Which Match Statistics Most Significantly Affect Tennis Match Results on the ATP Tour?

Teddy Kelly

Table of contents

Abstract

The Association of Tennis Professionals (ATP Tour) consists of the highest quality male tennis players in the world. With over a thousand professionally ranked players on the tour and approximately 65 tournaments played every season, there are a countless number of matches played each year, with each one containing large amounts of data of how each involved player performed in the specific matches. Such data includes valuable information on players' serving and returning numbers. This paper analyzes these match statistics using a tennis dataset containing every single match played on the ATP Tour between the years 1991 and 2022, with data on various in-match statistics from each one played. Using three logistic regression models, one for each surface (clay, grass, and hard courts), I have determined that gaining a significant advantage over the opponent in generating break points and break point conversion percentage is associated with increasing the odds of winning a match the most. Additionally, second serve win percentage also has a significant effect on the likelihood of winning, and in some cases, the surface does influence how certain statistics affect match outcomes.

Introduction

There are a few details that I should address before diving into the methodology used in this paper. Firstly, the motivation of this paper: why should anyone bother to care about analyzing tennis statistics from matches that have already been played? This is a fair question, however, the main goal of this paper is not to make predictions on who wins, but to instead determine the match statistics that are most significantly associated with the outcome of the match. Specifically, this paper seeks to find the marginal effect of each match statistic on the likelihood of winning using logistic regression. This is extremely valuable information for players and coaches because it allows for comparison of the magnitudes of the statistics that may contribute to higher chances of winning. For example, if the data suggests that increasing a particular statistical category by one unit in matches leads to a greater likelihood of winning than other statistical categories, then players will seek to gain an edge in that specific category more than the other statistics. In the results section, we will see which statistics most affect tennis match outcomes.

Secondly, it's important to understand the meaning of the match statistics discussed throughout this paper. Below is a summary of general tennis terminology that is needed to read this paper, along with their definitions and theoretical effect on tennis matches.

Serve

- The first shot played on a given point.
- The server, the player who hits the first shot of the point, has full control of where they choose to hit their serve. After the serve, all other shots are dictated by where the opponent chooses to hit.
- The player serving has a major advantage of winning the point since they get to play the first shot.
- A player gets two serves:
  - **First Serve**: Servers typically hit their first serves with more risk (faster and closer to the service lines). Therefore, landing a higher percentage of first serves is important for a server's success.
  - **Second Serve**: Second serves are usually played with more safety to reduce the risk of double faulting (less speed and more spin). Second serves are usually more vulnerable and are more easily attacked by the opponent. Therefore, winning a higher percentage of second serve points is also vital for a server's success since this will give them an edge over their opponent.

Aces

- When the server wins the point on their first shot (the serve) without the returner touching the ball with their racket.
- Players who hit more aces win their service points quicker and are more likely to win matches.

Double Faults

- When the server misses both their first and second serve. This results in the server automatically losing that point.
- Players who concede more double faults give their opponents more free points and are less likely to win.

Service Games and Return Games

- For a specific game that is played, it is a service game for one player and a return game for the other.
- For example, if player i is serving and player j is returning, then that specific game is a service game for player i and a return game for player j.
- The players alternate who serves and who returns after each game.
- Rules on how a player wins:
  - a player must **win 4 points** by a margin of two to **win a specific game**,
  - a player must **win 6 games** by a margin of 2 games to **win a set**.
  - Then, a player must **win 2 sets** to win the entire **match if it's best of 3 sets** or they must **win 3 sets** if it's a **best of 5 match**.

Break Point

- A break point is a point in which if the server loses that point, then they will lose that service game. If a server loses their own service game, then we say that the returner **broke serve**.
- Players who face more break points are likely to be broken more frequently and thus lose more matches.
- On the other hand, players who generate more break point opportunities on their return games are more likely to break and thus have a higher chance of winning.

Surface Types

There are three main surfaces on the ATP Tour that tennis matches are played on: clay courts, grass courts, and hard courts.

- **Clay Courts**
  - Slower surface: The clay literally slows the down the speed of the ball when it makes contact with the ground and causes the ball to have a higher bounce.
  - This allows players to have more time to react to their opponent's shots and leads to longer and more grueling rallies.
  - Clay favors more defensive players and hinders those who rely on their serve for success.
  - In the dataset that we will analyze, we will expect to see **lower serving stats** and **higher returning stats** on clay compared to the other surfaces.
- **Grass Courts**
  - Quicker surface: The ball skids through the grass as it makes contact with the ground and has a much lower bounce.
  - This reduces the amount of time that players have in reacting to their opponent's shot and leads to quicker rallies
  - Grass favors more offensive players and those who have bigger serves.
  - We will expect to see **higher serving statistics** and **lower returning statistics** on grass compared to the other surfaces.
- **Hard Courts**
  - In terms of court speed, hard courts are in between clay and grass, but are typically on the faster side (although the court speeds of certain hard courts can vary significantly). The ball bounces much more normally than clay or grass, and it is much easier to move on hard courts
  - Most of the matches on the ATP Tour are played on hard courts, so both defensive minded and offensive oriented players are forced to adapt their game to hard courts.
  - We will expect that both serving and returning statistics will fall in between the values of those for grass and hard courts.

Previous Research

Before diving into the methodology, it's important to acknowledge the previous research done by others in this subject area. Specifically, Michal Kokta published a blog post titled "Predicting ATP Tennis Match Outcomes Using Serving Statistics" on Medium.com. As the title suggests, he used only serving statistics to make these predictions, as well as only using matches played during the 2019 ATP tennis season. He specifically analyzed the following serving statistics:

- Aces
- Double Faults
- First Serve Percentage
- First Serve Winning Percentage
- Second Serve Winning Percentage

My paper also investigates these serving statistics, in addition to several returning statistics like break points generated and break point conversion rate. Kokta's analysis of these serving statistics primarily consisted of exploring the percentage of matches in which the winner had a higher level of the specified statistics. For example, his research developed the following conclusions:

- "The winner of a match hit more aces than the losing player only roughly 55% of the time" and match losers only hit more double faults than the winning player 49% of the time (Kokta, 2020).
- "The winner of the match had a higher 1st serve winning percentage roughly 80% of the time" and a "higher 2nd serve winning percentage roughly 74% of the time" (Kokta, 2020).

Therefore, his research suggests that 1st serve winning percentage and 2nd serve winning percentage are much better indicators of tennis match outcomes than the number of aces and double faults the player hit. This makes sense because double faults typically don't happen often enough to significantly determine the outcome of a match. Also for aces, just because a player is not hitting many aces does not mean that they are necessarily serving poorly. They could instead be hitting a lot of service winners (serves in which the opponent gets their racket on the ball but does not land the ball back in play) which do not count as aces.

Kokta also compared the median values of first serve percentage between the winners and losers of the matches. He found that the "median first serve percentage for the winner of the match was roughly 63%" and 61% for the losers (Kokta, 2020). These median values are perhaps closer than anticipated, however the match winners do have a higher median first serve percentage which is expected.

Although these findings from Kokta are insightful, there is still more room to be explored in this subject matter. For example, there are many other tennis statistics not explored by Kokta that could potentially be key indicators in how tennis matches unfold. This paper attempts to look at two more returning variables (break points generated and break point conversion rate) and evaluate their effect on match outcomes. Also, Kokta did not investigate the marginal effects of these serving statistics and how they compare to each other. First serve percentage, first serve winning percentage, and second serve winning percentage certainly all help players win more matches, but which statistic is **most linked** with winning and by **how much**? Utilizing regression, we can easily answer these questions by observing how a one unit change in any one of these statistics changes the likelihood of winning.

