Predicting UFC Fights

Group 1-9

Eric Beecher, Kyle Adams, Caden Dortch

Abstract

Problem Overview

The Ultimate Fighting Championship, or the UFC, brings together the highest level fighters in the world for the ultimate test of skill: an epic one on one battle that only ends when one competitor submits to the will of the other. Since the inception of the sport, spectators have tracked certain metrics on fighters such as height, weight, reach, etc, which have been used to predict potential winners. With recent developments in technology, these metrics have expanded to include almost every movement imaginable by the fighters from punches and kicks to takedowns and submission attempts. Our team has taken this vast amount of data and closely examined each fighter metric all with the end goal of predicting the winner of any given UFC matchup. An accurate model would prove interesting for enthusiasts, and potentially profitable for someone looking to place wagers.

Research Methods

The basic research method used to solve this problem can be summarized as follows: 1) data understanding, 2) data preparation, 3) modeling, and 4) model evaluation.

Summarized Findings

Our research revealed the most salient fighter metrics that aid in predicting a winner. These variables included but are not limited to a fighters age, their takedown abilities, average number of head-shots they land, and how they won their previous fights. Our final model uses logistic regression with the following outcomes: Recall = 54.1%; Precision = 34.1%; F-Measure = 41.8%; Specificity = 71.1%; Error = 32.6%; Accuracy = 67.4%; R2 = .1607; AUC = 68.9%.

Data

Description and Preparation

The original data that was refined and used in our models included 160 columns, over 3000 rows, and can be found on https://www.kaggle.com/rajeevw/ufcdata. Each record contains variables for each fighter; the more dominant fighter in the red corner, and the underdog being in the blue corner. A few important variables and their definitions can be viewed in the following table:

Attribute name	**Data type**	**Description**
yBinary	int	outcome variable
B-R_age	int	The difference in age of the fighters
B_avg_opp_TD_landed	float	blue corner avg opponent takedowns landed
B_avg_Head_landed	float	blue avg strikes to the head landed
B_avg_opp_DISTANCE_landed	float	blue avg distance of landed strikes
B_avg_HEAD_att	float	blue avg head strike attempts
R_avg_opp_GROUND_landed	float	red avg opponent strikes on the ground landed
B_avg_opp_CLINCH_att	float	blue avg opponent strikes in the clinch attempted
B_avg_OPP_BODY_landed	float	blue avg opponent body shots landed
B-R_Reach_cms	float	reach advantage, blue reach minus red reach
B_avg_opp_TOTAL_STR_landed	float	blue avg opponent total strikes landed
B-R_Height_cms	float	Height advantage, blue height minus red height

One of the first steps with this data set was to resolve missing data. Each row with more than 50% of the data missing was eliminated altogether. This was done because of the unpredictable nature of fights; we sought to build our models on actual data and thus chose this in favor of an average or other type of filler values. After eliminating these records, our models still had over 3000 rows of data to work with.

After some initial analysis which included correlation matrices, regression modeling, and coefficient examination, certain variables including fighter weight, number of draws, wins by doctor stoppage, and average leg kick attempts were dropped. Each excluded metric showed little correlation with the outcome variable, had insignificant coefficient p-values (and or coefficient weight), or overall had little to no impact on our models. A detailed description of each variable and why it was excluded can be found in the provided spreadsheet under the tab "Removed Vars".

Anomalies

In comparing quantile box plots with histograms, we were able to identify a few outliers in the dataset. We also used logical detection to look at data points that were very far removed from the standard data. As we had a large amount of data, we removed the small amount of records with outliers. While that data may have been correct, some of it was different enough to skew our predictions. Removing those helped us predict generic fights better.

