Investigating the probability that the New Orleans Saints win the 2026 NFL Super Bowl

24 pages

Introduction

The most popular sports competition in the United States is, and has been for several decades, the NFL. The championship sees 32 teams from around the country compete against each other in two stages. The first, the "regular season", consists of each team playing 17 games (as of 2021[i]) over 18 weeks. After this, the teams' win-loss ratios are compared within 8 divisions: National Football Conference (NFC) North, South, East and West, and American Football Conference (AFC) North, South, East and West. The top team from each division, as well as the top three remaining teams from each conference, then enter the "playoffs", which are a series of 13 elimination-bracket games over four rounds. The last of these 13 games is the Super Bowl, which is played on the second Sunday of every February.

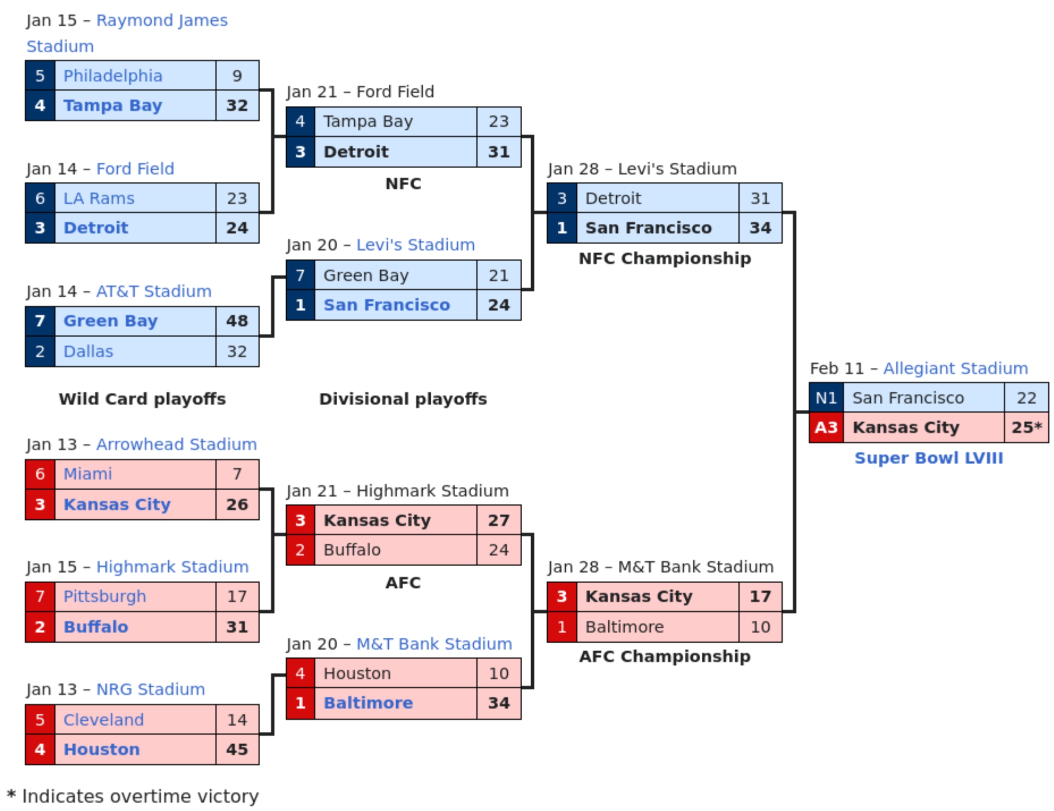
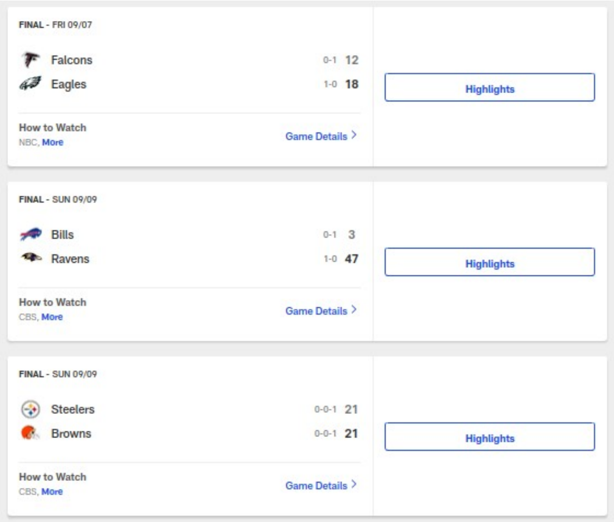


Figure 1.1: 2024 NFL playoffs bracket.[ii]

This investigation aims to model both the regular season and the postseason of the NFL and find the easiest way for my home team, the New Orleans Saints, to win the Super Bowl. The Saints are by no means the best team in the NFL, and they have only made it to the playoffs in 10 of the last 30 seasons[iii]. Despite only having won 47% of their lifetime regular season games and 44% of their playoff games as of 2024, as compared to the 55% and 63% of one of the most notable teams, the New England Patriots[iv], I believe that the Saints still have a decent chance of winning the Super Bowl. To test this, I will interpret game data from the last 7 years, and season record data from the last 25 years, to construct a model of the 2025 NFL season and find the easiest way for the Saints to win their second Super Bowl.

Data collection

The first steps for my investigation were to collect data for each team in previous years. I began by collecting data of each team's games played since 2018, recording the outcome and score difference. As I needed to record a total of around 3000 game results, efficiency was critical at this step in order to avoid spending too much time recording these games. As well as this, it was important to avoid making errors when recording games, as it would be hard to tell which of the 3000 games were incorrect if I did make errors. To combat these issues, I used tables which were available on Wikipedia, as they laid the information out much more densely than the NFL website, saving time, and were arranged by team, instead of by each week of the season. While Wikipedia can be an imperfect source, every game result on each page had a link to the game's analysis on the NFL website, which had the same game result. I checked the Wikipedia data against the NFL data at certain points, though not for every game, and every time I checked, the data was the same. For these reasons, I thought that the benefits of using the data from Wikipedia outweighed the potential concerns.



Week	Date	Opponent	Result	Record	Venue	Recap
1	September 9	Washington Redskins	L 6–24	0–1	State Farm Stadium	Recap
2	September 16	at Los Angeles Rams	L 0–34	0–2	Los Angeles Memorial Coliseum	Recap
3	September 23	Chicago Bears	L 14–16	0–3	State Farm Stadium	Recap
4	September 30	Seattle Seahawks	L 17–20	0–4	State Farm Stadium	Recap
5	October 7	at San Francisco 49ers	W 28–18	1–4	Levi's Stadium	Recap
6	October 14	at Minnesota Vikings	L 17–27	1–5	U.S. Bank Stadium	Recap
7	October 18	Denver Broncos	L 10–45	1–6	State Farm Stadium	Recap
8	October 28	San Francisco 49ers	W 18–15	2–6	State Farm Stadium	Recap
9	Bye					
10	November 11	at Kansas City Chiefs	L 14–26	2–7	Arrowhead Stadium	Recap
11	November 18	Oakland Raiders	L 21–23	2–8	State Farm Stadium	Recap
12	November 25	at Los Angeles Chargers	L 10–45	2–9	StubHub Center	Recap
13	December 2	at Green Bay Packers	W 20–17	3–9	Lambeau Field	Recap
14	December 9	Detroit Lions	L 3–17	3–10	State Farm Stadium	Recap
15	December 16	at Atlanta Falcons	L 14–40	3–11	Mercedes-Benz Stadium	Recap
16	December 23	Los Angeles Rams	L 9–31	3–12	State Farm Stadium	Recap
17	December 30	at Seattle Seahawks	L 24–27	3–13	CenturyLink Field	Recap

Figure 2.1 and 2.2: The layout of game data on the NFL website[v] versus Wikipedia[vi].

As all the games played by the team are displayed on each page, I had to avoid double-counting games, which would have happened if I had recorded every game on every page. The fastest way I found to do this was to only record the home games on each page: the games which did not say "at [other team]" in each line. In this way, I was able to efficiently ensure that my data had the correct number of games. For each game, I recorded the home team, the away team, the winner and the point difference. I did not end up using the point difference in my models, but I included it as I was unsure at this point what the model would entail.

Season	Home	Away	Winner	Difference
2018	NOS	AF	NOS	14
2018	NOS	CBr	NOS	3
2018	NOS	CP	CP	19
2018	NOS	LAR	NOS	10
2018	NOS	PE	NOS	41
2018	NOS	PS	NOS	3
2018	NOS	TBB	TBB	8
2018	NOS	WC	NOS	24
2019	NOS	AC	NOS	22
2019	NOS	AF	AF	17
2019	NOS	CP	NOS	3
2019	NOS	DC	NOS	2
2019	NOS	HT	NOS	2
2019	NOS	IC	NOS	27
2019	NOS	SF	SF	2
2019	NOS	TBB	NOS	7
2020	NOS	AF	NOS	15
2020	NOS	CP	NOS	3
2020	NOS	GBP	GBP	7
2020	NOS	KCC	KCC	3
2020	NOS	LAC	NOS	3
2020	NOS	MV	NOS	19
2020	NOS	SF	NOS	14
2020	NOS	TBB	NOS	11

Figure 2.3: A sample section of the first data table for the New Orleans Saints.

In order to have more data for modelling purposes, I also found a more general spreadsheet online which had data for each team's seasons since the start of the 1999 season. This data included many fields which I did not use, but the ones I focused on were wins, losses and ties over each season.