Methodology

As noted in the previous section, the primary goal of this paper is to determine the tennis match statistics that are most significantly associated with match outcomes using regression analysis. The reason for using regression is that it allows us to find the marginal effects of the specified match statistics on match outcomes, enabling us to determine the change in the likelihood of winning from changing one of the match statistics by one unit. Specifically, we will use **logistic regression** since the dependent variable, *match result*, is binary: A player can either win the match or lose the match. Before going in greater detail about the setup of the logistic regression, I will first introduce the dataset that was used to perform the regression analysis.

Tennis Dataset

The tennis data used for this paper comes from the "ATP matches" dataset created by Sijo Manikandan on Kaggle. It contains over 92,000 tennis matches played from 1991-2022 with data on the following match statistics for both the winning and losing player for each match:

- Aces, double faults, total service points, total service games
- Total 1st serves made, total 1st serve points won, total 2nd serve points won
- Total break points faced and total break points saved.

Data Preparation

I performed the following data cleaning procedures to prepare the data for regression analysis:

1. Removed any matches that had missing data for any of the match statistics, any matches with impossible match statistic values, and any matches that ended in a retirement.

2. Removed any matches that were played on "carpet courts" since the ATP Tour no longer holds matches on carpet.

3. Addressed match statistic outliers

    - I experimented with winsorizing the top and bottom 1% of data in each statistical category to reduce the effect of extreme values. However, the logistic regression output for the winsorized data was almost identical to the logistic regression output for the unwinsorized data, suggesting that the regression results are not driven by extreme outliers.
    - I also did not completely remove any observations with extremely high match statistics because match statistic values increase as the length of a match increases. So, deleting these observations would likely remove very tightly contested matches which are important to study.

4. Using these provide match statistics, I created the following serving and returning statistics for both the winners and losers of the matches that we will explore with our logistic regressions:

    - Serving Stats
        - 1st serve % (1st serves made / total service points)
        - 1st serve points won % (1st serve points won / 1st serves made)
        - 2nd serve points won %. (2nd serve points won / (total service points - 1st serves made))
        - Break points saved % (break points saved / break points faced)
    - Returning Stats
        - Break points created (**Opponent's** break points faced)
        - Break point conversion rate (1 - **opponent's** break points saved %)
        - Return points won (**opponent's** total service points - (**opponent's** 1st serve points won + **opponent's** 2nd serve points won))
        - Return points won % (Return points won / **opponent's** total service points)
    - In the results section, we will perform some explanatory data analysis by looking at the summary statistics of the newly created match statistics and compare the mean values of them for the winners and losers of matches using data visualization. This will give us an idea of what to expect for the coefficient estimates as we run the logistic regressions. But first, I will explain the methodology used to be able to run a logistic regression on this dataset.

5. Created Difference Variables for Logistic Regressions

    - With how the data currently exists within the dataset, we are not able to run any meaningful regressions because there are two columns corresponding to each match statistic: winners column and losers column.
    - The solution to this problem is to create new variables that represent the difference between two player's corresponding stats.
    - To accomplish this, I randomly assigned the players involved in each match as player A or player B so that for some matches, player A is the winner and for other matches player B is the winner.
    - This makes it so that the match statistic differences that are calculated are not always the same for every match. For example, for some matches, the difference in match statistics will be the winner's statistics minus the loser's, while for other matches, the differences will be calculated as the loser's statistics minus the winner's.
    - I created a `match_result` variable that tells us which player won the match:
        - If player A won, then `match_result = 1`
        - If player B won, then `match_result = 0`
    - This gives us a valid dependent variable that can be used for logistic regression since `match_result` is binary: 0 or 1. Since a match result of 1 corresponds to player A winning the match, I have calculated the difference statistics by subtracting player A's statistics by player B's statistics to have the difference statistics be from player A's point of view.
        - If the difference for a match statistic is positive, then player A had a higher value of the specified statistic
        - If the difference is negative, then player B had a higher value for the specified statistic
    - This will mean that for the logistic regressions, the interpretations will be from the perspective of player A. For example, a one unit change in the difference of a particular statistic will lead to a change in the log-odds of player A winning by the value of the coefficient estimate.

6. Standardize The Difference Variables

    - Since the scale on which the match statistics are measured differ from one another, it would be difficult to compare the effects that each statistic has on match outcomes.
    - For example, some of the statistics like aces, double faults, and break points created are **counts**. However, other statistics like 1st serve percentage, 2nd serve points won percentage, and break point conversion rate are **percentages**.

- Standardizing the differences of these match statistics allows for direct comparison of the effects of changing these variables on match outcomes.
- Specifically, I used R-studio's `scale()` function which automatically uses z-score standardization $z = \frac{x-\mu}{\sigma}$ to generate the standardized difference variables.
- Note that this changes the interpretations of the coefficient estimates for the regression results. (For a one standard deviation increase in the difference of match statistic x, the odds of player A winning increases by a factor of e^{β}, all else equal.)

7. Created Separate Data Frames for each Surface (Clay, Grass, Hard)
   - Along with the goal of understanding the marginal effects of tennis match statistics on match outcomes, this paper also seeks to understand how the surface of a match influences these marginal effects of the match stats.
   - I have created a separate data frame for each surface and will run separate regressions using the same variables to fully understand the influence of surface on the outcomes of tennis matches.
8. Split Each Surface Data Frame into 80% for Training and 20% for Testing
   - Although this paper does not seek to make predictions on future tennis matches, I have still left 20% of each of the clay, grass, and hard court data frames to test the validity of each of the regressions on unseen data.

Logistic Regression Estimating Equations

The following are the match statistics that I am most interested in studying for their effect on tennis match outcomes. I have also included their theoretical effects:

- Aces
  - Winners should hit more aces on average
  - Servers do not necessarily need to hit aces to be successful (Impact may not be as large as other statistics)
- 1st serve Percentage
  - Winners should have a higher first serve percentage than their opponents on average
  - The surface should also effect 1st serve percentage
    - Landing a higher percentage of your 1st serves than your opponent is likely more important on grass and hard court since those surfaces favor faster serves.

        * Clay helps the returner to neutralize faster serves, meaning that 1st serve percentage may not be as important on clay.
    - 1st serve percentage should overall have one of the largest marginal effects on winning matches.

- 2nd Serve Points Won Percentage
    - Winners should on average have higher 2nd serve winning percentages.
    - Increasing the difference between you and your opponent's 2nd serve win percentage should have a strong marginal effect because this is typically a shot that returners can attack.
    - Therefore, if a player can significantly out perform their opponent on 2nd serves, they will greatly increase their chances of winning.

- Break Points Created
    - Winners of matches should generate more break point chances than their opponents.
    - This alone however will not clearly decide the outcome of a match because generating break points means nothing unless a player converts those opportunities. Therefore, break point conversion rate is also needed.

- Break Point Conversion Rate
    - Match winners should on average have higher break point conversion rates than match losers.
    - However, having a high break point conversion rate for a match does not necessarily increase the chances of winning.
    - For example, if a player goes 1/1 on break points won, then their break point conversion rate is 100%, however their opponent could go 3/9 on break points which is 33% (worse than 100%), but the opponent likely won because they actually broke serve more times.
    - Therefore, break points created and break point conversion rate are needed together.
    - I did attempt to add break points converted, but this resulted in multicollinearity issues.

