances where players log big minutes, get sick, and miss the next game as non-injuries (0). The model may learn that heavy minutes are less risky than they really are if there are significant amounts of high minute trends that lead to sickness. Finally I acknowledge that the salary analysis process may be an oversimplification of real-world scenarios. For example, I don't take into account the money lost from *rest*, and I assume that salary is evenly distributed across an 82-game season, which may be inaccurate considering NBA playoffs. Additionally, variables were not reset at the end of each season and were continuously counted across the end and start of seasons.

Conclusion

This research demonstrates that publicly available data and machine-learning methods can meaningfully forecast NBA injury risk and, at the same time, quantify its financial consequences. By combining injury-probability generated from a machine learning model with salary-based cost estimates, the study introduces a new framework for NBA teams that connect topics in sports medicine and health analytics to economic decision making in basketball. While this study doesn't dive into an optimization system, the approach demonstrates how predictive models can inform load-management strategy while reducing expected salary loss. Future work may work on expanding this framework by optimizing rest-time, deciding on a better "risk-threshold", or taking into account the financial cost of resting players. Ultimately, injury forecasting offers front offices a simple, yet effective tool to preserve both player health and economic success.

Acknowledgements

I would like to express my gratitude for Professor Peter Kempthorne for his guidance in helping me develop the methodological foundation for this study and his continued support throughout the research process. I also cannot thank Brenden Eum enough for his mentorship throughout both the methodological and paper-writing process. His guidance was invaluable in the implementation of the nuanced parameter tuning for my model and all the other facets of this project. Finally, I would like to thank my favorite NBA team, the Philadelphia 76ers, for inspiring this project. Watching Joel Embiid and other crucial players battle through injuries, while my hopes for a championship slowly vanished, motivated me to explore injury forecasting.

