The Value of an NBA Timeout

Eric Gonzalez
egonzalez86@gatech.edu
903651349

Luke Keohane
lkeohane3@gatech.edu
903553153

May 4, 2021

Contents:

Partition of Roles:

Sourcing data **- Luke**
Preprocess data **- Luke**
Researching models **- Eric/Luke**
Prototyping model implementation **- Eric**
Iterating through model designs **- Eric/Luke**
Final report writing **- Eric/Luke**

Problem Statement

In sports, conventional wisdom is to always save a timeout for the end of the game. The reason varies by sport; however, coaches typically hold onto one or more timeouts due to some perceived value. After winning a close game, if a coach was able to call a timeout to draw up a play and the team wins, she is typically lauded for her prudent thinking. Following a tough loss, if a coach does hold onto a timeout for the closeout seconds of the game, the media tends to criticize the coach harshly, and the coach may be hard on himself for not saving it. Our study seeks to shed some light on precisely that decision point: whether or not to preserve a timeout for late-game scenarios.

The focus of our experiment will be around the value of saving a timeout in the National Basketball Association (NBA). In basketball, there are scenarios towards the end of a close game in which one team can gain some discrete advantage by calling a timeout. A team may want to advance the ball to half-court, to prevent a turnover by getting out of a tight press coverage, or to draw up a play for the final seconds of the game. Clearly, there is some value to using a timeout; however, teams have a limited number of timeouts to use per game, so spending a timeout carries some opportunity cost. Thus, by quantifying the value of a late-game timeout, we can move one step closer towards eventually evaluating a coach's decision to save a timeout objectively in terms of costs-vs.-benefits.

Our experiment will take multiple approaches to quantifying the value of a late-game timeout. Formally, we will address the following questions:

- What is the marginal point value gained by using a timeout near the end of the game?
- Does entering a late, close game with an available timeout increase a team's probability of winning?

Furthermore:

- What classifier (i.e. model and set of predictors) best predicts whether a team will win?

By addressing each of these questions, we will be able to better understand if a timeout has value, what the true value of a timeout is, and if it increases the probability of winning a game.

Data Source

Primary data source for this study is the NBA Play-By-Play data available at eighthirtyfour.com/. This data is free to download and use for non-commercial purposes.

This data contains detailed records of in-game events such as shots, fouls, player substitutions, and – most relevant – timeouts. The Play-by-Play (PbP) data is available for regular-season games spanning 11 NBA seasons starting with the 2008-2009 season and ending with the 2018-2019 season.

Estimated sample size:

Although this data set contains almost 6.5 million observations (each with over one-hundred data points), our study is concerned with specific sequences of events in a specific set of scenarios.

The analyses we are planning to conduct are (1) game-level and (2) play-level, respectively.

1. Filtering to **games** that meet our criteria of being "close games" at any point with two minutes or less remaining, we estimate that we will have a sample size of approximately 3000-5000.
2. Filtering to **offensive possessions** with the same criteria, we estimate that we will have a sample size of approximately 9000-20000.

Data To-Do List:

- The raw data is quite large; in fact, the raw data is too large to simply ingest in its entirety into a python application. We will need to create a set of utilities to extract, transform, and load the data from its set of raw data files to a cleaned, condensed set of files that allow us to use a python application to efficiently aggregate them into a single dataset for analysis.
- In the original data, game events have been quantized; each individual event was assigned a numerically coded "type"; however, the original source did not provide a dictionary. We will need to manually inspect the data to infer the plain-language meaning of these codes and reconstruct a dictionary of our own in order to ensure that we have an accurate understanding of the data and what it represents (e.g. EVTMSGTYPE==1 -> made Field Goal).
- Because our analysis explores a potential cause-and-effect relationship, we will need to abstract sequences of event data into discrete observations that basically capture "game scenario and its corresponding outcome". To do this, we will first need to calculate several derived metrics for each game state, such as point-differential, timeouts remaining (per team), and the game outcome (win or loss). These derived metrics will be point-in-time, trailing cumulative, or forward-looking, respectively, and each will require us to leverage different calculation techniques. After computing these derived metrics for each game state, then we will reduce the data to a set of discrete, independent observations.

Methodology

For this project, the goal is to understand the value of a timeout in a late, close-game scenario. We plan on analyzing this by answering the two distinct questions above. Our approach to addressing each of those questions, respectively, is to (1) model the point value of using a timeout, and (2) model the marginal probability of victory for a team given one or more timeouts over the base case (having zero timeouts). Additionally, for the type of scenario that is the focus of this study, we seek to (3) identify what classifier best predicts whether a team will win.

We will use the output of the logistic regression (2) as feedback to refine the set of predictors that we will eventually use for comparing all our classifier models (3).

Models for Each Question:

1. ANOVA
    - To better understand the **marginal point value of a timeout on an offensive play**, we will use an ANOVA test to compare the group means of the **points scored on an offensive play** ***following a timeout*** versus the **points scored on an offensive play** ***not following a timeout*** in a close game.
2. Logistic Regression
    - To identify a team's **marginal probability of victory** by holding a timeout at the end, we will use logistic regression to calculate the probability estimate. We will also use logistic regression to **refine the set of predictors** we will use for all the classifier models.
3. Classifiers (comparison)
    - To identify which classifier **best predicts whether a team will win**, we will test logistic regression, KNN, SVM, and Neural Net models with training and test data to figure out which model has the best accuracy of the four.