Below is the estimating equation that each of the three clay, grass, and hard court logistics regressions will use to estimate the effect that the match statistics above have on match outcomes:

$$\begin{aligned} Prob(match\_result_i = 1) = \beta_0 + \beta_1 \cdot ace\_diff_i + \beta_2 \cdot diff\_1stInPct_i + \\ \beta_3 \cdot diff\_2ndWonPct_i + \beta_4 \cdot bpCreated\_diff_i + \\ \beta_5 \cdot bpConvPct\_diff_i + u_i \end{aligned}$$

Since a match result of 1 refers to player A winning the match, the regression output will be interpreted from the perspective of player A's odds to win a match.

Results

Summary Statistics

Table 1: Winners vs Losers Summary Statistics

Statistic	Mean	St. Dev.	Min	Max
Winner Aces	6.53	5.37	0	113
Loser Aces	4.85	4.68	0	103
Winner Double Faults	2.77	2.37	0	26
Loser Double faults	3.52	2.63	0	26
Winner Break Points Saved	3.61	3.12	0	24
Loser Break Points Saved	4.90	3.30	0	28
Winner Break Points Faced	5.29	4.10	0	34
Loser Break Points Faced	8.91	4.15	0	38
Winner Break Points Converted	4.01	1.69	0	15
Loser Break Points Converted	1.68	1.59	0	13
Winner 1st Serve %	0.61	0.09	0.28	0.98
Loser 1st Serve %	0.59	0.09	0.23	0.98
Winner 1st Serve Points Won %	0.76	0.08	0.27	1.00
Loser 1st Serve Points Won %	0.66	0.10	0.06	1.00
Winner 2nd Serve Points Won %	0.56	0.10	0.00	1.00
Loser 2nd Serve Points Won %	0.45	0.10	0.00	1.00
Winner Break Points Saved %	0.68	0.28	0.00	1.00
Loser Break Points Saved %	0.51	0.20	0.00	1.00
Winner Break Points Converted %	0.49	0.20	0.00	1.00
Loser Break Points Converted %	0.32	0.28	0.00	1.00
Winner Return Points Won	34.65	11.64	5	112
Loser Return Points Won	25.99	12.62	0	117
Winner Return Points Won %	0.43	0.07	0.09	0.86
Loser Return Points Won %	0.32	0.07	0.00	0.68

N = 82,431 Matches

Summary Statistics interpretations

The summary statistics table displays many different variables, so I have only interpreted the results of the statistics that we will analyze for the regressions, along with some other interesting findings.

- Aces

  - On average, winners of tennis matches hit about **1.68 more aces** than losers do.

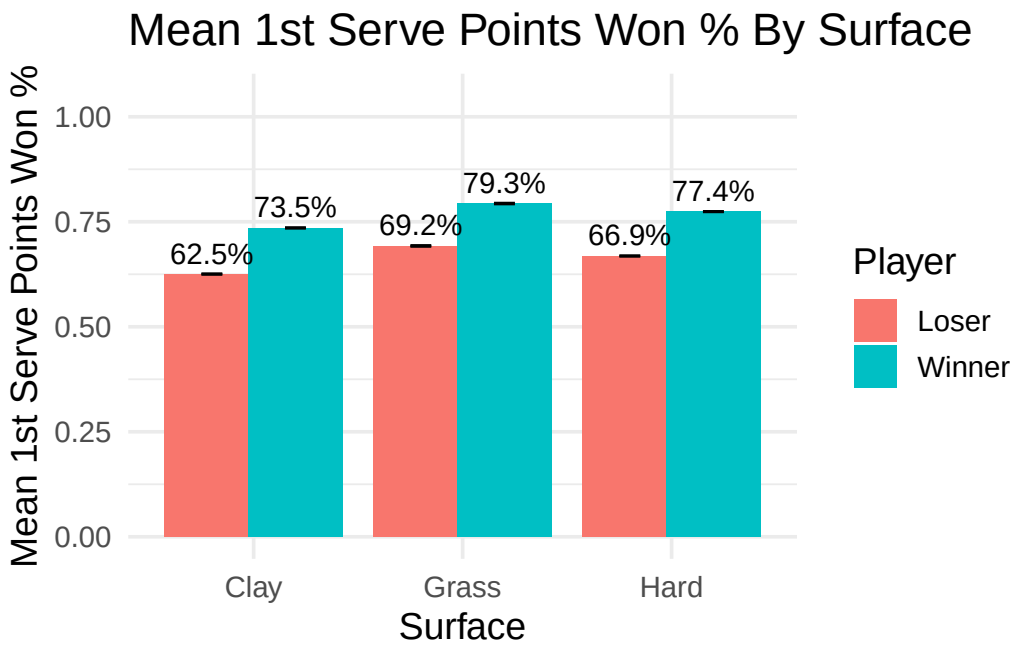
- Break Points Saved
  - This stat is interesting. On average, losing players save a higher amount of break points than winners do. In fact they save an average of **1.29 more** break points than winning players.
  - This stat does not mean that saving break points is bad, but rather, this happens because losing players ultimately face more break points.

- 1st serve %
  - The first serve percentage of winning players is on average **two percentage points** higher than the first serve percentages of losing players.

- 2nd serve points won %
  - On average, winning players win **56%** of their 2nd serve points which is **11 percentage points higher** than the mean 2nd serve percentage of losing players.

- Break Points won %
  - The mean break point conversion rate of winning players is **49%**, about **17%** points higher than the mean break point conversion rate of losing players.

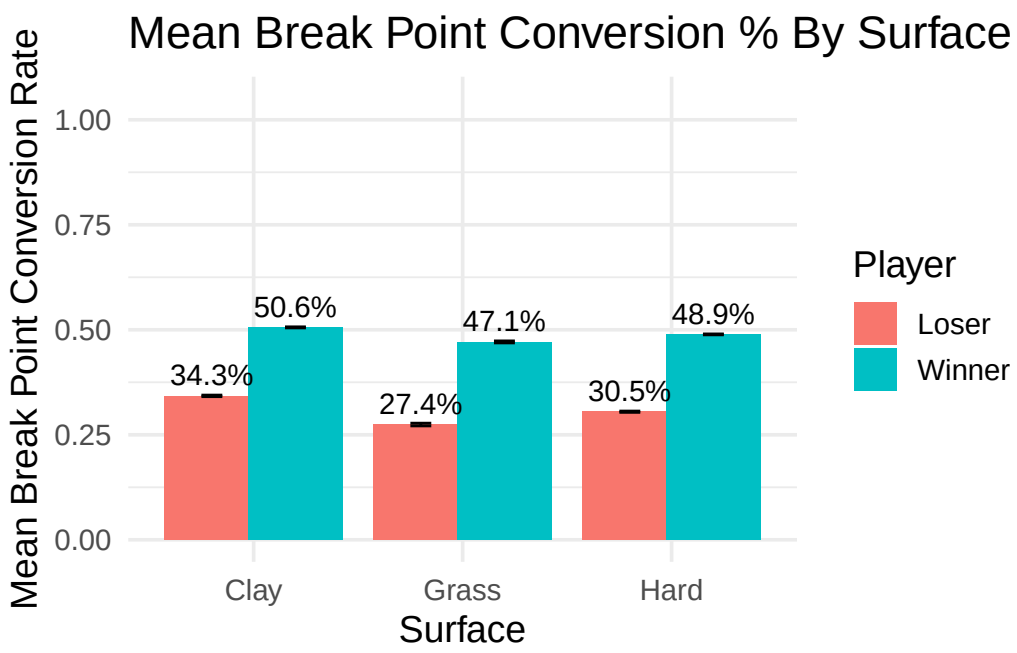
The summary statistics imply that 1st serve win percentage, 2nd serve win percentage, and break point conversion percentage are on average much higher for winners than losing players. Note, that these summary statistics compare the mean values of winning and losing players across all surfaces and does not provide any insight as to how the surface of a match might affect the statistics of winners and losers. To address this, I have included two graphs below which allow us to compare how the mean values of the match statistics between winning and losing players depend on the surface that matches are played on. Graphing all of these statistics in this paper would be too much, so I have only included the graphs for 1st serve points won % and break point conversion % as they illustrate the influence of the surfaces well.