	A	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD
1	season	defense_ave_wpa_pass	defense_ave_wpa_run	defense_success_rate_pass	defense_success_rate_run	points_scored	points_allowed	wins	losses	ties	score_differential
2	1999	-0.00412522005426879	-0.000649285907200897	0.42175572519084	0.414258188824663	245	382	6	10	0	-137
3	1999	-0.000676207679739205	-0.00170389537900682	0.402805611222445	0.38125	285	380	5	11	0	-95
4	1999	-0.00445562698181718	-0.00540084948128471	0.357627118644068	0.31830985915493	324	277	8	8	0	47
5	1999	-0.00466031267743428	-0.00545313002464531	0.352504638218924	0.320610687022901	320	229	11	5	0	91
6	1999	-0.000144964521961233	0.000148555870056836	0.452380952380952	0.416279069767442	421	381	8	8	0	40
7	1999	-0.000405411050332613	-0.00315005050878249	0.44281045751634	0.372596153846154	272	341	6	10	0	-69
8	1999	0.000529244862178862	-0.00388060169057029	0.464737793851718	0.366515837104072	283	460	4	12	0	-177
9	1999	0.00221511813991576	-0.000641363461757095	0.490842490842491	0.448630136986301	217	437	2	14	0	-220
10	1999	-0.0036305131127525	-0.00064789352977007	0.391608391608392	0.364089775561097	352	276	8	8	0	76
11	1999	-0.00471238267333638	-0.00628950049135313	0.422396856581532	0.358490566037736	314	318	6	10	0	-4
12	1999	-0.000666076819735538	-0.00403150818609676	0.435483870967742	0.35978835978836	322	323	8	8	0	-1
13	1999	-0.00094713484403281	-0.00299121477794903	0.419182948490231	0.36304347826087	357	341	8	8	0	16
14	1999	-0.00121535861186495	-0.00403274314497547	0.439799331103679	0.403022670025189	423	333	13	3	0	90
15	1999	-0.00462018114439366	-0.00632189436640326	0.371929824561403	0.326975476839237	396	217	14	2	0	179
16	1999	-0.00459018768393085	-0.00505043565074903	0.404255319148936	0.350746268656716	390	322	9	7	0	68
17	1999	-0.00208058957999145	-0.00424056910562977	0.386023294509152	0.299684542586751	526	242	13	3	0	284
18	1999	0.000228976701792467	-0.00792726409467132	0.431034482758621	0.31042654028436	269	316	8	8	0	-47
19	1999	-0.00195719453042459	-0.00232094776764917	0.403141361256545	0.352331606217617	390	329	8	8	0	61
20	1999	-0.0043627541608218	-0.00492829295676406	0.35672514619883	0.313924050632911	326	336	9	7	0	-10
21	1999	0.00153509324005884	-0.00473754346411329	0.448598130841121	0.358722358722359	399	335	10	6	0	64
22	1999	-0.00351762940559752	-0.004394028316419	0.374551971326165	0.343220338983051	299	284	8	8	0	15
23	1999	0.000139562638765037	-0.00167620750550476	0.429643527204503	0.435406698564593	260	434	3	13	0	-174
24	1999	-0.00207370022961635	-0.00743280649969397	0.415270018621974	0.350917431192661	299	358	7	9	0	-59
25	1999	-0.00399305307467044	-0.00108935140697962	0.425675675675676	0.356626506024096	308	309	8	8	0	-1
26	1999	-0.00570945121494981	-0.00284366129267612	0.415973377703827	0.376016260162602	272	357	5	11	0	-85

Figure 2.4: A section of the second data table which I used.[vii]

I then rearranged the data sheet and supplanted it with the records of each team for the 2023 and 2024 seasons, as this sheet only had data up to the 2022 season. Using this data, I would form correlational plots of each team's wins and losses, in order to estimate their wins and losses in the next season.

<u>Data analysis: Regular season</u>

The first piece of analysis which I did was to form regression models of each team's data as a function of games won over time. Immediately on looking at the data of "year versus wins", it was clear that linear regression curves would not be appropriate for the data, as many teams' data follow a curved trend.

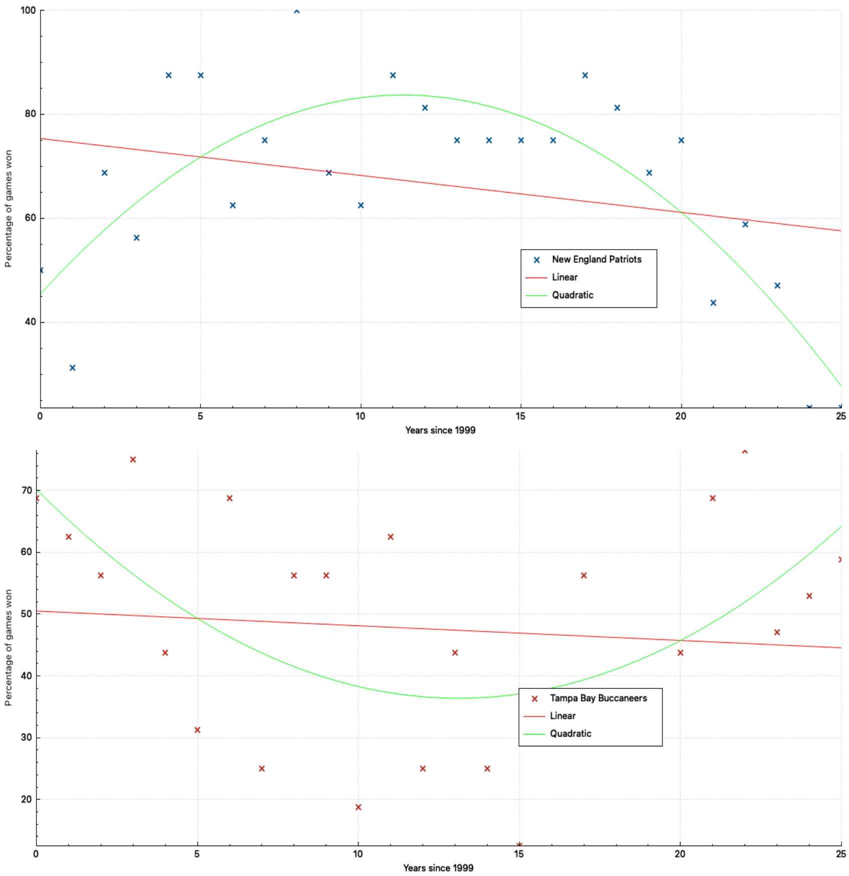


Figure 3.1 and 3.2: Two of the teams for which a linear correlation would be least appropriate.

As well as this, the NFL season changed from 16 to 17 games in 2021, so the last four years of data have, on average, higher values than those before on account of playing one more game per season. As a result, many teams' regression curves resulted in higher values for the 2025 season than would be accurate when I plotted the number of games won each year. In order to fix this issue, I instead graphed the percentage of games won each year. I then multiplied the predicted percentage of games won in the 2025 season by 17 to get back to a numerical result of how many games the team would win. I also set the x-axis as "years since 1999" so that the data points would start at x=0, instead of x=1999, which will make the graph calculations easier later on.

In order to find the relevant regression curves, I used a Levenberg-Marquardt algorithm (LMA), otherwise known as the damped least-squares (DLS) method. This model provides a solution for non-linear least-squares regression, which is exactly what I need. The least-squares approach to regression is an estimation of a line or curve which aims to find the lowest sum of the squares of the difference between each data value and the approximation curve, as displayed below.

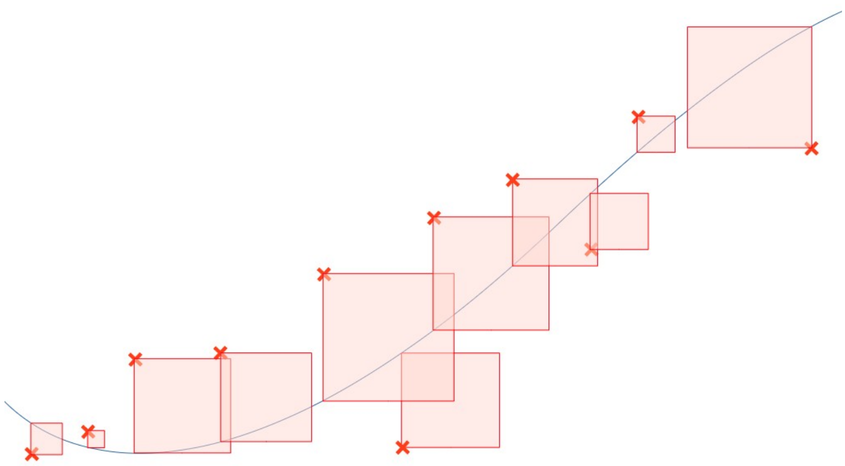


Figure 3.3: A visual representation of a least-squares model for a curve of degree 3.