ANOVA Methodology

We will split the play-level (a.k.a. "per-possession") data into categories for comparison:

- **lead_status** - [leading, tied, trailing] - is the team in possession of the ball leading
- **is_home** - [True, False] - is the team in possession of the ball the home team
- **poss_follow_to** - did the possession follow a timeout from the same team

After confirming the assumptions of the ANOVA test are met by calculating the summary statistics of the population for each group, we will construct 2-factor and/or 3-factor ANOVA tests to evaluate the difference in mean points scored across each bucket and interaction term.

We will look at the p-value of each term contained in the ANOVA tests to determine if there is a significant difference between groups in the sample means, thereby determining whether the **marginal point value of a timeout on an offensive play** is significant.

Logistic Regression / Classifier comparison Methodology

All variables we will use to answer questions 2 and 3 above are all constructed from the perspective of one particular team. So each game appears in the data twice: once from the perspective of the home team and once from the perspective of the visiting team. These variables are expressed as follows:

Dependent Variable: **did_win_game** (binary)

Independent Variables:
- **t_remaining_s** - time remaining in seconds
- **game points per min** - in that game, total points per minute scored by both teams combined
- **rel_score_margin** - point differential between the two teams (from the perspective of one team)
- **team_rem_timeouts** - the total timeouts remaining for a team
- **opp_rem_timeouts** - the total timeouts remaining for the opposing team

The process for selecting variables:

1. Run the logistic regression model with the full set of variables (t_remaining_s, game points per min, rel_score_margin, team_rem_timeouts, opp_rem_timeouts)
2. Evaluate the results of the model and the significance of each of the variables
3. Remove the insignificant variables with little impact
4. Repeat steps 2-3 until all the variables are significant

To answer the second question and third questions with various models, we will split our data 80% test and 20% train to check the accuracy of the model by using the expected outcome of the game as our test result in a confusion matrix.

Implementation and Analysis

Experiment 1: ANOVA

To better understand the **point value of a timeout**, we used an ANOVA test to compare the group means of the expected points for an offensive play following a timeout versus the expected points for an offensive play following no timeout in a close game. We split these groups based on whether a team was home or away, as well as whether a team was leading, trailing, or tied.

We took several steps to carefully select the dataset for the ANOVA test.

First, we used 2-sample Q-Q plots w.r.t. variable 'time remaining' (of various cross-sections of our dataset to identify the sample subset which strikes the best balance between controlling for confounding variables as much as possible vs. using as much data as is relevant. *(see Q-Q plots in appendix)*

- The Q-Q chart in the very top left [Last 3 Minutes, Following Timeout (Home vs. Away)] shows how the different the density of timeouts was for Home vs. Away over the last 3 minutes of play; we believe this is attributable to a recently-implemented rule that incentives teams to call timeouts before the two-minute-remaining mark, and has a disproportionate impact on the Home team.
- The Q-Q plots in the third and fourth columns (Timeout vs. Regular Play) indicate that plays following timeouts are skewed to correspond to lower values of time remaining (i.e. timeouts are more likely to be called later). We suspect this bias may correlate with some unobserved heterogeneity in the data, and prevent us from making a true apples-to-apples comparison of Timeout Following-Plays and Regular Plays.

Thus, we elected to reduce our ANOVA sample to the subset of plays that occurred with `time_remaining` less than 30 seconds.

Next, we confirm the assumptions of the ANOVA test are met by calculating the summary statistics of the population for each group:

lead_status	is_home	poss_follow_to	count	mean	std	min	25%	50%	75%	max
leading	**FALSE**	**FALSE**	3680	1.3704	0.8789	0	1	2	2	4
		TRUE	1039	1.3465	0.8819	0	1	2	2	3
	TRUE	**FALSE**	4603	1.3632	0.9107	0	1	2	2	4
		TRUE	1141	1.3865	0.8918	0	1	2	2	4
tied	**FALSE**	**FALSE**	232	0.8966	1.0761	0	0	0	2	3
		TRUE	307	0.8013	1.0616	0	0	0	2	5
	TRUE	**FALSE**	248	0.8105	1.0573	0	0	0	2	3
		TRUE	317	0.9527	1.1085	0	0	0	2	4
trailing	**FALSE**	**FALSE**	3373	1.1355	1.2002	0	0	1	2	5
		TRUE	3083	1.0431	1.1928	0	0	0	2	6
	TRUE	**FALSE**	2879	1.1355	1.2019	0	0	1	2	6

	TRUE	2696	1.0924	1.2027	0	0	0	2	5

2-Factor ANOVA

The assumption of equal STD is met within **lead_status** categories, so our approach was to stratify the data by lead bucket, then conduct 2-factor ANOVA tests on the **is_home** and **poss_follow_to** variables for each **lead_status** subset.