1st Serve Points Won % Graph Interpretations

- Winner's mean 1st serve points won percentage
    1. **Grass: 79.3%**
    2. Hard: 77.4%
    3. Clay: 73.5%
- Loser's mean 1st serve points won percentage
    1. **Grass: 69.2%**
    2. Hard: 66.9%
    3. Clay: 62.5%
- First serve points won percentage is highest on grass and lowest on clay.
- This makes sense because as discussed in the introduction, grass courts favor bigger serves, so landing a fast first serve limits the reaction time of the returner and increases the likelihood of the server winning that point.
- For clay however, the returner has more time to react to a fast first serve, so the chances of the server winning a point on their first serve will decrease slightly.
- Also, notice that the difference in mean 1st serve points won % between winning players and losing players is always roughly between 10-11 percentage points no matter what the surface is.

- This suggests that gaining an advantage over your opponent for the percentage of first serve points won is equally important across all surfaces.



Break Point Conversion Graph Interpretations

- Winner's mean break point conversion rate

  1. Clay: **50.6%**
  2. Hard:48.9%
  3. Grass: 47.1%

- Loser's mean break point conversion rate

  1. Clay: **34.3%**
  2. Hard: 30.5%
  3. Grass: 27.4%

- Mean break point conversion rate is the highest on clay courts and the lowest on grass courts which makes sense.
- Clay slows the speed of the ball down, giving returners more time to react to serves and leads to longer rallies which help the returner out.
- Grass causes the ball to move more quickly through the court, reducing the reaction time for returners and leading to quicker points which favors the server.

- Note: The greatest difference in mean break point conversion rate between winning and losing players is on **grass**, with winning players converting on average almost 20 percentage points more of their break point opportunities than losing players.
- This suggests that gaining an advantage over an opponent in break point conversion rate is more important on grass because there are typically fewer break point opportunities.

We will now discuss the regression results from the logistic regressions ran using the estimating equation displayed earlier.

Regression Results

Before analyzing the coefficient estimates produced from the logistic regressions, I will first ensure validity of the models by comparing their confusion matrix metrics.

Model Validity

As mentioned in the data preparation section, I split each of the surface data frames into 80% for training and the remaining 20% for testing the validity of the logistic models to ensure their accuracy. Below is a table comparing the metrics that were calculated using R studio's `confusionMatrix()` command from the `caret` package:

Table 2: Confusion Matrix Metric Comparison

Model	Accuracy	Sensitivity	Specificity	PosPredictedValue	NegPredictedValue
Clay	0.92	0.92	0.92	0.92	0.92
Grass	0.91	0.91	0.91	0.90	0.91
Hard	0.91	0.91	0.91	0.91	0.91

We can see from the table above that all three of the logistic regression models classify over 90% of the matches in the testing data as having the correct winner, meaning that the models are **generally valid**. I have also included interpretations of the other metrics included in the table.

- **Sensitivity (true positive rate)**: Out of the matches in which player A actually won a match, the models correctly classified player A as the winner in over 90% of those matches.
- **Specificity (true negative rate)**: Out of the matches that player B actually won, the models correctly classified player B as the winner in over 90% of those matches.
- **Positive Predicted Value**: Out of the matches in which the models identified player A as the winner, over 90% of those classifications were accurate.

- **Negative Predicted Value**: Out of the matches that the models identified player B as the winner, over 90% of those classifications were accurate.

With the validity of the logistic regression models confirmed, we can now interpret the coefficient estimates produced from these models.

Below is a table with the coefficient estimates of the logistic regressions for clay, grass, and hard courts.

Table 3: Logistic Regression Results for Every Surface

	Dependent variable:		
	Match Result		
	Clay	Grass	Hard
	(1)	(2)	(3)
Ace Differential	0.62***	0.44***	0.49***
	(0.03)	(0.04)	(0.02)
1st serves In % differential	0.20***	0.33***	0.30***
	(0.03)	(0.06)	(0.02)
2nd Serve Points Won % Diff	1.40***	1.33***	1.45***
	(0.04)	(0.08)	(0.04)
Break Points Created Difference	2.86***	2.42***	2.63***
	(0.05)	(0.08)	(0.04)
Break Points Conversion % Diff	2.77***	2.26***	2.35***
	(0.05)	(0.08)	(0.04)
Constant	-0.08***	-0.09**	-0.04*
	(0.03)	(0.04)	(0.02)
Observations	23,207	7,291	35,446
Log Likelihood	-4,974.95	-1,632.35	-7,920.53
Akaike Inf. Crit.	9,961.90	3,276.70	15,853.05
Note:		*p<0.1; **p<0.05; ***p<0.01	

Coefficient Interpretations

I will interpret the coefficient estimates for each variable seen in Table 2 for only the clay regression since the interpretations follow the same format across all of the surfaces. I will also acknowledge some of the differences in magnitudes for a select few of the match statistics between surfaces and identify what those differences suggest about the data.

Sign

- The sign of all of the coefficient estimates for the **independent variables** are **positive** across all of the surfaces. This is expected since they are all statistics that players want to have an advantage in over their opponents.
- When the specified match statistic differences are positive for player A (meaning they have an advantage over player B for a specified statistics) their likelihood of winning increases.
- The **constant terms** are the only estimates that are **negative**, however each of these estimates are relatively close to zero. This means that when the difference in these statistical categories are zero (A and B have equal values) the likelihood of whether A or B wins the match is relatively equal.

Statistical Significance

- All of the coefficient estimates for the independent variables are statistically significant at a significance level of $\alpha = 0.01$ because of the large number of observations in our sample.
- This means that the coefficient estimates are unlikely to have occurred by random chance, meaning we are confident that outperforming opponents in these statistical categories does have a positive effect on winning matches.

Magnitude

- **Ace Differential (Clay)**
  - $\beta_1 = 0.62$
  - Since the independent variables are standardized, the coefficient interpretation becomes slightly different. The one unit change in a particular match statistic in this case is a 1 standard deviation change in the match statistics, allowing for direct comparison between count and percentage variables. A 1 standard deviation change for a specified variable just refers to a change that is more than significant.
  - Therefore, a 1 standard deviation increase in the difference between the number of aces player A hits compared to player B leads to an increase in the log-odds of player A winning by about 0.62

- Alternatively, taking $e^{0.62}$ gives us the odds of player A winning. So the odds of player A winning increases by a factor of about 1.86 from a 1 standard deviation increase in the difference in player A's ace total compared to B's.
- An increase by a factor of 1.86 just means that the odds of player A winning increase by 1.86 times.
- The coefficient estimates for ace differential are both positive and statistically significant across all surfaces.