Feature Engineering

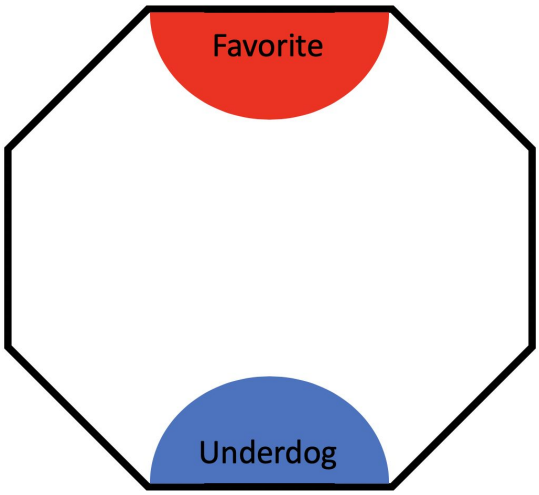
Most variables from the Kaggle dataset came pre-feature engineered. All metrics beginning with R_avg or B_avg represent average numbers up to the point in time when the fight took place. So for any given fighter, these averages could change with time. In addition to these averages, our

team was able to feature engineer two metrics of our own: height and reach advantage. As we ran models we realized that much like a fighter's weight, a fighter's reach did not hold much predictive information by itself. To solve this issue we created a new measure which subtracted the reach of one fighter from the other. This new metric, an easy to interpret difference in reach between the fighters, proved to hold much better predictive information. A similar approach was taken with fighter height, and these two variables greatly helped out our models.

Data Understanding

We initially thought that weight, or weight advantage could have an impact on the potential outcome of the fight, but we found that was not the case, as we studied we realized that the advantage doesn't exist because the fighters are already split up into separate weight classes.

As we understood this we were able to discount that variable and rely more on our metrics that we created, height and reach advantage, in order to account for physical imbalances between fighters



The other thing that we came to learn about our data was that fighters that were assigned to be in the red corner were winning much more than those in the blue corner. This was initially very confusing, if they

were assigned randomly then there should not have been a difference. So as we did more research we found that the favorite or higher seeded fighter is placed in the red corner, while the challenger or the underdog is placed in the blue corner. With this in mind, we were able to adjust our model to try and predict when an upset, or the fighter in the blue corner, would win.

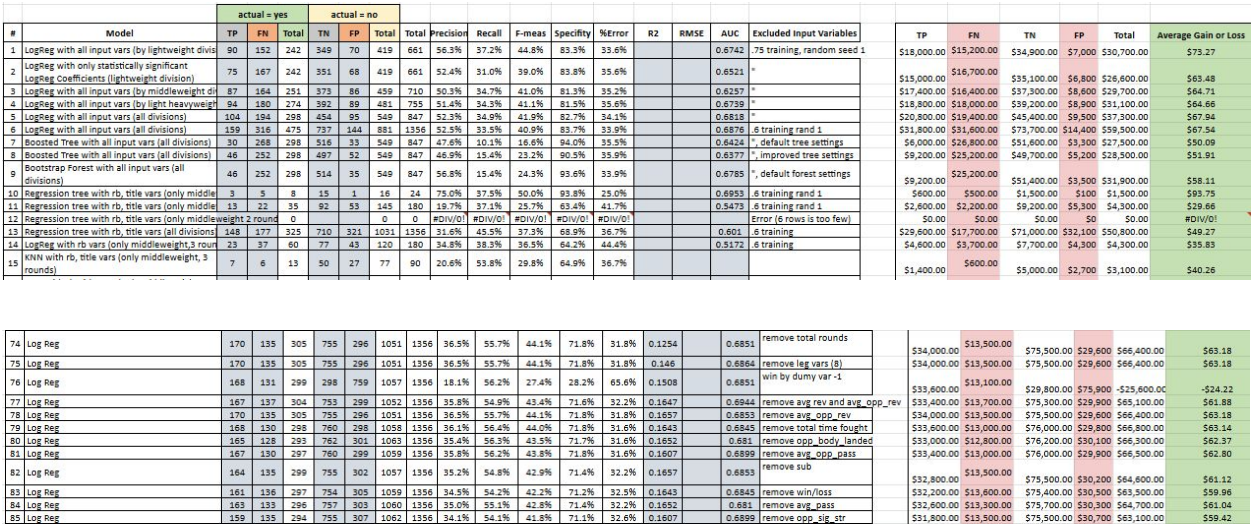
Input Variables and Models

Key Goals

What mattered most for our problem was being able to predict true positives which in this case was the underdog. To achieve this we paid special attention to the recall of our models. While precision also played a factor in our work, we maintained the perspective of an economic benefit from our model. All things equal, a higher recall is more important in betting on an underdog fighter. The worst outcome is going to generally be the same between a false positive and a false negative, as that money is lost either way.