Furthermore, the assumption of equal population for each sub-group was not met, so within each ANOVA test, we used downsampling to align the sample sizes across sub-groups.

ANOVA 2-factor (sliced to lead_status = tied)

	sum_sq	df	F	PR(>F)	signif
C(poss_follow_to)	0.009698	1	0.008404	0.926976	FALSE
C(is_home)	0.242457	1	0.210106	0.646792	FALSE
C(is_home):C(poss_follow_to)	4.552802	1	3.945326	0.047297	TRUE
Residual	1066.272	924			

ANOVA 2-factor (sliced to lead_status = leading)

	sum_sq	Df	F	PR(>F)	signif
C(poss_follow_to)	0.054139	1	0.068252	0.793912	FALSE
C(is_home)	0.000241	1	0.000303	0.986105	FALSE
C(is_home):C(poss_follow_to)	2.171559	1	2.737659	0.098084	FALSE
Residual	3293.44	4152			

ANOVA 2-factor (sliced to lead_status = trailing)

	sum_sq	df	F	PR(>F)	signif
C(poss_follow_to)	12.83234	1	8.93203	0.002808	TRUE
C(is_home)	1.425816	1	0.992448	0.319167	FALSE
C(is_home):C(poss_follow_to)	1.869807	1	1.30149	0.253967	FALSE
Residual	15487.26	10780			

Results from 2-Factor ANOVA

Based on the ANOVA from the third slice (lead_status = trailing), possessions following timeouts were determined to have a different mean points scored compared to possessions not following timeouts (significant at the 5% level). While the result confirming a difference in means is consistent with our hypothesis, the direction is the opposite of what we expected: (when trailing) the mean points scored following timeouts is actually *significantly lower* than otherwise. While this could be attributable to some undetected confounding factor, this finding suggests that the conventional wisdom of calling a timeout to improve a play is misguided.

3-Factor ANOVA Iterations

Additionally, we used a 3-factor ANOVA to test the effects of different combinations if **is_home**, **lead_status**, and **poss_follow_to** had any significance on point totals. The primary purpose of this test was to confirm the quantitatively overwhelming effect of **lead_status** and confirm the need to stratify the data per above in order to isolate the **poss_follow_to** factor and thereby address our original question about timeouts.

- The first ANOVA looked at the individual variables compared against the interaction variable between all three. Given the interaction variable was not significant, we then needed to compare the three variables with the interaction between two at a time.
- The second ANOVA looked at the individual variables compared against the interaction variables between two variables at a time. Any variables whose p-value was greater than 0.05 were removed for the third iteration.
- The third and final ANOVA looked at the remaining variables after the second iteration.

Experiment 1 – Iteration 3: 3-Factor ANOVA with variables and with interaction variables with p > 0.05 removed.

ANOVA (3-factor, statsmodels):

Variable	sum_sq	df	F	PR (>F)
intercept	3434.07471	1	3045.226	0.000
C(is_home, Sum)	0.9713	1	0.861	0.353
C(lead_status, Sum)	118.1300	2	52.377	0.000
C(poss_follow_to, Sum)	0.0920	1	0.082	0.775
C(is_home, Sum):C(poss_follow_to, Sum)	0.0057	1	0.005	0.943
Residual	3132.7263	2778		

Results from 3-Factor ANOVA

Based on the ANOVA from the final iteration, lead_status is the only variable that has a significant effect on the point totals. Whether a team is home, if the possession is following a timeout, and the combination of those variables has no significant effect on the outcome of the points total. Given this, we are **unable to reject the null hypothesis** that holding onto a timeout carries any significant value on point totals in a game.

Experiment 2: Logistic Regression

We used logistic regression to identify a team's **marginal probability of victory** by holding a timeout at the end of a game by calculating the probability estimate. We also used logistic regression to **refine the set of predictors** we used for all the classifier models in the third experiment below.

Variable Overview

In the first iteration of the Logistic Regression model, we fit on all six initial variables. At that point, we closely examined the model output, asked ourselves whether the results aligned with our intuition, and what we may be overlooking in our model construction. In this way, we were able to use the model output as feedback to refine the variables we selected in subsequent model iterations.

For example: in iterations subsequent to the first, **t_remaining_s** and **game_points_per_min** were removed as individual variables. However, in iterations 3 and 4, they were factored in via the addition of interaction terms: in iteration 3, dividing the **rel_score_margin** by **t_remaining_s** to create the variable **point_margin_per_min_rem**; and in iteration 4, replacing **point_margin_per_min_rem** with **std_point_margin_per_min_rem**, a term that attempted to control for the average scoring rate of the teams in that particular game.

Original six variables and our takeaways from the initial fit:

- **const (intercept)-** this variable was essentially 0.
    - Whether or not this is valid depends on what the expected odds would be to model the output if every variable were equal to 0.
    - With this particular set of predictors, the fitted value of 0 makes sense. Given x_j=0 for each predictor j, we would expect the model to output odds of 1:1 (i.e. each team should have a 50-50 shot at winning), which corresponds to an intercept of 0.