Least-squares methods find this approximation by estimating solutions to the equation

$\hat{\beta} = argmin_{\beta} S(\beta) = argmin_{\beta} \sum_{i=1}^{m} [y_i - f(x_i;\beta)]^2$ [viii]. $\hat{\beta}$ represents the estimated values of parameters β, which define the function $f(x;\beta) = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n \cdot x^n$, which result in the smallest (*argmin*) sum of squares between *m* number of data points (x_i, y_i) and the curve *f(x)*. Iterative least-squares algorithms, of which the LMA is one, aim to find these β values by calculating a change to these variables, $\Delta\beta$, to minimize the sum of squares. After each calculation, the new parameters are calculated as $\beta_{new} \leftarrow \beta_{old} + \Delta\beta$ and the function and sum of squares are calculated again, providing different numbers through which a different set of $\Delta\beta$ values are found. The process continues until the sum of squares is reduced by an amount that crosses a certain threshold, which is typically user-specified. For these calculations, I will set this threshold as $\Delta S(\beta) \leq -1$. The LMA is a combination of the Gauss-Newton least-squares algorithm (GNA) and the gradient-descent algorithm (GDA). The Gauss-Newton algorithm is itself based on another algorithm, the original Newton algorithm, which found the parametric change using the equation $J(x) + H(x)\Delta\beta = 0$ [ix]. Here *J(x)* is the Jacobian matrix: a matrix which describes all partial derivatives of the given function, $J(x) = \left[\frac{\partial f(x_i)}{\partial \beta_i}\right]$, and *H(x)* is the Hessian matrix, which is a matrix of each second partial derivative of the same function, $H(x) = \left[\frac{\partial^2 f(x)}{\partial x_i \partial x_j}\right]$. As calculating every second derivative is quite slow, the Gauss-Newton approach replaces the Hessian matrix with an approximation. This approximation results in the "normal equation", $J(x) + J^T(x) J(x) \Delta\beta = 0$. $J^T(x)$ here denotes the transpose of the Jacobian matrix, where the columns and rows of a matrix are switched: for instance, if $M(x) = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}$, $M^T(x) = \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix}$. By approximating the Hessian matrix, the Gauss-Newton algorithm avoids calculating second derivatives, making it easier and faster to carry out. The GDA, by comparison, finds minima by finding a point on a curve, calculating the derivative of the equation at that point to find the gradient, subtracting the gradient from the *x*-value of the point, then finding the gradient at the new point, repeating until the gradient reaches 0[x]. Gradient-descent methods are more useful when the initial parameter guess is close to the correct values, as it takes more iterations with further guesses. In this case, the function which is being minimized is the sum of squares, and this method would use the change in this sum to calculate the "gradient" of the parameter change.

In combining these two functions, the Levenberg-Marquardt algorithm uses the formula $(J^T J+\lambda\, diag(J^T J))\Delta\beta=J^T \boldsymbol{r}$ [xi]. Here, the formula is multiplying the parametric change, $\Delta\beta$, by the symmetric Jacobian matrix $J^T J$ as well as by a dampening variable λ multiplied by the diagonal values of the symmetric Jacobian matrix. This variable typically starts at a moderate number like 1, and is either multiplied by 10 if the model gets less accurate (i.e. if $S(\beta)$ increases between iterations) or divided by 10 if the model gets more accurate[xii]. When λ is large, the algorithm is closer to a gradient-descent method in updating the parameters of β, and when λ is small, the algorithm is more like a Gauss-Newton update. This equates to the transpose of the Jacobian matrix, multiplied by the column vector of the residuals **r** between the given *y* value and the *y* value supplied by the equation, found as $\boldsymbol{r_i}(\beta)=y_i-f(x_i;\beta)$. As this is an iterative method, it follows the same broad steps as the Gauss-Newton and gradient-descent approaches between iterations. The primary difference between the Levenberg-Marquardt and Gauss-Newton algorithms is the presence of the dampening variable λ in the Levenberg-Marquardt method. By regulating between the Gauss-Newton and gradient-descent methods, the dampening variable increases the reliability of the LMA as it is suitable for parametric guesses which are both close and far from the true values.

Using the data of the New Orleans Saints as an example, I can use the Levenberg-Marquardt algorithm to optimize the equation $f(x;\beta)$ to create a regression curve for the data, taking the number of years since 1999 as the x-variable, and the percentage of games won in that year as the y-variable. The first step is to define the parameters and the chosen equation: setting $\beta=(a,b,c)$ gives us the required parameters for a regression curve in x^2, defined as the parametric equation $f(x;a,b,c)=ax^2+bx+c$. For this first calculation, I estimated the regression curve as $f(x)=-0.1(x-15)^2+60$ or $f(x)=-0.1x^2+3x+37.5$ using an estimated turning point and completing the square.

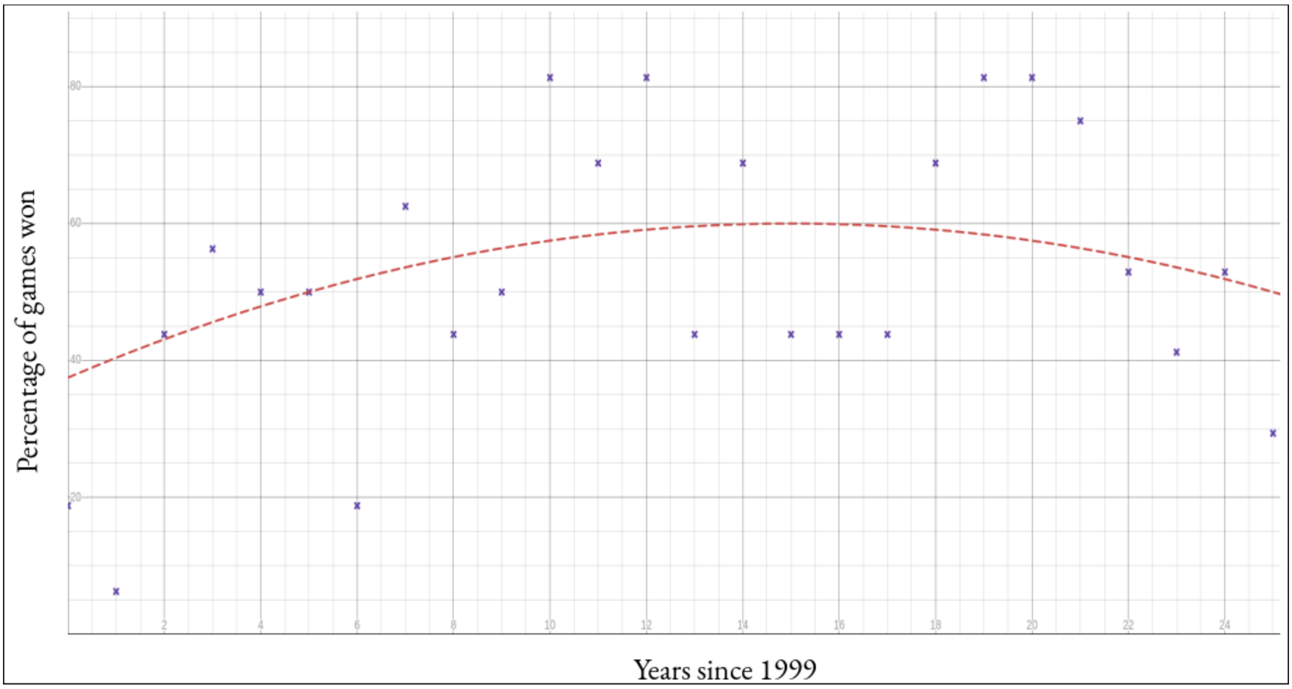


Figure 3.4: A plot of the Saints' season results along with my initial function guess, displayed in red.

This resulted in the initial parameters $a=-0.1$, $b=3$ and $c=37.5$. The next step in this equation is to calculate the residuals **r** for each value of *x* from 0 to 25, and finding the sum of their squares, $S=\sum_{i=0}^{25}(r_i^2)$. From this equation, the residuals and their sums can be calculated as shown in the below table:

x	0	1	2	3	4	5	6	7	8	9	10	11	12
y	18.75	62.50	43.75	56.25	50.00	50.00	18.75	62.50	43.75	50.00	81.25	68.75	81.25
f(x)	37.50	40.40	43.10	45.60	47.90	50.00	51.90	53.60	55.10	56.40	57.50	58.40	59.10
r	-18.75	22.10	0.65	10.65	2.10	0.00	-33.15	8.90	-11.35	-6.40	23.75	10.35	22.15
r^2	351.56	488.41	0.42	113.42	4.41	0.00	1098.92	79.21	128.82	40.96	564.06	107.12	490.62
x	13	14	15	16	17	18	19	20	21	22	23	24	25
y	43.75	68.75	43.75	43.75	43.75	68.75	81.25	81.25	75.00	52.94	41.18	52.94	29.41
f(x)	59.60	59.90	60.00	59.90	59.60	59.10	58.40	57.50	56.40	55.10	53.60	51.90	50.00
r	-15.85	8.85	-16.25	-16.15	-15.85	9.65	22.85	23.75	18.60	-2.16	-12.42	1.04	-20.59
r^2	251.22	78.32	264.06	260.82	251.22	93.12	522.12	564.06	345.96	4.67	154.26	1.08	423.95

Figure 3.5: A table of each y-value, f(x)-value and residual value for each x-value from 0 to 25.