Model Testing

As we had 160 input variables, we chose to not include a chart for each algorithm we ran. We collected data on 82 different attempts to change models and input variables to find which worked best. We removed 70 different input variables from the dataset. We used models such as MLR, CART, ANN, Neural Networks, Boosted Trees, KNN, and Random Forests. We found success with Random Forest, but wanted to get a better understanding of our input variables. Although we had other combinations featuring better recall rates, we decided to go with something offering us a higher R2 value, which in our case was a logistic regression. This gave us what we needed without sacrificing the quality of the model. It also enabled us to lose many variables that were not statistically significant.



#	Model	actual = yes TP	FN	Total	actual = no TN	FP	Total	Total	Precision	Recall	F-meas	Specifity	%Error	R2	RMSE	AUC	Excluded Input Variables	TP	FN	TN	FP	Total	Average Gain or Loss
1	LogReg with all input vars (by lightweight divis	90	152	242	349	70	419	661	56.3%	37.2%	44.8%	83.3%	33.6%			0.6742	.75 training, random seed 1	$18,000.00	$15,200.00	$34,900.00	$7,000	$30,700.00	$73.27
2	LogReg with only statistically significant LogReg Coefficients (lightweight division)	75	167	242	351	68	419	661	52.4%	31.0%	39.0%	83.8%	35.6%			0.6521	"	$15,000.00	$16,700.00	$35,100.00	$6,800	$26,600.00	$63.48
3	LogReg with all input vars (by middleweight di	87	164	251	373	86	459	710	50.3%	34.7%	41.0%	81.3%	35.2%			0.6257	"	$17,400.00	$16,400.00	$37,300.00	$8,600	$29,700.00	$64.71
4	LogReg with all input vars (by light heavyweigh	94	180	274	392	89	481	755	51.4%	34.3%	41.1%	81.5%	35.6%			0.6739	"	$18,800.00	$18,000.00	$39,200.00	$8,900	$31,100.00	$64.66
5	LogReg with all input vars (all divisions)	104	194	298	454	95	549	847	52.3%	34.9%	41.9%	82.7%	34.1%			0.6818	"	$20,800.00	$19,400.00	$45,400.00	$9,500	$37,300.00	$67.94
6	LogReg with all input vars (all divisions)	159	316	475	737	144	881	1356	52.5%	33.5%	40.9%	83.7%	33.9%			0.6876	.6 training rand 1	$31,800.00	$31,600.00	$73,700.00	$14,400	$59,500.00	$67.54
7	Boosted Tree with all input vars (all divisions)	30	268	298	516	33	549	847	47.6%	10.1%	16.6%	94.0%	35.5%			0.6424	", default tree settings	$6,000.00	$26,800.00	$51,600.00	$3,300	$27,500.00	$50.09
8	Boosted Tree with all input vars (all divisions)	46	252	298	497	52	549	847	46.9%	15.4%	23.2%	90.5%	35.9%			0.6377	", improved tree settings	$9,200.00	$25,200.00	$49,700.00	$5,200	$28,500.00	$51.91
9	Bootstrap Forest with all input vars (all divisions)	46	252	298	514	35	549	847	56.8%	15.4%	24.3%	93.6%	33.9%			0.6785	", default forest settings	$9,200.00	$25,200.00	$51,400.00	$3,500	$31,900.00	$58.11
10	Regression tree with rb, title vars (only middle	3	5	8	15	1	16	24	75.0%	37.5%	50.0%	93.8%	25.0%			0.6953	.6 training rand 1	$600.00	$500.00	$1,500.00	$100	$1,500.00	$93.75
11	Regression tree with rb, title vars (only middle	13	22	35	92	53	145	180	19.7%	37.1%	25.7%	63.4%	41.7%			0.5473	.6 training rand 1	$2,600.00	$2,200.00	$9,200.00	$5,300	$4,300.00	$29.66
12	Regression tree with rb, title vars (only middleweight 2 round	0					0	0	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!				Error (6 rows is too few)	$0.00	$0.00	$0.00	$0	$0.00	#DIV/0!
13	Regression tree with rb, title vars (all divisions	148	177	325	710	321	1031	1356	31.6%	45.5%	37.3%	68.8%	36.7%			0.601	.6 training	$29,600.00	$17,700.00	$71,000.00	$32,100	$50,800.00	$49.27
14	LogReg with rb vars (only middleweight,3 roun	23	37	60	77	43	120	180	34.8%	38.3%	36.5%	64.2%	44.4%			0.5172	.6 training	$4,600.00	$3,700.00	$7,700.00	$4,300	$4,300.00	$35.83
15	KNN with rb, title vars (only middleweight, 3 rounds)	7	6	13	50	27	77	90	20.6%	53.8%	29.8%	64.9%	36.7%					$1,400.00	$600.00	$5,000.00	$2,700	$3,100.00	$40.26