- **t_remaining_s** - this variable essentially had basically no predictive value.
    - The lack of predictive value makes sense given that **considered independently,** the time remaining should not affect the odds of a particular team winning. not **directly**, anyways. another way of thinking about this: imagine the model if this were the only predictor.
    - Dropped after the first iteration.

- **Game points per min -** this variable essentially had basically no predictive value.
    - The lack of predictive value makes sense given that, **considered independently,** the rate of points scored by both teams should not affect the odds of a particular team winning, at least **not directly**.
    - Dropped after the first iteration.

- **Rel_score_margin -** this variable was a strong predictor and the direction is intuitive.
    - Though the magnitude of the coefficient is a little lower than expected, 0.5496, intuitively, it feels like an overly simple predictor. Once the results are interpreted, it is easy to see how

rel_score_margin affects the odds of winning (e.g. a 7-point lead gives a team an ~80% chance of winning); however, it is clearly also a function of how much time is left.
    - Kept after the second iteration as part of newly created variables in the third and fourth iterations (see below).

- **team_rem_timeouts -** this variable was a strong predictor and the direction is intuitive.
    - The magnitude of the predictor seems as expected, albeit a little lower. In a tie game, p goes from .500 to 0.545 when you go from 0 timeouts to 1 timeout.
    - Interpreting further, if the coefficients of the two predictors are used to convert timeouts to rel_score_margin, 0.5496 / 0.1784 ~= 0.308, it shows that a timeout is worth 0.308 ***net*** points. Technically, we can't make this claim because it violates the "holding all else equal" assumption of the model. Additionally, after the first iteration, once we drop the 'opp_timeouts_rem' variable, the ratio between the fitted coefficients **changes substantially.**
    - Kept through all iterations.

- **opp_rem_timeouts -** this variable was a strong predictor and the direction is intuitive.
    - The magnitude is obviously intuitive given it should be the inverse of the team_rem_timeouts. Given the inverse correlation, this variable may mean that we are "double-counting" given the home/away split in the data. Given the set of information contained in one observation, it's possible to impute all the information of a second data point with perfect accuracy. Dropping this variable, that would no longer be the case but the double-counting may remain an issue.
    - Tested with an without in each iteration.

Subsequently derived variables:

- **point_margin_per_min_rem -** strong predictor created in the third iteration, factoring in both the relative score margin and time remaining.
    - Formula for **point_margin_per_min_rem** = rel_score_margin / (t_remaining_s / 60).
    - By combining the predictors, the time remaining is accounted for within the relative score margin and provides an expected positive coefficient. The variable also remains significant each of the tests.
- **std_point_margin_per_min_rem -** strong predictor created in the fourth iteration, again factoring in both the relative score margin and time remaining.
    - Formula for **point_margin_per_min_rem** = rel_score_margin / (t_remaining_s / 60) / game_points_per_min * 4.17 (the global average).
    - Once again, the time remaining is accounted for within the relative score margin and provides an expected positive coefficient. The variable also remains significant each of the tests.

Ultimately, we reached the following Logistic Regression model for predicting whether or not a team would win a game:

Experiment 2 - Iteration 4b: Logistic Regression with Team Timeout Variable only and Std. Points Margin per Minute.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288288
Method:	MLE	**Df Model:**	2
Date:	5/2/21	**Pseudo R-squ.:**	0.4742
Time:	21:04:39	**Log-Likelihood:**	-1.05E+05
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.2716	0.019	-14.230	0.000	-0.309	-0.234
team_rem_timeouts	0.1105	0.007	14.882	0.000	0.096	0.125
points_margin_per_min_rem	0.4042	0.002	220.859	0.000	0.401	0.408

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	**Predicted: Win**
Actual: Loss	31132	4907
Actual: Win	4864	33170

classification report (logistic regression, statsmodels, binary):

	precision	**recall**	**f1-score**	**support**
False	0.86	0.86	0.86	36039
True	0.86	0.87	0.86	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Results of Experiment 2: Logistic Regression

After running the logistic regression model with variable selection through four iterations, the final variables used were team_rem_timeouts and std_point_margin_per_min_rem with each of the variables having a **p-value of < 0.000.** The final logistic regression model obtained a **high accuracy of 86%** and a pseudo R-squared value of 0.4742, which shows an **excellent model fit** with the selected variables**.**

Experiment 3: Comparing Models

To identify which classifier **best predicts whether a team will win**, we tested logistic regression, KNN, SVM, and Neural Net models with training (80%) and test data (20%) to figure out which model has the best accuracy of the four.