From the r^2 data in this table, the sum of the squares can be found as $S(\beta)_1 = 6682.8$ .While this is not needed to calculate the next iteration of the algorithm, finding the S value allows the success of the algorithm to be assessed, and this means that the λ variable can be adjusted accordingly in following iterations. We can also then form the column vector of residuals with respect to the given parameters, equated as $\boldsymbol{r}(\beta) = \begin{bmatrix} -18.75 \\ 22.1 \\ \vdots \\ -20.59 \end{bmatrix}$, which will be used later in the equation. To find the Jacobian matrix, $f(x_i; a, b, c)$ must be partially derived for each value of *x* with respect to each of *a*, *b* and *c*, to form a 26x3 matrix, $J = \begin{bmatrix} \frac{\partial f(x_1)}{\partial a} & \frac{\partial f(x_1)}{\partial b} & \frac{\partial f(x_1)}{\partial c} \\ \vdots & \vdots & \vdots \\ \frac{\partial f(x_{26})}{\partial a} & \frac{\partial f(x_{26})}{\partial b} & \frac{\partial f(x_{26})}{\partial c} \end{bmatrix}$, where the derivative functions can be simplified to create the matrix $J = \begin{bmatrix} x_1^2 & x_1 & 1 \\ \vdots & \vdots & \vdots \\ x_{26}^2 & x_{26} & 1 \end{bmatrix}$ and the transpose $J^T = \begin{bmatrix} x_1^2 & \cdots & x_{26}^2 \\ x_1 & \cdots & x_{26} \\ 1 & \cdots & 1 \end{bmatrix}$. Using matrix multiplication, a 3x3 matrix can be formed as $J^T J = \begin{bmatrix} \sum_{x=0}^{25} x^4 & \sum_{x=0}^{25} x^3 & \sum_{x=0}^{25} x^2 \\ \sum_{x=0}^{25} x^3 & \sum_{x=0}^{25} x^2 & \sum_{x=0}^{25} x \\ \sum_{x=0}^{25} x^2 & \sum_{x=0}^{25} x & \sum_{x=0}^{25} x^0 \end{bmatrix}$. Using the formulae for each of these sums, $\sum_{x=0}^{n} x = \frac{1}{2}n^2 + \frac{1}{2}n$, $\sum_{x=0}^{n} x^2 = \frac{1}{3}n^3 + \frac{1}{2}n^2 + \frac{1}{6}n$, $\sum_{x=0}^{n} x^3 = \frac{1}{4}n^4 + \frac{1}{2}n^3 + \frac{1}{4}n^2$ and $\sum_{x=0}^{n} x^4 = \frac{1}{5}n^5 + \frac{1}{2}n^4 + \frac{1}{3}n^3 - \frac{1}{30}n$ [xiii] nets the matrix $J^T J = \begin{bmatrix} 2153645 & 105625 & 5525 \\ 105625 & 5525 & 325 \\ 5525 & 325 & 26 \end{bmatrix}$ .The diagonal matrix can also be found as $diag(J^T J) = \begin{bmatrix} 2153645 & 0 & 0 \\ 0 & 5525 & 0 \\ 0 & 0 & 26 \end{bmatrix}$. Using these matrices, and using an initial dampening variable value of $\lambda=1.0$, the first side of the iterative equation can be equated to

$(J^T J+diag(J^T J))\Delta\beta=\begin{bmatrix}4307290 & 105625 & 5525\\ 105625 & 11050 & 325\\ 5525 & 325 & 52\end{bmatrix}\begin{bmatrix}\Delta a\\ \Delta b\\ \Delta c\end{bmatrix}$. This then simplifies to a column vector

in three variables, $\begin{bmatrix}4307290\Delta a+105625\Delta b+5525\Delta c\\ 105625\Delta a+11050\Delta b+325\Delta c\\ 5525\Delta a+325\Delta b+52\Delta c\end{bmatrix}$. To find the other half of the equation, we

multiply $\boldsymbol{r}(\beta)$ by J^T to get the column vector $J^T \boldsymbol{r}(\beta)=\begin{bmatrix}846.37\\ 192.63\\ 16.47\end{bmatrix}$. Because two 3x1 column vectors

have been formed, a set of simultaneous equations can be formed:

$4307290\Delta a+105625\Delta b+5525\Delta c=846.37$, $105625\Delta a+11050\Delta b+325\Delta c=192.63$ and

$5525\Delta a+325\Delta b+52\Delta c=16.47$. Solving this system of equations results in the $\Delta\beta$ values of

$\Delta a=-5.097\times10^{-4}$, $\Delta b=0.013963$ and $\Delta c=0.28362$ to four significant figures. We can then add these values back to each of *a*, *b* and *c* to find the new equation $f(x)-0.10051x^2+3.0139b+37.784$, which results in a sum of squares value of $S(\beta)_2=6672.2$. By comparing this to the first value, we can see that the first iteration was successful, so we can set $\lambda=0.1$ for the next iteration. After the fifth iteration, decreasing λ by a factor of 10 after each successful iteration, the sum of squares S decreased by less than 1 as the algorithm converged to the function $f(x)=-0.13543x^2+3.87678x+34.703$.

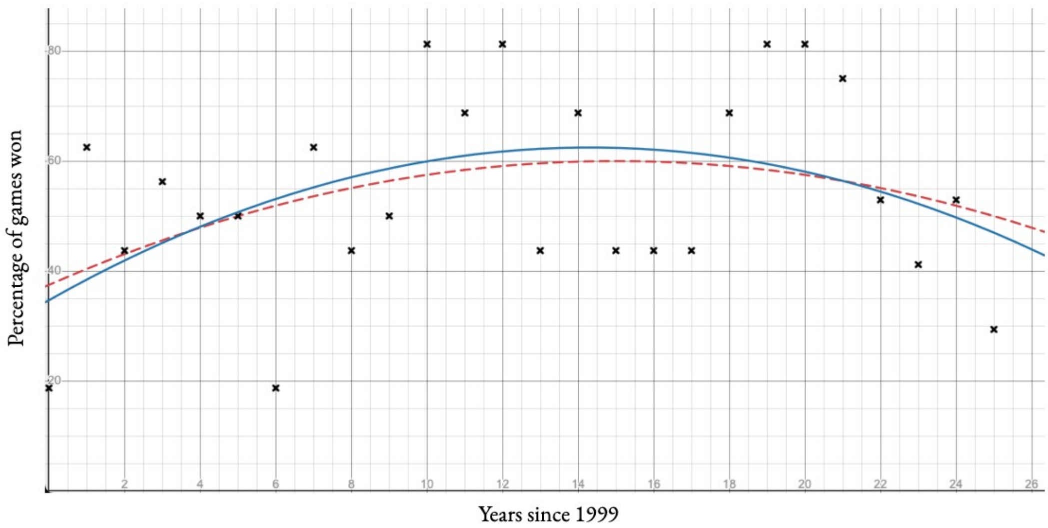


Figure 3.6: A comparison between my initial guess (in red) and the final equation (in blue) against the data points of the Saints.

By using this function and setting $x=26$, the theoretical percentage of games won in the 2025 season by the Saints can be estimated as 43.95%, or 7.47 games.

One limitation of the LMA is that it can only calculate local minima, rather than the global minimum, for any guessed parameter values. If there are multiple minima, the algorithm can only approximate these points if the initial guess is close to the final solution. This is due to the characteristic of the gradient-descent update algorithm: once a minimum point is found, the algorithm stops, therefore to find the global minimum, initial guesses must be provided which are closer to the global minimum than a local minimum, as opposed to a stock guess. Because my algorithm model is quite simple, only involving 3-4 parameters, it is unlikely that this will present a considerable issue.

For the rest of the algorithm calculations, I used a graphing software instead of manually calculating the correlations. For these, instead of manually guessing each team's parameters to start, I used stock guesses of 1 for each value, as this was the default in the graphing program. After collating all of the data and calculating correlation curves, I formed a table of results, arranged by division and conference. One point to note is that the Houston Texans were established in 2002, therefore their model is using 23, rather than

26, data points. Although I could have discarded all teams' data values prior to 2002, I thought it was a better trade-off to have slightly less consistent data for the Texans, rather than sacrificing a significant portion of all teams' data. For each team, I found the correlation curve in both x^2 and x^3 and used the data of the curve which fit the data better. My intention in doing this is to increase the reliability of the model which is being used to find the trend in the teams' performance. By using multiple correlation curves for each team, I can assess correlation coefficients for each and use the curve which is more appropriate for each team.

To calculate the fit, I used Kendall's rank correlation coefficient (KRCC) for non-linear correlation models. The KRCC is found by taking a set of *n* objects, $S=\{a,b,c...,x,y\}$ which, for an example sample set of 4 objects $S=\{a,b,c,d\}$ is then transformed into any ordered set *O*, for instance $O_X=[a,d,b,c]$. This ordered set can then be arranged into a set of ordered pairs, $P_X=\{[a,d],[a,b],[a,c],[d,b],[d,c],[b,c]\}$ and then compared to a different set of ordered pairs, for instance $P_Y=\{[a,d],[a,c],[a,b],[d,c],[d,b],[c,b]\}$ based on a different ordered set $O_Y=[a,d,c,b]$. By taking the symmetric difference distance d_Δ as the total number of discordant pairs: the number of pairs which are in one set but not the other, which would take a value of 2 for the above example due to the two different pairs [c,b] and [b,c], an absolute numerical value of the difference between the sets can be found. When calculating the symmetric difference distance, the order of the pairs within the set does not matter, only the order of the values within each pair. Following this, the symmetric difference d_Δ is normalized as τ_a to take values between -1 and +1, where -1 is the largest possible difference, found in a perfect negative correlation and +1 is the smallest possible difference, found in a perfect positive correlation. This results in the formula $\tau_a=\frac{N-d_\Delta(P_x,P_y)}{N}$ [xiv] where N

represents the number of ordered-pair elements in each set P_i, calculated as $\frac{n(n-1)}{2}$. This formula can then be simplified to $\tau_a = 1 - \frac{d_\Delta}{N}$. Due to the nature of the data being used here, however, there is a guarantee that some values of games won will be tied, as there are only 16 possible *y*-values that can be taken for each of the 22 *x*-values from *x*=0 to *x*=21, so the tie-corrected coefficient, τ_b, must be calculated. To find this value, *N* is first equated as $N = C + D + T_X + T_Y$, where *C* is equal to the number of concordant (matching) pairs in each set P_i, *D* is equal to the number of discordant pairs in each set, or $\frac{d_\Delta}{2}$, T_X represents the number of tied pairs – pairs of two tied values – in list P_X, and T_Y represents the number of tied pairs in list P_Y. The correlation coefficient is then re-calculated as

$\tau_b = \frac{C - D}{\sqrt{(C + D + T_X)(C + D + T_Y)}}$ [xv], or $\tau_b = \frac{C - D}{\sqrt{(N - T_X)(N - T_Y)}}$. By removing the tied pairs in the denominator, this second formula effectively readjusts the Kendall correlation coefficient for the number of non-tied pairs without having to change the dataset, making it more appropriate for this case.