- **1st Serve Percentage Difference (Clay)**
  - $\beta_2 = 0.20$
  - The odds of player A winning a match increase by a factor of $e^{0.20} = 1.22$ from a 1 standard deviation increase in the difference between Player A's and player B's 1st serve percentage on clay, holding all else constant.

- **2nd Serve Points Won percentage difference (Clay)**
  - $\beta_3 = 1.40$
  - For every 1 standard deviation increase in the difference between player A's and player B's 2nd serve points won percentage, holding all else constant, the odds of player A winning increase by a factor of $e^{1.40} = 4.055$.
  - This means that player A's odds of winning increase about 4 times if they can increase their 2nd serve points won % compared to player B by 1 standard deviation.

- **Break Points Created Difference (Clay)**
  - $\beta_4 = 2.86$
  - For every 1 standard deviation increase in the difference between the number of break points player A generates compared to player B, holding all else equal, the odds of player A winning increase by a factor of $e^{2.86} = 17.46$ which is a substantial amount.
  - This means that the odds of player A winning increase by about 17 times for a 1 standard deviation increase in the difference between how many break points they generate compared to their opponent.

- **Break points Conversion Rate Difference (Clay)**
  - $\beta_5 = 2.77$

- The odds of player A winning a match increases by a factor of $e^{2.77} = 15.96$ for every additional 1 standard deviation increase in the difference between player A's break point conversion rate and player B's conversion rate which is also a significant amount.
- Player A's odds of winning increase by 15.96 times for a more than significant increase in their break point conversion rate compared to their opponent's.

Key takeaways

- The match statistics with the strongest magnitudes across all surfaces are the two break point variables: **break points created difference** and **break points conversion percentage difference**.
- 2nd serve points won differential also has a relatively strong magnitude in effecting the outcome of tennis matches.
- For the ace differential coefficient estimates, the clay regression has the **strongest magnitude** compared to the other surfaces. This suggests that a 1 standard deviation increase in the difference in player A's ace total compared to player B's ace total will lead to a more significant increase in the odds of player A winning on clay than the other surfaces.
- This may seem counterintuitive at first glance, however, since aces are rarer on clay (because of the slow court speeds), then getting a significant ace advantage over an opponent on clay will have more of an impact on winning. Whereas, aces are more frequent on grass and hard courts, so getting an advantage in aces will not have as much of an effect on the match outcome.
- For the break point variables, clay also has a stronger magnitude compared to grass and hard courts.
- This result seems to have the opposite intuition as I explained for why the clay ace differential estimate has the strongest magnitude. I would have expected, since break points are much rarer on the faster surfaces like grass and hard courts, that gaining a significant advantage over the opponent in break points created and conversion rate would have more of an effect on winning on the fast surfaces. This is precisely the conclusion I made in the summary statistics section where we compared the mean break point conversion rates between match winners and losers.
- However, the regression results indicate that gaining an advantage over your opponent in both break points converted and break point conversion rate is more significant on clay.
- This likely is the case because the "recipe for success" on clay is to generate longer rallies on the opponent's serve and to create as many break point chances as possible. Hence, the player who succeeds in this will increase their chances of winning more significantly

on clay. Whereas on grass, players can more heavily rely on their serve for success if they fail to create break opportunities on their opponent's serve.

Conclusions

- This paper analyzed a tennis dataset consisting of over 82,000 professional tennis matches played on the ATP Tour from 1991-2022.
- The dataset included information on several serving and returning match statistics, as well as additional variables that I created from the existing data.
- I built a logistic regression for each surface (clay, grass, and hard) to identify the match statistics that had the greatest marginal effect on the outcomes of tennis matches, as well as to understand the role that the surfaces play in these findings. In the results section, we developed the following conclusions:
  - Gaining a more than significant advantage over the opponent in the number of **break point chances created** and in **break point conversion rate** is associated with winning matches the most.
  - Gaining an advantage in the **percentage of 2nd serve points won** is also fairly significantly associated with winning matches.
  - Gaining a significant advantage over the opponent in aces hit is **more significantly associated** with winning on **clay** than the other surfaces.
  - Gaining a significant advantage over the opponent in the break point statistics is also **more significantly associated** with winning on **clay** than the other surfaces.

Future Work

Despite the insightful conclusions drawn from this research, there are still several points of research that I think are worth exploring for the future.

- **Explore the effect of more match stats**. The dataset I used did not have the following key match stats: winners hit, unforced errors, the number of long rallies (9+ shots) each player won, return winners, etc. Investigating these match stats would be very interesting and I think generate more insightful findings about how tennis matches are decided.
- **Compare results for different eras**: The dataset included matches spanning 30 years from 1991-2022. I think it would be very interesting to compare regression results for each decade to see if certain match statistics have gained more or less importance in deciding outcomes in recent years. For example, players serve and volleyed more frequently during the 90s compared to the players of today. This could potentially mean that 1st serve percentage may have been more important in previous generations since the serve an volley strategy is normally utilized on only 1st serves.

- **Compare the results for best of 3 set matches and best of 5 set matches**: In best of 5 set matches, players may seek to attack more frequently to reduce the length of rallies more so than for shorter best of 3 set matches. This potentially could play a factor in whether serving or returning statistics are more important for a particular match.
- **Develop a model that predicts the outcomes of future matches**: This was my original goal for this project, however it would be much more difficult because it would require lots of information. For example, to predict the outcome of a specific match, the model would need to know both players' historical statistic's, their success on that specific surface, their past success at that tournament, their recent match results, their head to head record, and so on.

References

ATP Staff. (2025, June 10). *What is the 2025 ATP Tour calendar?* ATP Tour. Retrieved December 11, 2025, from https://www.atptour.com/en/news/what-is-the-2025-atp-tour-calendar

Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables. R package version 5.2.3. https://CRAN.R-project.org/package=stargazer

Kokta, M. (2020, April 18). *Predicting ATP tennis match outcomes using serving statistics.* Medium. Retrieved December 11, 2025, from https://medium.com/swlh/predicting-atp-tennis-match-outcomes-using-serving-statistics-cde03d99f410

Manikandan, S. (n.d.). *ATP matches (ATPdata)* [Data set]. Kaggle. Retrieved December 11, 2025, from https://www.kaggle.com/datasets/sijovm/atpdata

Appendix

Loading in the Data and removing any NA or Incorrect Values

```
# Clear environment and load all necessary R packages
rm(list=ls())
library(tidyverse)
library(stargazer)
library(DataExplorer)
library(MASS)

# Import full tennis dataset
tennis_temp <-
  read.csv("/Users/teddykelly/Downloads/atp_matches_till_2022.csv")

# Filter the dataset to only contain the matches with match statistics
tennis_df <- tennis_temp |> dplyr::filter(!is.na(w_df))


# Delete any variable columns that are not important for the regressions

df <- tennis_df |> dplyr::select(- winner_seed, -winner_ht,
                                      -loser_seed, -loser_ht, -minutes,
                                      -winner_ioc, -winner_entry,
                                      -loser_ioc, loser_entry
                                      )

# Remove any incorrect observations (numerical variables with negative values)
quantitative_cols <- sapply(df, is.numeric) | sapply(df, is.integer)
quantitative_var_names <- names(df)[quantitative_cols]
df <- df |> filter(across(all_of(quantitative_var_names), ~ . >= 0))

# Remove any matches that ended in a retirement
df <- df |> filter(!grepl("RET", score))

# Remove any matches that were played on Carpet Courts
df <- df |> filter(surface != "Carpet")
```