Comparing the Models

Experiment 3 – Model 1: Logistic Regression

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	**Predicted: Win**
Actual: Loss	31132	4907
Actual: Win	4864	33170

classification report (logistic regression, statsmodels, binary):

	precision	**recall**	**f1-score**	**support**
False	0.86	0.86	0.86	36039
True	0.86	0.87	0.86	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Experiment 3 – Model 2: KNN

confusion matrix (KNN, statsmodels, binary):

	Predicted: Loss	**Predicted: Win**
Actual: Loss	31271	4768
Actual: Win	5971	30063

classification report (KNN, statsmodels, binary):

	precision	**recall**	**f1-score**	**support**

False	0.84	0.87	0.85	36039
True	0.86	0.83	0.85	36034
accuracy			0.85	72073
macro avg	0.85	0.85	0.85	72073
weighted avg	0.85	0.85	0.85	72073

Experiment 3 – Model 3: SVM

confusion matrix (SVM, statsmodels, binary):

	Predicted: Loss	**Predicted: Win**
Actual: Loss	31137	4902
Actual: Win	4869	33165

classification report (SVM, statsmodels, binary):

	precision	**recall**	**f1-score**	**support**
False	0.86	0.86	0.86	36039
True	0.86	0.86	0.86	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Experiment 3 – Model 4: Neural Net

confusion matrix (Neural Net, statsmodels, binary):

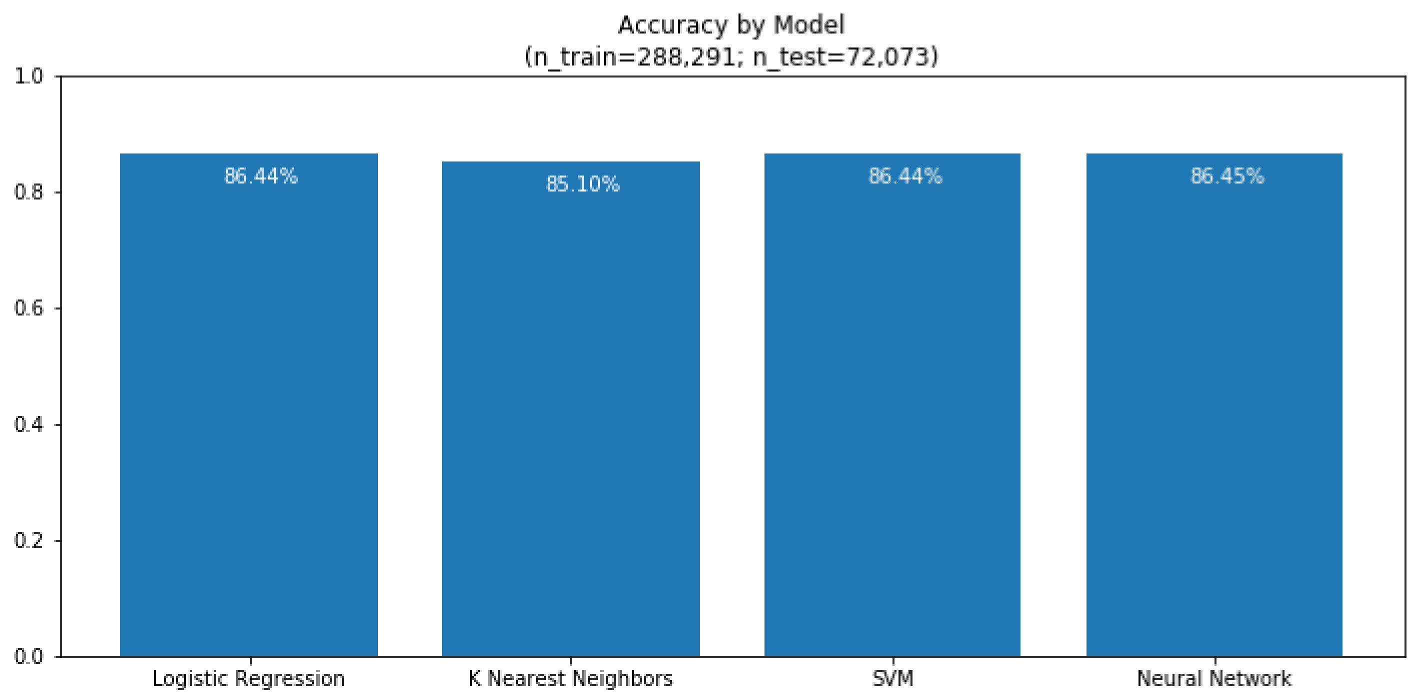
	Predicted: Loss	**Predicted: Win**
Actual: Loss	30440	5599
Actual: Win	4167	31867

classification report (Neural Net, statsmodels, binary):

	precision	**recall**	**f1-score**	**support**

False	0.88	0.84	0.86	36039
True	0.85	0.88	0.87	36034
accuracy			0.86	72073
macro avg	0.87	0.86	0.86	72073
weighted avg	0.87	0.86	0.86	72073

Results of Experiment 3: Comparing Classifiers



Comparing the four models, Neural net (86.45%) has a **slight edge in accuracy** over logistic regression and SVM (86.44%). The narrow margin between the two is negligible, however, and the logistic regression model is fitted in 0.3 seconds, which is **98.3% faster** than the Neural net model, which took 17.9 seconds to fit. Logistic regression was also **98.9% faster** than the SVM model, which took 27 seconds to fit.

Given the significant speed advantage logistic regression has over the other two models, and its high accuracy, **logistic regression is the best model** to use to predict whether a team will win.

Final Results and Evaluation

For this study, we sought to find out whether or not a coach in the NBA should preserve a timeout for late-game scenarios. To answer this question, we tested against three questions we had on the onset to quantifying the value of a late-game timeout.

