Predicting NFL Outcomes Using Twitter Data

Josh Jiang & Declan McNamara

Department of Statistics
University of Michigan

April 21, 2021

Outline

1. Problem Statement & Goals
2. Assumptions
3. Data & Preprocessing
4. Classification Results
5. Extension: Gambling Analysis

Problem Statement & Goals

Motivation

- The project is centered on the phenomenon known as "the wisdom of crowds", first coined by James Surowiecki [1].
- The idea is that the collective judgment of a group is more accurate than that of any individual.
- In 1906, Francis Galton visited a livestock fair where he encountered a contest to guess the weight of an ox. About 800 people participated. When studying the entries, Galton noticed that while any individual entry could be way off the mark, the average of all the guesses was within 1 pound of true weight of the ox [1].

Goals

- We apply this theory to predicting the outcome of NFL games.
- For a given NFL game, we hope to extract the opinion of the public regarding the game and use that to make an informed prediction of its outcome.
- As an extension, we hope to develop a model that would be profitable when using it to gamble on games.

Difficulty

This is a very difficult problem for a variety of reasons. Viewed as a classification task, predicting a "Win" or "Loss" (class label 1 or 0) is nondeterministic and is complicated by the randomness of sports outcomes.

If we consider market consensus as the "gold standard" of this task, we can see how difficult the problem is. Betting the odds-on favorite for every NFL game in the 2020-2021 season results in a classification accuracy of only about 68.4%.

Using only Twitter language data to predict, we should not expect our model to perform as well as the market consensus, which presumably makes uses of magnitudes more information than our model.

Assumptions

Key Assumptions

Our biggest assumption is that Twitter language used when discussing the NFL is similar across the 2019-2020 and 2020-2021 seasons. ”Similar” includes notions of:

- Volume of tweets
- Sentiments about certain players and teams
- Common vernacular

Further, we assume that the observations are independent from each other, i.e. the outcome of one game does not affect the outcome of another game.

Data & Preprocessing

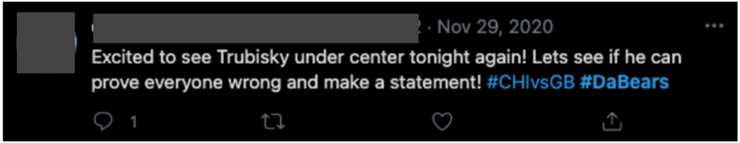
Gathering and Processing Tweets

- We collect the tweets that contains a hashtag associated with the home/away team up to 3 days before every game in a given season [2].
- Each tweet is reduced to a ”bag of words”. The frequency of each word is counted.
- We train on all 256 games from the 2019-2020 season, and test on all 256 games from the 2020-2021 season. The number of tweets used in the analysis is below.

	2019-2020	2020-2021
Home	353,398	326,344
Away	334,411	322,862

Figure: Tweet Volume

One Tweet Example



[redacted] [redacted] · Nov 29, 2020

Excited to see Trubisky under center tonight again! Lets see if he can prove everyone wrong and make a statement! #CHIvsGB #DaBears

```
['excited','see','trubisky','center','tonight','let',
'see','prove','everyone','wrong','make','statement',
'chivsgb','dabear']
```

```
{'excited':1,'see':2,'trubisky':1,'center':1,'tonight':1,
'let':1,'prove':1,'everyone':1,'wrong':1,'make':1,
'statement':1,'chivsgb':1,'dabear':1}
```

Generating the Corpus and Vectorizing

- We have a dictionary of words with their respective frequencies.
- A corpus of words is obtained by taking the words that have a frequency above a given threshold (eg. $> 0.1\%$ of all tweets).
- A corpus is developed for home and away teams separately.
- Using the corpus, we can get a vector representation of each game.
- This procedure can be naturally extended to n-grams. In our analysis, we used unigrams and bigrams.

$$G_i = [f_{i1}^H, \dots, f_{iN_H}^H, f_{i1}^A, \dots, f_{iN_A}^A]^T$$

G_i is the vector representation for the i^{th} game. f_{ij}^H is the frequency of the j^{th} word in the home corpus for that game. N_H and N_A is the size of the home and away corpus respectively. Note: in our analysis, f_{ij} is the log transform of $1 + \mathrm{frequency}$.