As an example, taking the Saints' data and setting $a \leftarrow x = 0, b \leftarrow x = 1 \ldots z \leftarrow x = 25$ to create the sample set $S = \{a, b, c, \ldots, z\}$, the data points of both the actual data and the values given by the regression function can be arranged in ascending order to create the ordered sets

$P_A = \{a, g, z, x, c, i, n, p, q, r, e, f, j, w, y, d, b, h, l, o, s, v, k, m, t, u\}$ for the actual values and

$P_T = \{a, b, c, d, z, e, y, f, x, g, w, h, v, i, u, j, t, k, s, l, r, m, q, n, p, o\}$ for the theoretical values.

These two sets then created the ordered pair sets $O_A = \{(a, g), (a, z), \ldots, (m, u), (t, u)\}$ and $O_T = \{(a, b), (a, c), \ldots, (n, o), (p, o)\}$. Using the formula for τ_b and plugging in the appropriate values for O_A and O_T, the correlation coefficient can be calculated as $\tau_b = \frac{179 - 116}{\sqrt{(325 - 30) \cdot (325 - 0)}}$, with a final result of $\tau_b = 0.2035$. As this is not a particularly strong correlation, it is likely that the cubic correlation curve will be better for this team. This weak trend is likely to be due to the fact that the Saints' data seems to fluctuate quite a lot from year to year, even if the midpoint of these fluctuations does appear to follow a correlation curve.

Calculating the KRCC for each correlation curve of each team, I formed this table of values, highlighting for each team which curve was a better fit.

NFC (cubic)							
North		South		East		West	
Team	τ	Team	τ	Team	τ	Team	τ
Lions	0.306	Buccaneers	0.393	Eagles	0.358	Rams	0.533
Vikings	0.152	Falcons	0.332	Commanders	0.188	Seahawks	0.135
Packers	0.032	Panthers	0.330	Cowboys	0.212	Cardinals	0.350
Bears	0.307	Saints	0.236	Giants	0.430	49ers	0.139
AFC (cubic)							
North		South		East		West	
Team	τ	Team	τ	Team	τ	Team	τ
Ravens	0.064	Texans*	0.297	Bills	0.522	Chiefs	0.498
Steelers	0.147	Colts	0.449	Dolphins	0.351	Chargers	0.354
Bengals	0.268	Jaguars	0.316	Jets	0.374	Broncos	0.228
Browns	0.325	Titans	0.259	Patriots	0.508	Raiders	0.388

NFC (quadratic)							
North		South		East		West	
Team	τ	Team	τ	Team	τ	Team	τ
Lions	0.306	Buccaneers	0.393	Eagles	0.212	Rams	0.439
Vikings	0.146	Falcons	0.326	Commanders	0.201	Seahawks	0.225
Packers	0.076	Panthers	0.311	Cowboys	0.225	Cardinals	0.350
Bears	0.301	Saints	0.203	Giants	0.430	49ers	0.139
AFC (quadratic)							
North		South		East		West	
Team	τ	Team	τ	Team	τ	Team	τ
Ravens	0.102	Texans*	0.248	Bills	0.529	Chiefs	0.517
Steelers	0.128	Colts	0.373	Dolphins	0.306	Chargers	0.112
Bengals	0.262	Jaguars	0.322	Jets	0.361	Broncos	0.209
Browns	0.019	Titans	0.145	Patriots	0.514	Raiders	0.202

Figure 3.7: A table of τ_b values for cubic (top) and quadratic (bottom) correlation curves.

Using these τ values to determine which curve to use as the final model, I formed another table, showing each team's predicted number of wins with the preferred correlation curve, by calculating each function at x=26. In the case that both correlation constants were the same between curves, I took the average values of the result of the cubic and quadratic regression curve.

NFC							
North		South		East		West	
Team	Games won	Team	Games won	Team	Games won	Team	Games won
Vikings	11	Buccaneers	12	Eagles	15	49ers	10
Lions	11	Falcons	7	Cowboys	9	Seahawks	10
Packers	10	Saints	6	Commanders	9	Rams	9
Bears	6	Panthers	2	Giants	4	Cardinals	5
AFC							
North		South		East		West	
Team	Games won	Team	Games won	Team	Games won	Team	Games won
Ravens	12	Colts	9	Bills	15	Chiefs	16
Steelers	10	Texans	8	Dolphins	9	Chargers	11
Bengals	10	Jaguars	6	Jets	5	Broncos	7
Browns	10	Titans	5	Patriots	3	Raiders	5

Figure 3.8: The modelled table of how many games each team is projected to win.

Due to the Saints not making it to the playoffs in this initial, the next step will be to calculate the chance that they make it through to the playoffs. There are two ways for this to happen: finishing top of the division, or getting a "wild card" position: finishing in the top 3 of the conference (NFC) when these four division winners are excluded. In order to finish top of the division, the Saints would have to win a total of 13 games. By comparison, to beat one of the three next-best teams in the division, the Saints would only have to beat the Cowboys, with 9 games. In order to calculate the most likely 6 teams which the Saints would have already beaten, and the next most likely 4 for the Saints to beat to make it to the playoffs, I used a second spreadsheet of data, which recorded individual games between teams from 2018 to 2024. Using this in conjunction with the games which have been announced for the 2025 season[xvi], I formed this chart to see the teams which the Saints have had the best track record with.

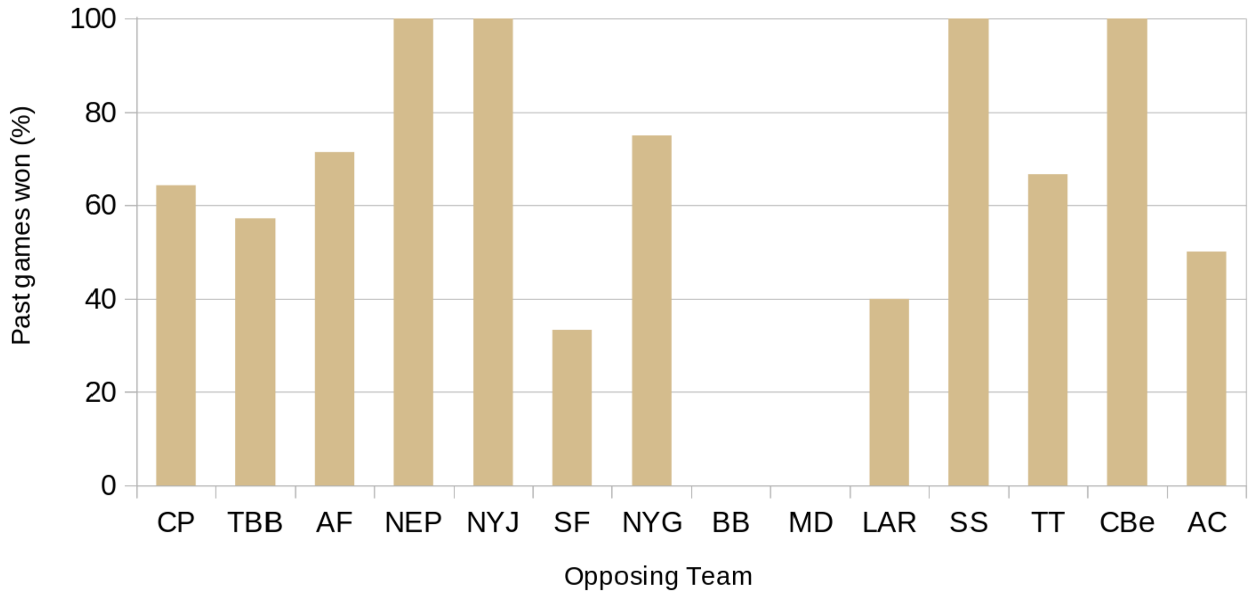


Figure 3.9: A chart of the percentage of games which the Saints have won since 2018 against each of the teams they will play in 2025.

Using this graph, it can be predicted that the 6 games which the Saints are predicted by the previous model to win are most likely to be won against the Falcons, the Patriots, the Jets, the Giants, the Seahawks and the Bears. Taking these six wins as guaranteed, I then calculated the probability of each way for the Saints to get to the playoffs, winning four or more games, using a tree diagram. I did not include the Bills or Dolphins in this diagram due to the Saint's 0% win rate against them in the past 6 years. In the name of simplicity, I also left out the 49ers and Buccaneers. This is because they are top of their division, meaning they already have a spot in the playoffs. The NFL playoff seeds for each division are assigned so that seeds 1 through 4 are the four division leaders in order of the number of games won, and seeds 5 through 7 are the teams with the first, second and third most wins excluding the four division leaders. If the number of games that the division leaders win changes, the seeding for the playoffs would change and this would lead to a high number of possible outcomes, which would not be feasible to visualize and calculate in their entirety. This left six potential games against 5 teams (the Saints play each of the Panthers, Buccaneers and Falcons twice), of which the Saints had to win four, as illustrated below:

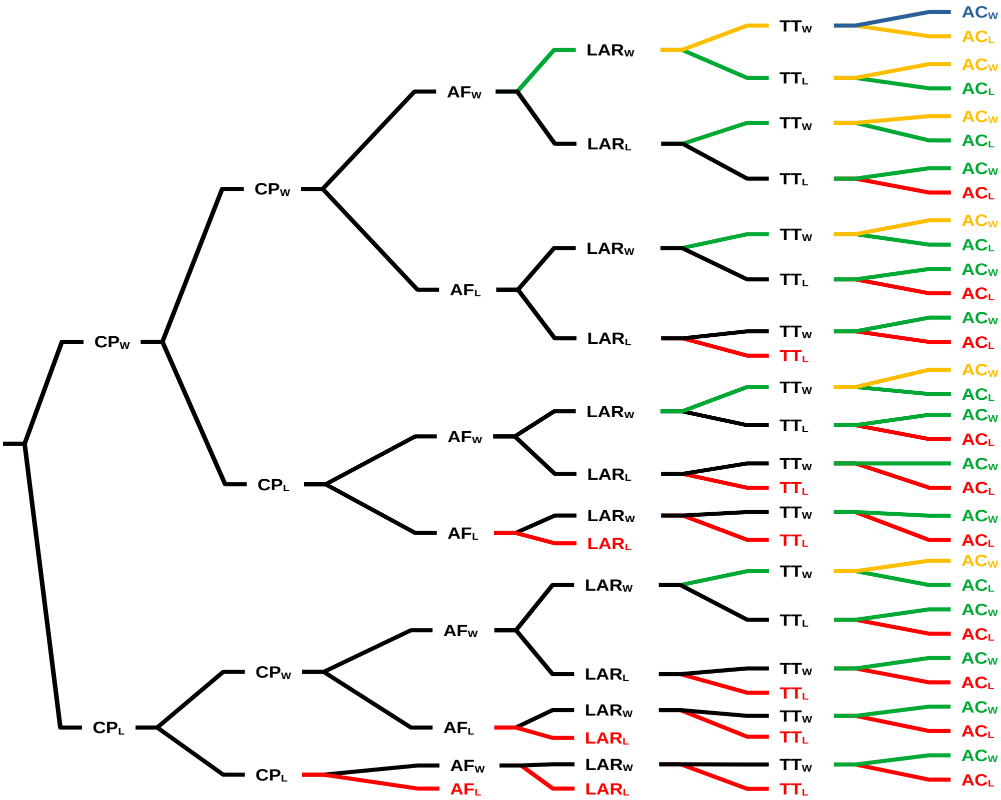


Figure 3.10: A tree diagram of each potential combination of 4 or more wins from the 6 games.

This diagram represents all winning outcomes for the Saints to get to the playoffs. Though not displayed on the diagram due to space constraints, the probabilities for each win are as follows, with the probability of the loss being $X_L = 1 - X_W$:

$CP_W = 0.643, AF_W = 0.714, LAR_W = 0.400, TT_W = 0.667, AC_W = 0.500$. Outcomes of four or more wins are displayed in green, outcomes of five or more wins in yellow and outcomes of six wins in blue. Any combination of three losses is displayed in red, as the Saints can only lose a maximum of two games in this scenario. This creates three distinct seeding combinations: one where the Saints have 10 wins, reaching #7 seed, one where they have 11 wins, reaching #6 seed, and one where they have 12 wins, reaching #5 seed. The probability of each of these outcomes will be denoted as $P(S7)$, $P(S6)$ and $P(S5)$. Assuming that the probabilities of each game are independent of each other, the three outcomes can be found by multiplying along each path of the tree diagram adding the results together for each outcome, for instance, calculating the probability of the top path gives us the equation $P(S_5)=CP_W \cdot CP_W \cdot AF_W \cdot LAR_W \cdot TT_W \cdot AC_W$ to get the result of 0.0394, or 3.94%. Carrying out the same calculations for the other two outcomes gives the results $P(S_6)=0.1778$ or 17.78% and $P(S_7)=0.2715$ or 27.15%. This results in the total chance of the Saints making it to the playoffs as $P(S_P)=0.4887$ or 48.87%. Although I did not calculate this directly in the name of saving time, the probability of the Saints not making the playoffs is calculated as $P(S_L)=1-P(S_P)$, with the result of $P(S_L)=0.5113$, or 51.13%. Given that only 7 of the 16 teams in each conference make it to the playoffs, this puts the Saints slightly above the predicted average for each team of $\frac{7}{16}$ or 43.75%. Using these outcomes, three tables can be created to show the seeds in each case:

P(S5): 3.94%	NFC	AFC
Seed 1	Eagles	Chiefs
Seed 2	Buccaneers	Bills
Seed 3	Vikings	Ravens
Seed 4	49ers	Colts
Seed 5	Saints	Chargers
Seed 6	Lions	Steelers
Seed 7	Seahawks	Dolphins

P(S6): 17.78%	NFC	AFC
Seed 1	Eagles	Chiefs
Seed 2	Buccaneers	Bills
Seed 3	Vikings	Ravens
Seed 4	49ers	Colts
Seed 5	Lions	Chargers
Seed 6	Saints	Steelers
Seed 7	Seahawks	Dolphins

P(S7): 27.15%	NFC	AFC
Seed 1	Eagles	Chiefs
Seed 2	Buccaneers	Bills
Seed 3	Vikings	Ravens
Seed 4	49ers	Colts
Seed 5	Lions	Chargers
Seed 6	Seahawks	Steelers
Seed 7	Saints	Dolphins

Figure 3.11: The three tables of seeds for the playoffs, along with the relative probability that each happens.

Data analysis: playoffs

In the first round of the NFL playoffs, the "Wild Card playoffs", the #2 seed of each conference plays the #7 seed of the same conference, #3 plays #6, and #4 plays #5. The #1 seed skips this first round, and plays the lowest remaining seed in the second round, the "Divisional playoffs". The remaining two teams then face each other as the second game in this round. The third round, the NFC/AFC Championship, sees the final two remaining teams from each conference play each other, and the winner of each of these games goes on to play in the final game, the Super Bowl. Using these steps, I formed a bracket for the first round of playoffs, which I will refer back to after modelling each round.

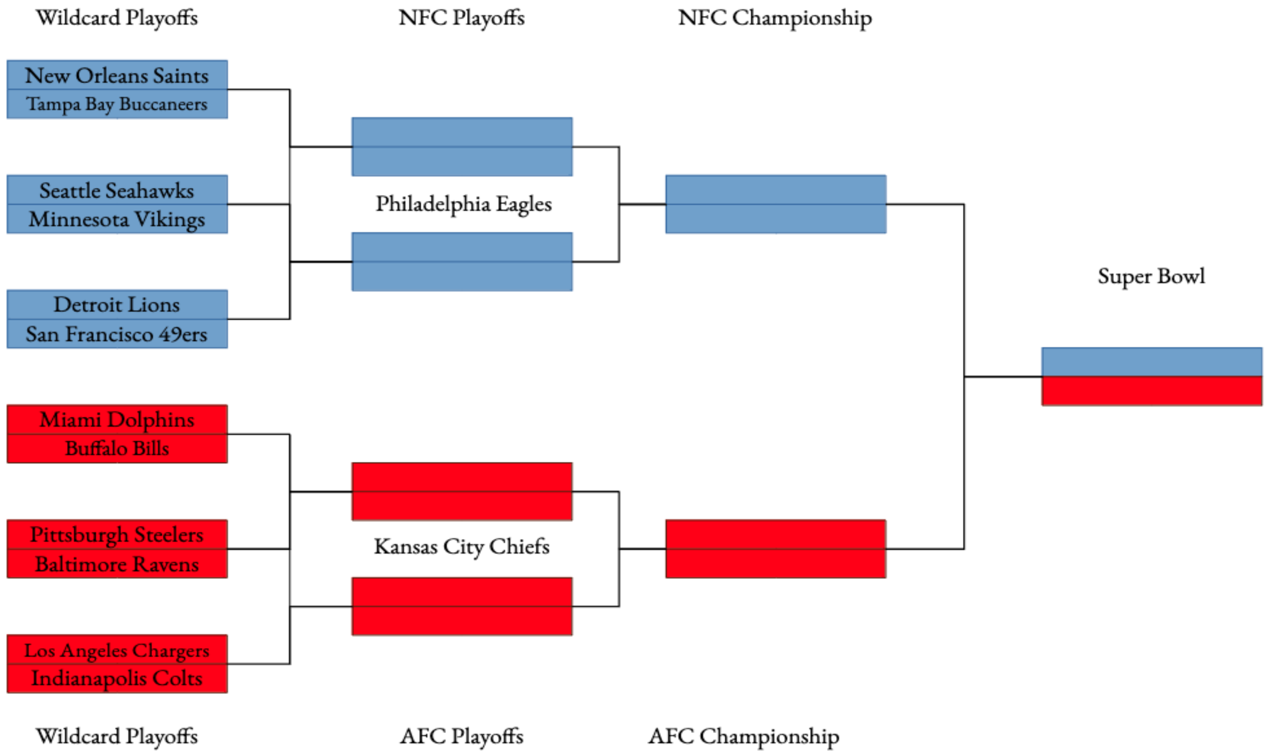


Figure 4.1: A tournament bracket for the first round of the playoffs.

The next thing to do was to construct a set of tree diagrams of all playoff game results up to the conference championship, to see the chances of each team making it to the Super Bowl. There are three tree diagrams for the NFC, one for each of the Saints' seed positions, and one for the AFC, as the seed positions are the same in each case in the AFC. Any game result where the Saints lose is shown in red, and any result where the Chiefs, Dolphins or Bills win their championship are also shown in red, as the Saints have not won against them since 2018, and as a result my model would make it impossible for these teams to lose the Super Bowl against the Saints. Shown in yellow is each game result which would also be impossible using my model as they have won no games against the given opponent since 2018.