Winsorize Win Loss Variables to Control for Outliers (Commented Out)

```
#library(DescTools)

# Variables to winsorize
#winsor_vars <- c(
 # "w_ace", "w_df", "w_svpt", "w_1stIn", "w_1stWon", "w_2ndWon",
  #"w_SvGms", "w_bpSaved", "w_bpFaced",
  #"l_ace", "l_df", "l_svpt", "l_1stIn", "l_1stWon", "l_2ndWon",
  #"l_SvGms", "l_bpSaved", "l_bpFaced",
  #"winner_age", "winner_rank", "winner_rank_points",
  #"loser_age", "loser_rank", "loser_rank_points"
#)

#winsorize_custom <- function(x, p = 0.01) {
 # lo <- quantile(x, p, na.rm = TRUE)
  #hi <- quantile(x, 1 - p, na.rm = TRUE)
  #x[x < lo] <- lo
  #x[x > hi] <- hi
  #return(x)
#}


#df <- df %>%
 # group_by(surface) %>%
  #mutate(
   # across(
    #  all_of(winsor_vars),
     # ~ winsorize_custom(.x, p = 0.01),
      #.names = "{.col}"
    #)
  #) %>%
  #ungroup()

#df <- as.data.frame(df)
```

Create New Variables of Interest for Winners and Losers

```
# First serve percentage winners and losers
df$w_1stInPct <- df$w_1stIn / df$w_svpt
```

```r
df$l_1stInPct <- df$l_1stIn / df$l_svpt

# First serve percentage won winners and losers
df$w_1stWonPct <- df$w_1stWon / df$w_1stIn
df$l_1stWonPct <- df$l_1stWon / df$l_1stIn

# 2nd serve % won winners and losers
df$w_2ndWonPct <- df$w_2ndWon / (df$w_svpt - df$w_1stIn)
df$l_2ndWonPct <- df$l_2ndWon / (df$l_svpt - df$l_1stIn)

# Break point save percentage
df$w_bpSavedPct <- df$w_bpSaved / df$w_bpFaced
df$l_bpSavedPct <- df$l_bpSaved / df$l_bpFaced
df$w_bpSavedPct[is.na(df$w_bpSavedPct)] <- 1
df$l_bpSavedPct[is.na(df$l_bpSavedPct)] <- 1

# Break points converted
df$w_bpConv <- df$l_bpFaced - df$l_bpSaved
df$l_bpConv <- df$w_bpFaced - df$w_bpSaved


# Break Point conversion rate winners and losers
df$w_bpConvPct <- (1 - df$l_bpSavedPct)
df$l_bpConvPct <- (1 - df$w_bpSavedPct)

# Create return points won
df$w_returnPtsWon <- (df$l_svpt - (df$l_1stWon + df$l_2ndWon))
df$l_returnPtsWon <- (df$w_svpt - (df$w_1stWon + df$w_2ndWon))

# Create return points won percentage
df$w_returnPtsWonPct <- df$w_returnPtsWon / df$l_svpt
df$l_returnPtsWonPct <- df$l_returnPtsWon / df$w_svpt


# Remove any matches with missing values
df <- na.omit(df)

# Filter dataset to only contain 2nd serve win % less than or equal to 1
df <- df|> dplyr::filter(w_2ndWonPct <= 1)
```

Summary Statistics comparing Winners and Losers Match Statistics

```
# Select only the desired statistics to appear in the summary stats

selected_vars <- df[, c("w_ace", "l_ace", "w_df", "l_df",
                        "w_bpSaved", "l_bpSaved",
                        "w_bpFaced", "l_bpFaced",
                        "w_bpConv", "l_bpConv",
                        "w_1stInPct", "l_1stInPct",
                        "w_1stWonPct", "l_1stWonPct",
                        "w_2ndWonPct", "l_2ndWonPct",
                        "w_bpSavedPct", "l_bpSavedPct",
                        "w_bpConvPct", "l_bpConvPct",
                        "w_returnPtsWon", "l_returnPtsWon",
                        "w_returnPtsWonPct",
                        "l_returnPtsWonPct")]

# Create Clean Variable names for summary stats table
stats_labels <- c(
  "Winner Aces",
  "Loser  Aces",
  "Winner Double Faults",
  "Loser  Double faults",
  "Winner Break Points Saved",
  "Loser  Break Points Saved",
  "Winner Break Points Faced",
  "Loser  Break Points Faced",
  "Winner Break Points Converted",
  "Loser Break Points Converted",
  "Winner 1st Serve %",
  "Loser  1st Serve %",
  "Winner 1st Serve Points Won %",
  "Loser  1st Serve Points Won %",
  "Winner 2nd Serve Points Won %",
  "Loser  2nd Serve Points Won %",
  "Winner Break Points Saved %",
  "Loser  Break Points Saved %",
  "Winner Break Points Converted %",
  "Loser  Break Points Converted %",
  "Winner Return Points Won",
  "Loser  Return Points Won",
  "Winner Return Points Won %",
```

```
  "Loser  Return Points Won %"
)
stargazer(selected_vars, type = "text",
          covariate.labels = stats_labels,
          omit.summary.stat = c("N"),
          notes = "N = 82,411 Matches",
          digits = 2, title = "Table 1: Winners vs Losers Summary Statistics")
```