- What is the marginal point value gained by using a timeout near the end of the game?
- Does entering a late, close game with an available timeout increase a team's probability of winning?
- What classifier (i.e. model and set of predictors) best predicts whether a team will win?

Using ANOVA, we investigate points scored per play at the end of close games under different combinations of factors to see if there was a significant difference in points scored following a timeout versus not following a timeout. The other factors we considered were whether or not the team was home and the lead status.

Upon reviewing the summary statistics for the subgroups to be used in the ANOVA test, we perceived a large discrepancy in the variance of the points scored per play depending on lead_status group, so we initially elected to stratify the data by lead_status and conduct 2-factor ANOVA tests considering whether a team is home, if the possession is following a timeout, and/or the combination of those two variables. The results indicated that, when tied or leading, plays following timeouts do not score a significantly different number of points, on average. However, when trailing, possessions following timeouts were determined to have a different mean points scored compared to possessions not following timeouts (significant at the 5% level). While the result confirming a difference in means is consistent with our hypothesis, the direction is the opposite of what we expected: (when trailing) the mean points scored following timeouts is actually significantly lower than otherwise. While this could be attributable to some undetected confounding factor, this finding suggests that the conventional wisdom of calling a timeout to improve a play is misguided.

The 3-factor ANOVA test considering all factors + interaction terms confirmed that lead_status overwhelms the other factors and renders them insignificant. Without controlling for lead_status, whether a team is home, if the possession is following a timeout, and/or the combination of those variables all have no significant effect on the outcome of the mean points per possession. From this, we were unable to reject the null hypothesis that - at the end of close games in general - using a timeout carries a significant boost to points scored for the following play.

We used logistic regression to test whether entering a close, late game with an available timeout increases a team's probability of winning. After running four iterations, two variables remained - std_points_margin_per_min and team_rem_timeouts. Given the timeouts remaining for a team were one of the two significant variables for the final logistic regression model, it did have a statistically significant impact on the overall probability of winning. The final model was 86% accurate and obtained a pseudo R-squared value of 0.4742, which further proved the model had an excellent fit. We used the final variables selected for in the logistic regression experiment to test the classifier accuracy in the final experiment.

There were four classifiers tested for in the final experiment - logistic regression, KNN, SVM, and Neural net. We used an 80% / 20% split between training and testing data to observe how accurate each of the models were. Logistic regression, SVM, and Neural net all obtained about the same accuracy at 86.44%, 86.44%, and 86.45% respectively. The key difference between the three was the amount of time each took to run. Logistic regression ran the fastest of the three models at 0.3 seconds and though it ran for 27 seconds, we had to cap the max iterations of SVM at 1,000 runs as the model would not converge. Given the similar accuracy and cheaper cost to run logistic regression, that was the best model to run to predict whether a team will win.

Each of the experiments used a large enough sample size that we feel confident in the results. Ten years of play-by-play data were used, therefore, this should reduce variability and should eliminate any potential confounding variables from biasing the data.