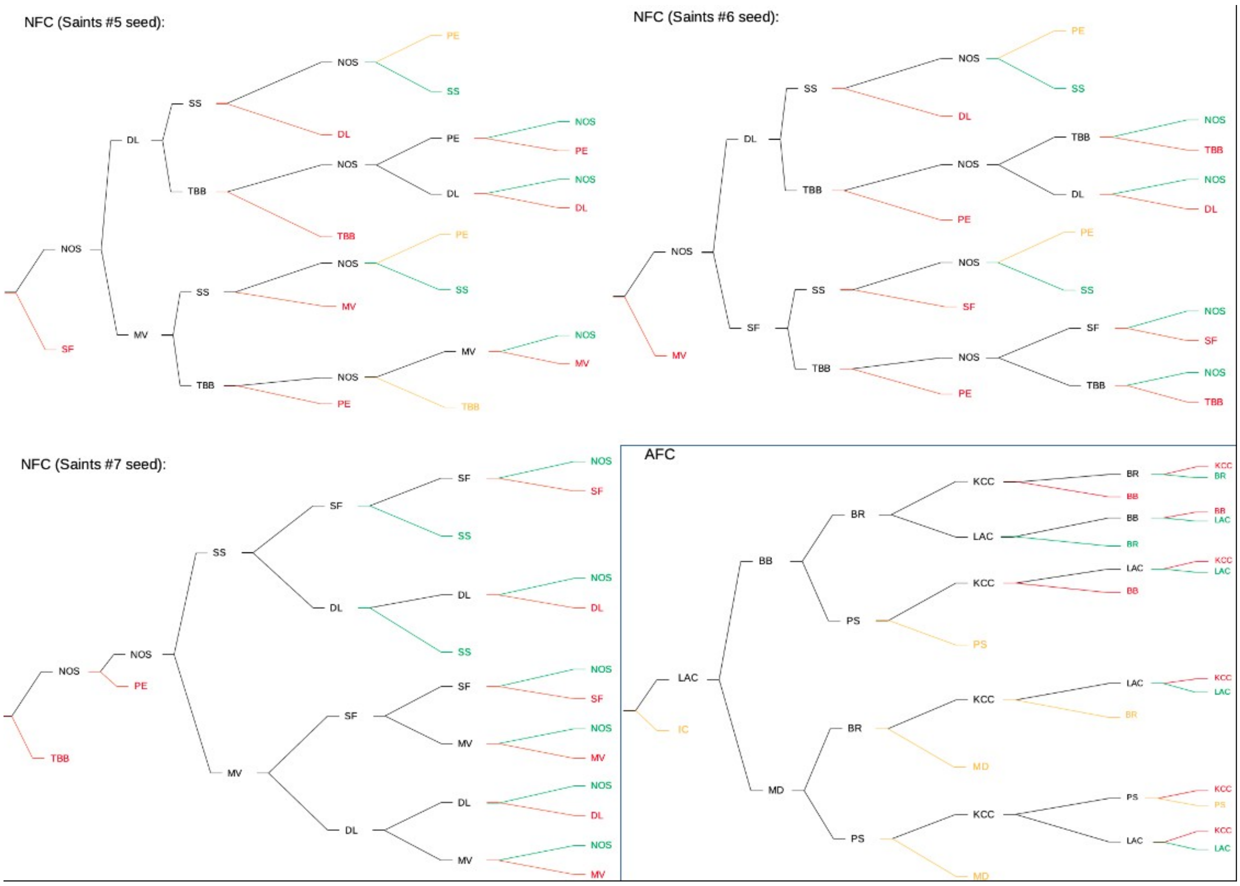


Figure 4.3: Four tree diagrams encompassing all possibilities of the Super Bowl playoffs according to the previous models.

From these diagrams, I formed a table of probabilities of each team getting to the Super Bowl which would make it possible for the Saints to win the Super Bowl: any championship result shown in green. For this, I used the same approach as calculating the Saints' chance of making it to the Super Bowl: assuming each game won is independent, and multiplying along each "branch" of the diagram to find the probability of each possible championship winner. Again, in the sake of saving time, I only calculated the paths which make it possible for the Saints to win. Using this approach gave the following results:

$P(NOS_5)$	$P(NOS_6)$	$P(NOS_7)$	P(LAC)	P(BR)
12.07%	14.09%	20.17%	5.68%	15.34%

Figure 4.4: A table of each probability of getting to the Super Bowl for the Saints, Chargers and Ravens.

Here, $P(NOS_n)$ denotes the probability of the Saints winning the NFC championship, having started as seed *n.* As the seeding doesn't affect the Super Bowl matchups, I multiplied each probability of the Saints getting to the Super Bowl by the chance of them getting to the playoffs, adding those three together to get the overall probability of the Saints getting to the Super Bowl,

$P(NOS)=P(NOS_5 \cap S_5)+P(NOS_6 \cap S_6)+P(NOS_7 \cap S_7)$, as 8.45%.

To calculate the chance of the Saints winning the Super Bowl against each of the two possible opponents, the Chargers and the Ravens, I expanded my dataset of individual games to include every game played between the Saints and each of these teams from 1996 to 2024[xvii]. I didn't do this for the initial data collection as it would have taken far too long, but it was easy to do here because I was only looking at three teams. I took 1996 as the cutoff date because that was the first year which the Ravens played in the NFL, making 1996-2025 the longest period of time which both teams played. This gave the overall percentage of wins against the Chargers $P(NOS \succ LAC)=62.5\%$, and the percentage of wins against the Ravens $P(NOS \succ BR)=25\%$. Using these, the final probability of the Saints winning the Super Bowl can be found as $P(SB_W)=P(NOS)\cdot(P(NOS \succ LAC)+P(NOS \succ BR))$, presented numerically as

$P(SB_W)=0.0845\cdot(0.625+0.25)=0.739$ or 7.39%.

<u>Conclusion</u>

When assessing the reliability and/or accuracy of my model, the biggest barrier was time. Because I was looking at 32 teams in a variety of ways, every calculation that I performed took a number of hours, as even simple calculations would have to be performed 32 times. The issue of time also presented itself when finding the probability of the Saints making the playoffs: in an ideal world, I would have been able to find all possibilities of the Saints getting to the playoffs, from winning 10 games to 16 games. However, the sheer number of possibilities – 1024 even when the first 6 wins are taken as a given – prevented me from being able to fully calculate this. There were also many variables which I did not investigate in my model, most notably team rosters. These variables would have been very hard to model, due to their qualitative nature providing a restriction on both past modelling and future predictions. For these reasons, I know that this exploration has limited accuracy, as the data and methods were limited in their scope.

However, the methods which I did use appear to be quite well-suited to the exploration. The regression curves which I used were, for the most part, suitable for the data. While the correlation coefficients showed largely low fits, I believe that this is mostly due to the high variability of the number of wins. In retrospect, I would likely have gotten higher correlations when taking the average of every two points (1 and 2, 3 and 4, and so on), as this would have removed a good portion of the fluctuation. There were also indicators along the way which suggested that my model worked well. With both the probability of the Saints making it to the playoffs and the overall probability of them winning the Super Bowl, the Saints were slightly above the baseline probabilities of 1/16 and 1/32, respectively. This seems to suggest that my model is mostly accurate, as none of these values are significantly higher or lower than the baseline.

The primary application for this kind of exploration is, perhaps predictably, in gambling. Personally, I would not and will not apply this math to that field, however it cannot be denied that sports betting is a thriving industry, especially with the NFL: in the 2024 season alone, around $35 billion were spent on various bets[xviii]. Interestingly, my calculated estimate of an 8.45% chance of the Saints winning the Super Bowl is much higher than the current betting odds of 0.6%[xix], placing them as the fourth-least likely to win the Super Bowl. This difference is likely due to the factors which I did not look at, as well as a more extensive data set built up over years, not months.

Overall, I think that this model has worked quite well, as there were no significant flaws which presented themselves at any point which would have proven the investigation wholly incorrect. The result is also somewhat good to know as a Saints supporter, as it suggests that they have a relatively good chance at winning the Super Bowl.

Appendix A: Team matchup probabilities

P(NOS≻CP)	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
0.6875	0.5714	0.2857	1.0000	0.7500	0.3333	0.6667
0.0000	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
0.7500	0.0000	0.0000	0.4000	1.0000	1.0000	0.5000

0.0000	**0.0000**	**P(DL≻PE)**	**P(SF≻PE)**	**P(MV≻PE)**	**P(TBB≻PE)**	**P(PE≻TBB)**
0.4000	1.0000	0.3333	0.6667	0.5000	0.7500	0.2500
0.0000	**0.0000**	**P(DL≻TBB)**	**P(SF≻TBB)**	**P(MV≻TBB)**	**P(TBB≻MV)**	**P(PE≻MV)**
0.5714	0.5000	0.2500	0.8000	1.0000	0.0000	0.5000
0.0000	**0.0000**	**P(DL≻MV)**	**P(SF≻MV)**	**P(MV≻SF)**	**P(TBB≻SF)**	**P(PE≻SF)**
0.5000	0.6000	0.5000	0.2500	0.7500	0.2000	0.3333
0.0000	**0.0000**	**P(DL≻SF)**	**P(SF≻DL)**	**P(MV≻DL)**	**P(TBB≻DL)**	**P(PE≻DL)**
0.3333	0.4286	0.3333	0.6667	0.5000	0.7500	0.6667
0.0000	**0.0000**	**P(DL≻SS)**	**P(SF≻SS)**	**P(MV≻SS)**	**P(TBB≻SS)**	**P(PE≻SS)**
0.5000	0.8000	0.2000	0.5714	0.4000	0.5000	0.0000
0.0000	**0.0000**	**P(DL≻NOS)**	**P(SF≻NOS)**	**P(MV≻NOS)**	**P(TBB≻NOS)**	**P(PE≻NOS)**
1.0000	0.0000	0.5000	0.6667	0.5000	0.4286	0.6000