Randomly Assign each Player as A or B

```
# Add column that indicates whether player A or player B won

set.seed(123)
df$swap <- rbinom(nrow(df), 1, 0.5)

# Create Win loss binary variable (1 indicates that player A won
# and 0 indicates player B won)
df$player_A <- ifelse(df$swap == 1, df$winner_name, df$loser_name)
df$player_B <- ifelse(df$swap == 0, df$winner_name, df$loser_name)

# Creat match_outcome variable
df$match_result <- ifelse(df$swap == 1, 1, 0)

# Rankings
df$A_rank <- ifelse(df$swap == 1, df$winner_rank, df$loser_rank)
df$B_rank <- ifelse(df$swap == 0, df$winner_rank, df$loser_rank)

df$A_rank_points <- ifelse(df$swap == 1, df$winner_rank_points,
                           df$loser_rank_points)
df$B_rank_points <- ifelse(df$swap == 0, df$winner_rank_points,
                           df$loser_rank_points)

# Handedness
df$A_hand <- ifelse(df$swap == 1, df$winner_hand, df$loser_hand)
df$B_hand <- ifelse(df$swap == 0, df$winner_hand, df$loser_hand)

# Aces
df$A_ace <- ifelse(df$swap == 1, df$w_ace, df$l_ace)
df$B_ace <- ifelse(df$swap == 0, df$w_ace, df$l_ace)
```

```r
# Double faults
df$A_df <- ifelse(df$swap == 1, df$w_df, df$l_df)
df$B_df <- ifelse(df$swap == 0, df$w_df, df$l_df)

# Total service points
df$A_svpt <- ifelse(df$swap == 1, df$w_svpt, df$l_svpt)
df$B_svpt <- ifelse(df$swap == 0, df$w_svpt, df$l_svpt)

# First serve in
df$A_1stIn <- ifelse(df$swap == 1, df$w_1stIn, df$l_1stIn)
df$B_1stIn <- ifelse(df$swap == 0, df$w_1stIn, df$l_1stIn)

# First serve points won
df$A_1stWon <- ifelse(df$swap == 1, df$w_1stWon, df$l_1stWon)
df$B_1stWon <- ifelse(df$swap == 0, df$w_1stWon, df$l_1stWon)

# Second serve points won
df$A_2ndWon <- ifelse(df$swap == 1, df$w_2ndWon, df$l_2ndWon)
df$B_2ndWon <- ifelse(df$swap == 0, df$w_2ndWon, df$l_2ndWon)

# Service games
df$A_SvGms <- ifelse(df$swap == 1, df$w_SvGms, df$l_SvGms)
df$B_SvGms <- ifelse(df$swap == 0, df$w_SvGms, df$l_SvGms)

# Break points saved
df$A_bpSaved <- ifelse(df$swap == 1, df$w_bpSaved, df$l_bpSaved)
df$B_bpSaved <- ifelse(df$swap == 0, df$w_bpSaved, df$l_bpSaved)

# Break points faced
df$A_bpFaced <- ifelse(df$swap == 1, df$w_bpFaced, df$l_bpFaced)
df$B_bpFaced <- ifelse(df$swap == 0, df$w_bpFaced, df$l_bpFaced)

# Break Points generated on return
df$A_bpCreated <- ifelse(df$swap == 1, df$l_bpFaced, df$w_bpFaced)
df$B_bpCreated <- ifelse(df$swap == 0, df$l_bpFaced, df$w_bpFaced)

# Break Point Converted on return
df$A_bpConv <- ifelse(df$swap == 1, df$w_bpConv, df$l_bpConv)
df$B_bpConv <- ifelse(df$swap == 0, df$w_bpConv, df$l_bpConv)

# First serves made percentage
df$A_1stInPct <- df$A_1stIn / df$A_svpt
```

```r
df$B_1stInPct <- df$B_1stIn / df$B_svpt

#Also going to create a first serve win % variable
df$A_1stWonPct <- df$A_1stWon / df$A_1stIn
df$B_1stWonPct <- df$B_1stWon / df$B_1stIn

# 2nd Serve Win percentage
df$A_2ndWonPct <- df$A_2ndWon / (df$A_svpt - df$A_1stIn)
df$B_2ndWonPct <- df$B_2ndWon / (df$B_svpt - df$B_1stIn)

# Break point save percentage
df$A_bpSavedPct <- df$A_bpSaved / df$A_bpFaced
df$B_bpSavedPct <- df$B_bpSaved / df$B_bpFaced
df$A_bpSavedPct[is.na(df$A_bpSavedPct)] <- 1
df$B_bpSavedPct[is.na(df$B_bpSavedPct)] <- 1

# Break Point Conversion precetage
df$A_bpConvPct <- (1 - df$B_bpSavedPct)
df$B_bpConvPct <- (1 - df$A_bpSavedPct)

# Return Points Won
df$A_returnPtsWon <-
  ifelse(df$swap == 1, df$w_returnPtsWon, df$l_returnPtsWon)

df$B_returnPtsWon <-
  ifelse(df$swap == 0, df$w_returnPtsWon, df$l_returnPtsWon)


# Return Points Won percentage
df$A_returnPtsWonPct <-
  ifelse(df$swap == 1, df$w_returnPtsWonPct, df$l_returnPtsWonPct)
df$B_returnPtsWonPct <-
  ifelse(df$swap == 0, df$w_returnPtsWonPct, df$l_returnPtsWonPct)
```

Create Difference Variables Between Player A and B (A-B)

```r
# ranking difference
df$rank_diff <- df$A_rank - df$B_rank

# Ranking points difference
```

```
df$rank_points_diff <- df$A_rank_points - df$B_rank_points

# Ace differential
df$ace_diff <- df$A_ace - df$B_ace

# Double Fault differential
df$df_diff <- df$A_df - df$B_df

# Service Points differential
df$svpt_diff <- df$A_svpt - df$B_svpt

# 1st serves in differentail
df$diff_1stIn <- df$A_1stIn - df$B_1stIn

# 1st serve points won differential
df$diff_1stWon <- df$A_1stWon - df$B_1stWon

# 2nd serve points won differential
df$diff_2ndWon <- df$A_2ndWon - df$B_2ndWon

# Service Games Differential
df$SvGms_diff <- df$A_SvGms - df$B_SvGms

# Break Points Saved Differential
df$bpSaved_diff <- df$A_bpSaved - df$B_bpSaved

# Break Points faced differential
df$bpFaced_diff <- df$A_bpFaced - df$B_bpFaced

# Break points created differential
df$bpCreated_diff <- df$A_bpCreated - df$B_bpCreated

# Break points won differential
df$bpConv_diff <- df$A_bpConv - df$B_bpConv

# 1st serve percentage difference
df$diff_1stInPct <-
  (df$A_1stInPct - df$B_1stInPct)

# 1st serve points won % difference
df$diff_1stWonPct <-
  (df$A_1stWonPct - df$B_1stWonPct)
```

```r
# 2nd serve points won % difference
df$diff_2ndWonPct <-
  (df$A_2ndWonPct - df$B_2ndWonPct)

# Break Points saved % difference
df$bpSavedPct_diff <-
  (df$A_bpSavedPct - df$B_bpSavedPct)

# Break point conversion rate difference
df$bpConvPct_diff <-
  (df$A_bpConvPct - df$B_bpConvPct)

# return points won difference
df$returnPtsWon_diff <-
  (df$A_returnPtsWon - df$B_returnPtsWon)

# Return points won % difference
df$returnPtsWonPct_diff <-
  (df$A_returnPtsWonPct - df$B_returnPtsWonPct)
```

Standardize the Difference Variables Before Regressions

```r
# Creating vector that contains all of the difference variable names
diff_vars <- grep("^diff|_diff$", names(df), value = TRUE)

# Standardizing the diffrence variables
df[diff_vars] <- scale(df[diff_vars])
```

Create Separate Datasets for each Surface

```r
df_clay <- df |> dplyr::filter(surface == "Clay")
df_grass <- df |> dplyr::filter(surface == "Grass")
df_hard <- df |> dplyr::filter(surface == "Hard")
```

Create Training and testing data for each surface

```
# CLAY TRAINING and TESTING DATA
set.seed(123)
# Generate a vector of indices for the training set

train_index_clay <-
  sample(x = nrow(df_clay),  size = round(0.8 * nrow(df_clay)))

train_clay <- df_clay[train_index_clay, ]

test_clay  <- df_clay[-train_index_clay, ]

# GRASS TRAINING and TESTING DATA
train_index_grass <-
  sample(x = nrow(df_grass),  size = round(0.8 * nrow(df_grass)))

train_grass <- df_grass[train_index_grass, ]

test_grass <- df_grass[-train_index_grass, ]


# HARD TRAINING and TESTING DATA
train_index_hard <-
  sample(x = nrow(df_hard),  size = round(0.8 * nrow(df_hard)))

train_hard <- df_hard[train_index_hard, ]

test_hard <- df_hard[-train_index_hard, ]
```

Clay Regressions

```
clay_reg1 <- glm(match_result ~ ace_diff +
    diff_1stInPct + diff_2ndWonPct +
      bpCreated_diff + bpConvPct_diff,
    family = "binomial", data = train_clay)

stepAIC(clay_reg1,
        direction = "backward")
```