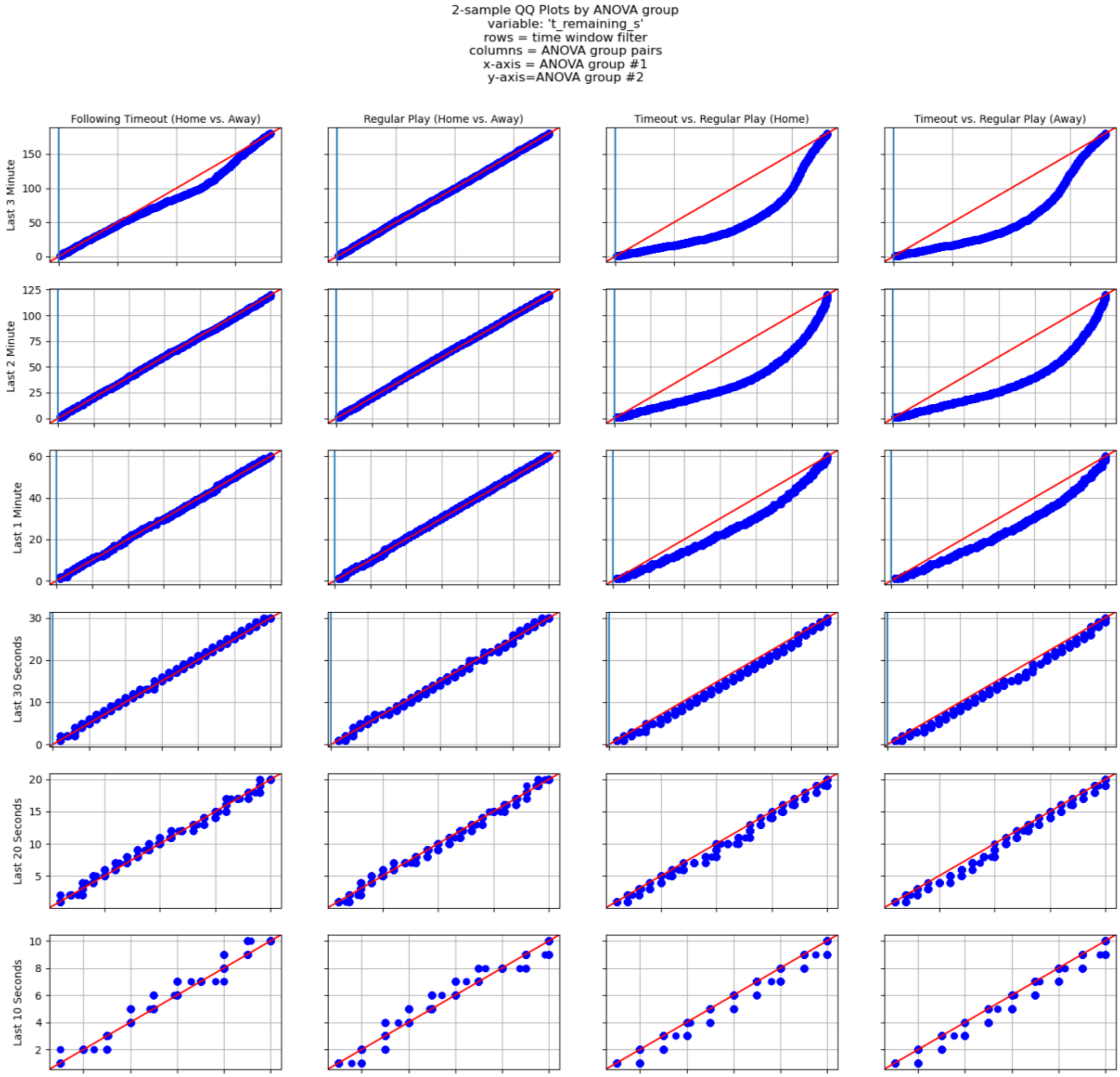
To improve our results for the future, the quality of the team holding a timeout should be accounted for. If a team has excellent players who have a higher likelihood of making a good play, the timeout may be more valuable to hold rather than a team that is going to purely rely on luck to win a close game. Also, the coach calling the timeout may be a consideration with the assumption that a good coach can draw up a better play with more time whereas a bad coach may have no impact with added time. We also may want to look at the different quarters of play and the specific time buckets within quarters to see the impact using a timeout has on a granular-level.

Our findings could be used to inform specific teams on when the right time or situation is to use a timeout. It can also be used for sports betting to know given certain criteria, what the odds are for a team to win and whether or not to bet on that team.

Appendix:

1. ANOVA Data Prep - Q-Q plots of ANOVA group pairs w.r.t. t_remaining_s
2. ANOVA - Intermediate Iterations of 3-Factor test
3. (a-f) Logistic Regression - Intermediate Iterations, refining variable selection

Appx 1: ANOVA Data Prep - Q-Q plots of ANOVA group pairs w.r.t. t_remaining_s



Appx 2: ANOVA - Intermediate Iterations of 3-Factor test

Experiment 1 - Iteration 1: 3-Factor ANOVA with all variables and the interaction variables between all three.

ANOVA (3-factor, statsmodels):

Variable	sum_sq	df	F	PR (>F)
Intercept	3403.0474	1	2961.725	0.000
C(poss_follow_to, Sum)	0.0000	1	0.000	1.000
C(lead_status, Sum)	111.5366	2	48.536	0.000
C(is_home, Sum)	0.4152	1	0.361	0.548
C(lead_status, Sum):C(poss_follow_to, Sum)	1.6185	2	0.704	0.495
C(is_home, Sum):C(poss_follow_to, Sum)	4.8333	1	4.207	0.040
C(lead_status, Sum):C(is_home, Sum)	0.7119	2	0.310	0.734
C(lead_status, Sum):C(is_home, Sum):C(poss_follow_to, Sum)	0.7852	2	0.342	0.711
Residual	3185.0517	2772		

Experiment 1 - Iteration 2: 3-Factor ANOVA with variables and the interaction variable between two variables at a time and without the interaction variable between all three.

ANOVA (3-factor, statsmodels):

Variable	sum_sq	df	F	PR (>F)
Intercept	3383.1756	1	2986.117	0.000
C(poss_follow_to, Sum)	0.6038	1	0.533	0.465
C(lead_status, Sum)	114.0797	2	50.346	0.000
C(is_home, Sum)	0.9343	1	0.825	0.364
C(lead_status, Sum):C(poss_follow_to, Sum)	5.7895	2	2.555	0.078
C(is_home, Sum):C(poss_follow_to, Sum)	8.6297	1	7.617	0.006
C(lead_status, Sum):C(is_home, Sum)	4.9332	2	2.177	0.114
Residual	3142.8542	2774		