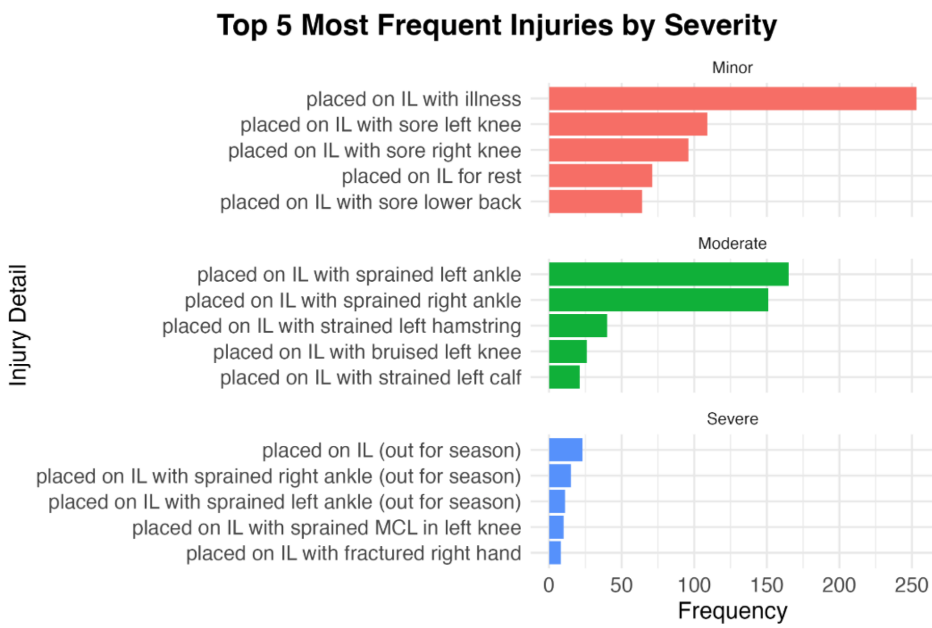
References

1. Smith, S. (2016, May 16). *What is the real cost of injuries in professional sport?* Medium. https://medium.com/@stephensmith_ie/what-is-the-real-cost-of-injuries-in-professional-sport-fee1d66a7502
2. Dowsett, B. (2022, August 8). *The NBA is turning to wearable sensors to prevent player injuries*. FiveThirtyEight. https://fivethirtyeight.com/features/the-nba-is-turning-to-wearable-sensors-to-prevent-player-injuries ABC News
3. Cohan, A., Schuster, J., & Fernandez, J. (2021). *A deep learning approach to injury forecasting in NBA basketball*. *Journal of Sports Analytics, 7*(1), 1–12. https://doi.org/10.3233/JSA-200529
4. Charest, J., & Samuels, C. H. (2021). Impacts of travel distance and travel direction on back-to-back games in the National Basketball Association. *Journal of Clinical Sleep Medicine, 17*(11), 2269-2274. https://doi.org/10.5664/jcsm.9446
5. Lu, Y., et al. (2022). Machine learning for predicting lower extremity muscle strains in the NBA. *Orthopaedic Journal of Sports Medicine*. https://pubmed.ncbi.nlm.nih.gov/35923866
6. Chan, C. C., Yung, P. S. H., & Mok, K.-M. (2024). The relationship between training load and injury risk in basketball: A systematic review. *Healthcare*, 12(18), 1829. https://doi.org/10.3390/healthcare12181829
7. GHopkins. (2018). *NBA injuries (2010–2018)* [Dataset]. Kaggle. https://www.kaggle.com/datasets/ghopkins/nba-injuries-2010-2018
8. SzymonJWiak. (n.d.). *NBA traditional stats* [Dataset]. Kaggle. https://www.kaggle.com/datasets/szymonjwiak/nba-traditional

9. Justinas. (n.d.). *NBA players data* [Dataset]. Kaggle. https://www.kaggle.com/datasets/justinas/nba-players-data
10. NathanLauga. (n.d.). *NBA games details: EDA* [Code/dataset]. Kaggle. https://www.kaggle.com/code/nathanlauga/nba-games-eda-let-s-dive-into-the-data/input?select=games_details.csv
11. Ratin21. (n.d.). *NBA player stats and salaries (2010–2025)* [Dataset]. Kaggle. https://www.kaggle.com/datasets/ratin21/nba-player-stats-and-salaries-2010-2025
12. Jwiak, S. (2024). *NBA traditional boxscores 1997–2024* [Dataset]. Kaggle. https://www.kaggle.com/datasets/szymonjwiak/nba-traditional
13. Benson, L. C., Clermont, C. A., Bošnjak, E., Ferber, R., & Osis, S. T. (2021). *Evaluating methods for imputing missing data from athlete workload datasets. Sensors*, 21(7), 2371. https://doi.org/10.3390/s21072371
14. Mohammed, M. B., Zulkafli, H. S., Adam, M. B., Ali, N., & Baba, I. (2021). *Comparison of five imputation methods in handling missing data in a continuous frequency table. AIP Conference Proceedings, 2355*(1), 040006. https://doi.org/10.1063/5.0053286
15. Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324
16. Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R News, 2(3), 18–22. https://journal.r-project.org/articles/RN-2002-022/RN-2002-022.pdf

Supplementary

S1.



To get an understanding of the frequency and different types of injury that occur in my dataset, I construct three severity tiers to categorize injuries for better understanding: severe, moderate, and minor. Frequency of injuries in my dataset can be seen in Graph A. In graph B, the top five most frequent injuries are displayed by severity. In my dataset, the highest frequency in the severe injury category were injuries that sidelined players for the season but with no further injury detail. The second most common was a sprained right ankle that ruled players out for the season, followed by a sprained left ankle of the respective nature. For the moderate tier, regular sprained left and right ankles were of highest frequency. For minor injuries, sore left and right knees were the most frequent. It is also noticeable that there are much more minor and moderate injuries compared to severe ones.