74	Log Reg	170	135	305	755	296	1051	1356	36.5%	55.7%	44.1%	71.8%	31.8%	0.1254		0.6851	remove total rounds	$34,000.00	$13,500.00	$75,500.00	$29,600	$66,400.00	$63.18
75	Log Reg	170	135	305	755	296	1051	1356	36.5%	55.7%	44.1%	71.8%	31.8%	0.146		0.6864	remove leg vars (8)	$34,000.00	$13,500.00	$75,500.00	$29,600	$66,400.00	$63.18
76	Log Reg	168	131	299	298	759	1057	1356	18.1%	56.2%	27.4%	28.2%	65.6%	0.1508		0.6851	win by dumy var -1	$33,600.00	$13,100.00	$29,800.00	$75,900	-$25,600.00	-$24.22
77	Log Reg	167	137	304	753	299	1052	1356	35.8%	54.9%	43.4%	71.6%	32.2%	0.1647		0.6944	remove avg rev and avg_opp_rev	$33,400.00	$13,700.00	$75,300.00	$29,900	$65,100.00	$61.88
78	Log Reg	170	135	305	755	296	1051	1356	36.5%	55.7%	44.1%	71.8%	31.8%	0.1657		0.6853	remove avg_opp_rev	$34,000.00	$13,500.00	$75,500.00	$29,600	$66,400.00	$63.18
79	Log Reg	168	130	298	760	298	1058	1356	36.1%	56.4%	44.0%	71.8%	31.6%	0.1643		0.6845	remove total time fought	$33,600.00	$13,000.00	$76,000.00	$29,800	$66,800.00	$63.14
80	Log Reg	165	128	293	762	301	1063	1356	35.4%	56.3%	43.5%	71.7%	31.6%	0.1652		0.681	remove opp_body_landed	$33,000.00	$12,800.00	$76,200.00	$30,100	$66,300.00	$62.37
81	Log Reg	167	130	297	760	299	1059	1356	35.8%	56.2%	43.8%	71.8%	31.6%	0.1607		0.6899	remove avg_opp_pass	$33,400.00	$13,000.00	$76,000.00	$29,900	$66,500.00	$62.80
82	Log Reg	164	135	299	755	302	1057	1356	35.2%	54.8%	42.9%	71.4%	32.2%	0.1657		0.6853	remove sub	$32,800.00	$13,500.00	$75,500.00	$30,200	$64,600.00	$61.12
83	Log Reg	161	136	297	754	305	1059	1356	34.5%	54.2%	42.2%	71.2%	32.5%	0.1643		0.6845	remove win/loss	$32,200.00	$13,600.00	$75,400.00	$30,500	$63,500.00	$59.96
84	Log Reg	163	133	296	757	303	1060	1356	35.0%	55.1%	42.8%	71.4%	32.2%	0.1652		0.681	remove avg_pass	$32,600.00	$13,300.00	$75,700.00	$30,300	$64,700.00	$61.04
85	Log Reg	159	135	294	755	307	1062	1356	34.1%	54.1%	41.8%	71.1%	32.6%	0.1607		0.6899	remove opp_sig_str	$31,800.00	$13,500.00	$75,500.00	$30,700	$63,100.00	$59.42

The image above is a small portion of our testing.