Appx 3a: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 1: Logistic Regression with all variables.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288285
Method:	MLE	**Df Model:**	5
Date:	5/1/21	**Pseudo R-squ.:**	0.5517
Time:	11:44:19	**Log-Likelihood:**	-89591
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.0434	0.068	-0.637	0.524	-0.177	0.090
t_remaining_s	8.73E-05	0.000	0.683	0.494	0.000	0.000
game_points_per_min	0.0080	0.015	0.552	0.581	-0.020	0.036
rel_score_margin	0.5496	0.002	266.502	0.000	0.546	0.554
team_rem_timeouts	0.1784	0.008	22.041	0.000	0.162	0.194
opp_rem_timeouts	-0.1757	0.008	-21.701	0.000	-0.192	-0.160

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	30861	5178
Actual: Win	4571	31463

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.87	0.86	0.86	36039
True	0.86	0.87	0.87	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Appx 3b: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 2a: Logistic Regression with Score Margin and All Timeout Variables.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288287
Method:	MLE	**Df Model:**	3
Date:	5/1/21	**Pseudo R-squ.:**	0.5517
Time:	11:48:34	**Log-Likelihood:**	-89591
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.0086	0.020	-0.439	0.661	-0.047	0.030
rel_score_margin	0.5496	0.002	266.502	0.000	0.546	0.554
team_rem_timeouts	0.1795	0.008	23.604	0.000	0.165	0.194
opp_rem_timeouts	-0.1746	0.008	-22.948	0.000	-0.190	-0.160

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	30659	5380
Actual: Win	4360	31674

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.88	0.85	0.86	36039
True	0.85	0.88	0.87	36034
accuracy			0.86	72073
macro avg	0.87	0.86	0.86	72073
weighted avg	0.87	0.86	0.86	72073

Appx 3c: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 2b: Logistic Regression with Score Margin and Team Timeout Variable only.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288288
Method:	MLE	**Df Model:**	2
Date:	5/1/21	**Pseudo R-squ.:**	0.5503
Time:	11:49:56	**Log-Likelihood:**	-89597
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.2445	0.017	-14.618	0.000	-0.277	-0.212
rel_score_margin	0.5584	0.002	273.585	0.000	0.554	0.562
team_rem_timeouts	0.1098	0.007	15.841	0.000	0.096	0.123

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	31121	4918
Actual: Win	4856	31178

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.87	0.86	0.86	36039
True	0.86	0.87	0.87	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Appx 3d: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 3a: Logistic Regression with All Timeout Variables and Points Margin per Minute.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288287
Method:	MLE	**Df Model:**	3
Date:	5/1/21	**Pseudo R-squ.:**	0.475
Time:	11:51:03	**Log-Likelihood:**	-1.05E+05
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.0097	0.023	-0.416	0.677	-0.055	0.036
team_rem_timeouts	0.1627	0.008	20.380	0.000	0.147	0.178
opp_rem_timeouts	-0.1584	0.008	-19.814	0.000	-0.174	-0.143
points_margin_per_min_rem	0.4014	0.002	216.071	0.000	0.398	0.405

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	30967	5072
Actual: Win	4711	31323

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.87	0.86	0.86	36039
True	0.86	0.87	0.87	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Appx 3e: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 3b: Logistic Regression with Team Timeout Variable only and Points Margin per Minute.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288288
Method:	MLE	**Df Model:**	2
Date:	5/1/21	**Pseudo R-squ.:**	0.474
Time:	11:52:33	**Log-Likelihood:**	-1.05E+05
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.2674	0.023	-0.416	0.677	-0.305	-0.230
team_rem_timeouts	0.1084	0.008	20.380	0.000	0.094	0.123
points_margin_per_min_rem	0.4069	0.002	220.652	0.000	0.403	0.411

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	31129	4910
Actual: Win	4862	31172

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.86	0.86	0.86	36039
True	0.86	0.87	0.86	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073

Appx 3f: Logistic Regression - Intermediate Iterations, refining variable selection

Experiment 2 - Iteration 4a: Logistic Regression with All Timeout Variables and Std. Points Margin per Minute.

Dep. Variable:	did_win_game	**No. Observations:**	288291
Model:	Logit	**Df Residuals:**	288287
Method:	MLE	**Df Model:**	3
Date:	5/1/21	**Pseudo R-squ.:**	0.475
Time:	11:55:08	**Log-Likelihood:**	-1.05E+05
converged:	True	**LL-Null:**	-2.00E+05
Covariance Type:	nonrobust	**LLR p-value:**	0.000

Variable	coef	std err	z	P >|z|	[0.025	0.975]
const	-0.0097	0.023	-0.419	0.675	-0.055	0.035
team_rem_timeouts	0.1671	0.008	21.007	0.000	0.152	0.183
opp_rem_timeouts	-0.1629	0.008	-20.441	0.000	-0.178	-0.147
std_points_margin_per_min_rem	0.3985	0.002	216.216	0.000	0.395	0.402

confusion matrix (logistic regression, statsmodels, binary):

	Predicted: Loss	Predicted: Win
Actual: Loss	30966	5073
Actual: Win	4708	31326

classification report (logistic regression, statsmodels, binary):

	precision	recall	f1-score	support
False	0.87	0.86	0.86	36039
True	0.86	0.87	0.86	36034
accuracy			0.86	72073
macro avg	0.86	0.86	0.86	72073
weighted avg	0.86	0.86	0.86	72073