P(MD≻KCC)	**P(PS≻KCC)**	**P(LAC≻KCC)**	**P(IC≻KCC)**	**P(BR≻KCC)**	**P(BB≻KCC)**	**P(KCC≻BB)**
0.0000	0.0000	0.2143	1.0000	0.2000	0.8000	0.2000
P(MD≻BB)	**P(PS≻BB)**	**P(LAC≻BB)**	**P(IC≻BB)**	**P(BR≻BB)**	**P(BB≻BR)**	**P(KCC≻BR)**
0.1429	0.2500	0.3333	0.6667	0.7500	0.2500	0.8000
P(MD≻BR)	**P(PS≻BR)**	**P(LAC≻BR)**	**P(IC≻BR)**	**P(BR≻IC)**	**P(BB≻IC)**	**P(KCC≻IC)**
0.5000	0.3571	0.0000	0.3333	0.6667	0.3333	0.0000
P(MD≻IC)	**P(PS≻IC)**	**P(LAC≻IC)**	**P(IC≻LAC)**	**P(BR≻LAC)**	**P(BB≻LAC)**	**P(KCC≻LAC)**
0.2500	0.4000	1.0000	0.0000	1.0000	0.6667	0.7857
P(MD≻LAC)	**P(PS≻LAC)**	**P(LAC≻PS)**	**P(IC≻PS)**	**P(BR≻PS)**	**P(BB≻PS)**	**P(KCC≻PS)**
0.5000	0.5000	0.5000	0.6000	0.6429	0.7500	1.0000
P(MD≻PS)	**P(PS≻MD)**	**P(LAC≻MD)**	**P(IC≻MD)**	**P(BR≻MD)**	**P(BB≻MD)**	**P(KCC≻MD)**
0.5000	0.5000	0.5000	0.7500	0.5000	0.8511	1.0000

Appendix B: Regression curve equations

Team	Cubic	Quadratic
Cardinals	$-0.005084x^3+0.5868x^2+1.827x+27.76$	$-0.1320x^2+3.696+24.25$
Falcons	$0.008821x^3-0.4757x^2+6.896x+27.19$	$-0.1449x^2+3.652+33.28$
Bills	$-0.004853x^3+0.3511x^2-4.900x+55.92$	$0.1691x^2-3.116x+52.57$
Ravens	$0.003517x^3-0.09186x^2+0.6377x+51.75$	$-0.07003x^2+1.720x+55.59$
Bears	$0.007140x^3-0.3699x^2+4.611x+37.81$	$-0.1021+1.986+42.73$
Bengals	$0.01451x^3-0.6364x^2+8.284x+18.55$	$-0.09208x^2+2.946+28.57$
Browns	$0.01237x^3-0.4190x^2+3.801x+24.77$	$0.04473x^2-0.7463x+33.31$
Panthers	$-0.005999x^3+0.06982x^2+1.162x+40.18$	$-0.1552x^2+3.368x+36.04$
Broncos	$0.006672x^3-0.3241x^2+3.709x+48.58$	$-0.07388x^2+1.256x+53.18$
Cowboys	$0.004877x^3-0.2455x^2+4.027x+35.51$	$-0.06267x^2+2.234x+38.87$
Lions	$-0.002013x^3+0.1891x^2-2.553+39.25$	$0.1137x^2-1.813x+37.86$
Packers	$-0.002294x^3+0.06406x^2-0.1570x+59.35$	$-0.02197x^2+0.6866x+57.77$
Texans	$0.01714x^3-0.8570x^2+13.04x-11.99$	$-0.1370x^2+4.313x+16.09$
Colts	$0.01192x^3-0.4965x^2+4.434$	$-0.04960x^2+0.05217+69.80$
Jaguars	$0.005594x^3-0.1201^2-1.294x+58.77$	$0.08963x^2-3.351x+62.63$
Chiefs	$-0.01063x^3+0.5431x^2-6.016x+60.15$	$0.1440x^2-2.108x+52.82$
Chargers	$0.02263x^3+0.9559x^2+11.16x+22.18$	$-0.1073x^2+2.838x+37.80$
Rams	$-0.02463x^3+1.179x^2-15.53x+92.11$	$0.2559x^2-6.478x+75.11$
Raiders	$-0.02281x^3+0.9745x^2-11.64x+71.30$	$0.119x^2-3.249x+55.56$
Dolphins	$-0.007036x^3+0.3853x^2-5.775+67.65$	$0.1214x^2-3.188x+62.79$
Vikings	$-0.001570x^3+0.1349x^2-2.040x+86.46$	$0.07603x^2-1.463x+55.38$
Patriots	$-0.004093x^3-0.1455x^2+5.261+48.26$	$-0.01993x^2-0.5636x+54.46$
Saints	$-0.009327x^2+0.2143x^2+0.4475x+41.14$	$-0.1354x^2+3.877x+34.70$
Giants	$0.005479x^3-0.2910x^2+3.045x+47.61$	$-0.08557x^3+1.030x^2+51.39$
Jets	$0.004582x^3-0.1917x^2+1.121x+51.30$	$-0.01993x^2-0.5636x+54.46$
Eagles	$0.0186x^3-0.6197x^2+4.879x+53.54$	$0.08051x^2-1.987x+66.42$
Steelers	$0.005562x^3-0.2786+3.765x+51.75$	$-0.07003x^2+1.720x+55.59$
49ers	$0.001283x^3-0.07139x^2+1.115x+39.62$	$-0.1021x^2+1.986x+42.73$
Seahawks	$-0.007715x^3+0.2544x^2-1.597x+53.95$	$-0.03488x^2+1.240x+48.63$
Buccaneers	$-0.0006969x^3+0.2136x^2-5.233x+70.12$	$0.1875x^2-4.976x+69.64$
Titans	$-0.01832x^3+0.7637x^2-9.545x+82.01$	$0.07682x^2-2.811x+69.37$
Commanders	$0.001116x^3+0.04210x^2-1.938x+53.57$	$0.008596x^2-2.348x+54.35$

i *NFL Season To Feature 17 Regular-Season Games Per Team*. Retrieved 23/2/2025. Available on: https://www.profootballhof.com/news/2021/03/nfl-season-to-feature-17-regular-season-games-per-team/

ii *2023-24 NFL Playoffs*. Retrieved 24/2/2025. Available on: https://en.wikipedia.org/wiki/2023-24_NFL_playoffs

iii *New Orleans Saints Playoff History.* Retrieved 24/2/2025. Available on: https://www.pro-football-reference.com/teams/nor/playoffs.htm

iv *List of all the Pro Football Franchises.* Retrieved 24/2/2025. Available on: https://www.pro-football-reference.com/teams/

v *2018 Games – Week 1*. Retrieved 25/2/2025. Available on: https://www.nfl.com/scores/2018/reg1

vi *2018 Arizona Cardinals season.* Retrieved 21/2/2025. Available on: https://en.wikipedia.org/wiki/2018_Arizona_Cardinals_season

vii Yurko, R. *National Football League Team Statistics.* Retrieved 22/2/2025. Available on: *https://data.scorenetwork.org/football/nfl-team-statistics.html*

viii Vandenberghe, L. *Nonlinear least squares.* Retrieved 26/2/2025. Available on: https://www.seas.ucla.edu/~vandenbe/133A/lectures/nlls.pdf

ix Nathan and Myron. *Gauss-Newton Method.* Retrieved 6/4/2025. Available on: https://medium.com/@nathan.myron/gauss-newton-method-7dbb3d4ac32c

x *Linear regression: Gradient descent.* Retrieved 6/4/2025. Available on: https://developers.google.com/machine-learning/crash-course/linear-regression/gradient-descent

xi Gavin, H. *The Levenberg-Marquardt algorithm for non-linear least-squares curve-fitting problems.* Retrieved 26/2/2025. Available on: https://people.duke.edu/~hpgavin/lm.pdf

xii Ranganathan, A. *The Levenberg-Marquardt algorithm.* Retrieved 6/4/2025. Available on: https://sites.cs.ucsb.edu/~yfwang/courses/cs290i_mvg/pdf/LMA.pdf

xiii Tavora, M. *Proving Bernoulli's Sum of Powers.* Retrieved 6/4/2025. Available on: https://medium.com/data-science/proving-bernoullis-sum-of-powers-22f50df188e9

xiv Abdi, H. *The Kendall Rank Correlation Coefficient.* Retrieved 28/2/2025. Available on: https://personal.utdallas.edu/~herve/Abdi-KendallCorrelation2007-pretty.pdf

xv *Kendall Tau-b Correlation Coefficient.* Retrieved 7/4/2025. Available on: https://online.stat.psu.edu/stat509/lesson/18/18.3

xvi *2025 Opponents Determined.* Retrieved 27/2/2025. Available on: https://operations.nfl.com/updates/the-game/2025-opponents-determined/

xvii *New Orleans Saints Records by Opponent.* Retrieved 11/4/2025. Available on: https://www.footballdb.com/teams/nfl/new-orleans-saints/teamvsteam

xviii Greenberg, D. *NFL betting wrapped: Which teams, players and props attracted the most money this season?* Retrieved 11/4/2025. Available on: https://www.espn.com/espn/betting/story/_/id/43906434/2024-nfl-betting-season-round-playmakers-oddities-more

xix *2025-26 New Orleans Saints Odds: Odds to win Super Bowl, NFL Playoffs, Division.* Retrieved 11/4/2025. Available on: https://foxsports.com/articles/nfl/2025-26-new-orleans-saints-odds-odds-to-win-super-bowl-nfl-playoffs-division