Interpretation of Input Variables

In using LogReg, we were able to see into the black box somewhat. However, because of the format of the data, each variable is tied to 7 other ones. For example, a fighter's average head-shot attempts is linked to how many they landed, how many their current opponent takes, and how many head-shots their previous opponents took. This leads to messy data where some variables need to stay because others are important to the model. However, over many hours we were able to find out which sets of variables could be removed. These include the percentage of significant strikes, leg-shots, rounds fought, win/loss streaks, submission attempts, stance, wins by doctor-stoppage and time fought. We also learned that there were very important factors for winning: age, ground-shots, distance, clinch and head-shots all were indicative of a potential upset. This means that the younger fighters who could get their opponents on the ground and hit their heads a lot had a much better chance of overcoming the odds. As mentioned earlier,

keeping weight classes together also helped make this project manageable in this time period and lead to an increase in our recall from 30% to near 60%.

Clusters

The most obvious subpopulations in our dataset were the different weight classes. We started out by trying out models for each of the different weight classes. This proved to be ineffective, as most of the fighters were in 3 of the weight classes and the other 12 classes had very few records, making it impossible to run certain models on all of the weight classes. We decided to not pursue clustering by weight class. We also looked at clustering by fighting style because we knew that existed and could see traces of that in the data, but we didn't have the time to investigate that further.

Economic Analysis

In an attempt to link our models to a monetary equivalent, we created a wager analysis to be automatically generated along with confusion matrix data from each model that we ran. To generate this analysis, we first had to define true positives, false negatives, true negatives, and false positives. These components along with their corresponding definitions can be interpreted as follows:

True positive: a bet on the underdog (blue corner) was successful

False negative: a bet on the favorite (red corner) was unsuccessful

True negative: a bet on the favorite was successful

False positive: a bet on the underdog was unsuccessful

Along with these definitions, we assumed a $100 dollar bet to be placed on each fight with +200 and -100 odds. These assumptions were based on the simple fact that successful bets on an

underdog will usually be more lucrative than wagers on the favorite. Although very simplified, these assumptions provided a solid basis in assessing each model as it ran. For a gambler using a given model, a true positive will double their money, a false negative will lose them $100, a true negative will make $100, and a false positive will also lose them $100. Our spreadsheet totals the gains, subtracts the losses, and also generates an average gain or loss per fight. Not being expert gamblers ourselves and as mentioned before, our team used this analysis as a simple assessment of model quality. In the real world odds for each fight would be very different, but for our purposes this model worked. When betting odds are changed, certain models will perform better than others. As a note of future improvement, our team would like to include actual betting odds for each fight in the dataset to generate a more accurate payout for each fight.

Excel estimator

As a first step towards implementing our model as a useful tool, our group created a web-service-based Excel estimator. To stay consistent with our previous model, we are still using a LogReg to make predictions here. Although not as fine tuned as our other models, this predictor still had a recall of 66% and an AUC of 69%. .

Future Research

Had we had more time, we would have looked at the various ways in which the different weight classes played into the data. We know that there are a variety of fighting styles, but due to the limited data on some of the weight classes and fighting styles we were not able to complete that analysis as of yet. We also would create a more real time input for the value of betting into the model. If it can have less simplistic rates of odds for the underdog, then we can create a model that would work better in a real life environment. Lastly, we would look at the difference in

fighting styles and what makes a fighter successful over time. All sports evolve with time, so giving precedence to more recent fights should give us more accurate predictions in the future. Of course, a large part of winning in fighting comes down to the mental aspect and things that can not be predicted before, but we believe that by refining our models further we could guess correctly enough to come out victorious.