Grass Regressions

```
grass_reg1 <- glm(match_result ~ ace_diff +
    diff_1stInPct + diff_2ndWonPct +
      bpCreated_diff + bpConvPct_diff,
    family = "binomial", data = train_grass)

stepAIC(grass_reg1,
        direction = "backward")
```

Hard Regressions

```
hard_reg1 <- glm(match_result ~ ace_diff +
    diff_1stInPct + diff_2ndWonPct +
      bpCreated_diff + bpConvPct_diff,
    family = "binomial", data = train_hard)

stepAIC(hard_reg1,
        direction = "backward")
```

Create Bar Graphs Comparing Specific Statistics Across Surfaces

```
# Bind together all the training dataframes together
df_all <- dplyr::bind_rows(train_clay, train_grass, train_hard)

# FIRST SERVE WIN PERCENTAGE

# Reshape df_all into long format
plot_1stWonPct <- df_all |>
  dplyr::select(surface, w_1stWonPct, l_1stWonPct) |>
  tidyr::pivot_longer(
    cols = c(w_1stWonPct, l_1stWonPct),
    names_to = "player_type",
    values_to = "first_serve_won_pct"
  ) |>
  dplyr::mutate(player_type = ifelse(player_type == "w_1stWonPct",
                                     "Winner", "Loser"))
```

```r
# Compute the means for labeling
label_df1 <- plot_1stWonPct |>
  group_by(surface, player_type) |>
  summarise(mean_pct = mean(first_serve_won_pct, na.rm = TRUE)) |>
  ungroup()

# Plot with labels
ggplot(plot_1stWonPct, aes(x = surface, y = first_serve_won_pct,
                           fill = player_type)) +
  stat_summary(fun = mean, geom = "bar",
               position = position_dodge(width = 0.8)) +

  # Add error bars
  stat_summary(fun.data = mean_se, geom = "errorbar",
               position = position_dodge(width = 0.8),
               width = 0.2) +

  # Add percentage labels
  geom_text(data = label_df1,
            aes(x = surface,
                y = mean_pct,
                label = paste0(round(mean_pct * 100, 1), "%"),
                group = player_type),
            position = position_dodge(width = 0.8),
            vjust = -0.5,
            size = 4) +

  labs(
    title = "Mean 1st Serve Points Won % by Winners vs
    Losers Across Surfaces",
    x = "Surface",
    y = "Mean 1st Serve Points Won %",
    fill = "Player"
  ) +
  theme_minimal(base_size = 14) +
  ylim(0, 1.05)
```

```r
# BREAK POINT CONVERSION RATE
plot_bpConvPct <- df_all |>
  dplyr::select(surface, w_bpConvPct, l_bpConvPct) |>
  tidyr::pivot_longer(
```

```
    cols = c(w_bpConvPct, l_bpConvPct),
    names_to = "player_type",
    values_to = "break_point_conversion_pct"
  ) |>
  dplyr::mutate(player_type = ifelse(player_type == "w_bpConvPct",
                                     "Winner", "Loser"))

# Compute the means for labeling
label_df2 <- plot_bpConvPct |>
  group_by(surface, player_type) |>
  summarise(mean_pct = mean(break_point_conversion_pct, na.rm = TRUE)) |>
  ungroup()

# Plot with labels
ggplot(plot_bpConvPct,
       aes(x = surface, y = break_point_conversion_pct,
           fill = player_type)) +
  stat_summary(fun = mean, geom = "bar",
               position = position_dodge(width = 0.8)) +

  # Add error bars
  stat_summary(fun.data = mean_se, geom = "errorbar",
               position = position_dodge(width = 0.8),
               width = 0.2) +

  # Add percentage labels
  geom_text(data = label_df2,
            aes(x = surface,
                y = mean_pct,
                label = paste0(round(mean_pct * 100, 1), "%"),
                group = player_type),
            position = position_dodge(width = 0.8),
            vjust = -0.5,
            size = 4) +

  labs(
    title = "Mean Break Point Conversion % By Surface",
    x = "Surface",
    y = "Mean Break Point Conversion Rate",
    fill = "Player"
  ) +
  theme_minimal(base_size = 14) +
```

```
  ylim(0, 1.05)
```

Regression Results

```
cov_labels <- c("Ace Differential",
                                "1st serves In % differential",
                                "2nd Serve Points Won % Diff",
                  "Break Points Created",
                                "Break Points Conversion % Diff"
                                )
stargazer(clay_reg1, grass_reg1, hard_reg1, type = "text",
          title = "Table 3: Logistic Regression Results for Every Surface",
          digits = 2, column.labels = c("Clay", "Grass", "Hard"),
          dep.var.labels = c("Match Result"),
          covariate.labels = cov_labels)
```

Testing Model Validity on Testing Data with Confusion Matrices

```
library(caret)
# Clay court Confusion Matrix
test_clay$pred_log <- round(predict(clay_reg1, newdata = test_clay,
                                type = "response"), 2)

# Threshold is 0.5
test_clay$pred_outcome <- ifelse(test_clay$pred_log >= 0.5, 1, 0)

# Confusion Matrix for Clay Data
clay_conf <- confusionMatrix(as.factor(test_clay$match_result),
                as.factor(test_clay$pred_outcome),
                positive = "1")

# Grass court predictions
test_grass$pred_log <- round(predict(grass_reg1, newdata = test_grass,
                                type = "response"), 2)

# Threshold is 0.5
test_grass$pred_outcome <- ifelse(test_grass$pred_log >= 0.5, 1, 0)
```

```r
# Confusion Matrix for Grass Data
grass_conf <- confusionMatrix(as.factor(test_grass$match_result),
                as.factor(test_grass$pred_outcome),
                positive = "1")


# Hard Court predictions

test_hard$pred_log <- round(predict(hard_reg1, newdata = test_hard,
                                type = "response"), 2)

# Threshold is 0.5
test_hard$pred_outcome <- ifelse(test_hard$pred_log >= 0.5, 1, 0)

# Confusion Matrix for Hard Data
hard_conf <- confusionMatrix(as.factor(test_hard$match_result),
                as.factor(test_hard$pred_outcome),
                positive = "1")




# Create table comparing the comfusion matrix metrics
#for each logistic regression

confusion_df <- data.frame(Model = c("Clay", "Grass ","Hard"),
                           Accuracy = c(clay_conf$overall["Accuracy"],
                                        grass_conf$overall["Accuracy"],
                                        hard_conf$overall["Accuracy"]),
                           Sensitivity = c(clay_conf$byClass["Sensitivity"],
                                           grass_conf$byClass["Sensitivity"],
                                           hard_conf$byClass["Sensitivity"]),
                           Specificity = c(clay_conf$byClass["Specificity"],
                                           grass_conf$byClass["Specificity"],
                                           hard_conf$byClass["Specificity"]),
                           PosPredictedValue =
                             c(clay_conf$byClass["Pos Pred Value"],
                               grass_conf$byClass["Pos Pred Value"],
                               hard_conf$byClass["Pos Pred Value"]),
                           NegPredictedValue =
                             c(clay_conf$byClass["Neg Pred Value"],
                               grass_conf$byClass["Neg Pred Value"],
```

```
                                hard_conf$byClass["Neg Pred Value"]))

kableExtra::kable(confusion_df, digits = 2)
```