Final Data Matrix

- In the end, we obtain the final data matrix M to use for classification.
- The i^{th} row $M_i = [G_i, R_i]$, where $R_i \in \{0, 1\}$ is the indicator that the i^{th} game resulted in the home team winning.

$$M = \left[\begin{array}{cccc|cccc|c} f_{11}^{H} & f_{12}^{H} & & \cdots & f_{11}^{A} & f_{12}^{A} & & \cdots & R_1 \\ \vdots & \ddots & & & \vdots & \ddots & & & \vdots \\ f_{N1}^{H} & & & f_{NN_H}^{H} & f_{N1}^{A} & & & f_{NN_A}^{A} & R_N \end{array}\right]$$

Classification Results

Methods

- For the 256 games in the 2020-2021 season, our test set, the home team exactly 50% of the time. Given the naive strategy to pick the home team for every game, 0.5 will serve as the naive standard.
- The corpus is created from the 2019-2020 season.
- We use a suite of classification tools to train on the 2019-2020 season and test on the 2020-2021 season.
- Due to the high dimensionality of the data, we also used PCA to reduce the columns of the data matrix to a few PC scores.

Full Train Set Results

	Test Accuracy - Unigrams	Test Accuracy - Bigrams
Only Home	50.0%	50.0%
Logistic None	57.8%	57.4%
Logistic L1	58.2%	60.5%
Logistic L2	62.5%	61.3%
Adaboost	52.0%	56.3%
Random Forest	55.9%	59.0%
Naive Bayes	55.9%	53.1%
Neural Network	57.8%	59.4%

Figure: Results when using full training set

Dimensionality Reduction

	Test Accuracy - Unigrams	Test Accuracy - Bigrams
Only Home	50.0%	50.0%
Logistic None	60.5%	58.5%
Logistic L1	59.0%	57.8%
Logistic L2	60.9%	61.3%
Adaboost	54.7%	53.5%
Random Forest	61.3%	57.8%
Naive Bayes	52.7%	50.4%
Neural Network	54.7%	58.2%

Figure: Results when using training set with dimensionality reduction

Semantic Meaning: Unigrams

To provide a view of the types of unigrams models may see as useful, we examine some unigrams from the L1 penalized logistic regression from the previous slide.

Unigram	Home/Away
'drew'	H
'force'	H
'surgery'	A
'ranking'	H
'cold'	A
'turnover'	A
'michigan'	A
'ravensnation'	H

Figure: Selected Unigrams with Largest Positive Coefficients

Unigram	Home/Away
'terrible'	A
'turned'	A
'ravensnation'	A
'lmao'	H
'kcchiefs'	A
'watt'	A
'preseason'	H
'tyler'	A

Figure: Selected Unigrams with Largest Negative Coefficients

Semantic Meaning: Bigrams

To provide a view of the types of bigrams models may see as useful, we run an L1 penalized logistic regression which yields a coefficient vector with 26 nonzero entries, some of which are below.

Bigram	Coef. Sign	Home/Away
('super', 'bowl')	+1	H
('pro', 'bowl')	+1	H
('ravensflock', 'ravensnation')	+1	H
('injury', 'report')	-1	A
('next', 'year')	-1	A
('kansa', 'city')	-1	A
('next', 'year')	-1	A
('bengal', 'whodey')	+1	A
('russell', 'wilson')	-1	A

Figure: Selected Bigrams from L1 Penalized LR

Extension: Gambling Analysis

Beating the Market

Can we beat the market in this predictive task?

- Betting the odds-on favorite in every game of the 2020-2021 season results in an accuracy of 68.4%.

Idea: let our models use the money line as an input, in addition to Twitter data, to maximize accuracy.

$$M^* = \left[\begin{array}{ccc|ccc|cc|c} f_{11}^H & f_{12}^H & \cdots & f_{11}^A & f_{12}^A & \cdots & m_1^H & m_1^A & R_1 \\ \vdots & \ddots & & \vdots & \ddots & & \vdots & & \vdots \\ f_{N1}^H & & f_{NN_H}^H & f_{N1}^A & & f_{NN_A}^A & m_N^H & m_N^A & R_N \end{array}\right]$$

Prediction Results

We run the same models as before, now including the money line for the home/away teams to the models.

	Test Accuracy	# Underdogs
Only Favorite	68.4%	0
Logistic L1	68.4%	0
Logistic L2	68.4%	8
Adaboost	62.1%	84
Random Forest	64.5%	80
Neural Network	70.3%	31

Figure: Results with Money Lines

Have We Really Gained Anything?

- Based on these results, it seems like we don't really beat the market - or do we?
- Accuracy alone may not be the best metric to measure information gain.
- We employ a simple betting strategy where we bet $100 on our predicted team to win, and compute the payout from the various models.

	Test Accuracy	# Underdogs	Payout	Return
Only Favorite	68.4%	0	-$618.92	-2.4%
Logistic L1	68.4%	0	-$618.92	-2.4%
Logistic L2	68.4%	8	-$593.70	-2.3%
Adaboost	62.1%	84	$346.42	1.4%
Random Forest	64.5%	80	$1621.23	6.3%
Neural Network	70.3%	31	$1511.00	5.9%

Figure: Results & Payouts

References

- Surowiecki, James. The Wisdom of Crowds. Reprint, Anchor, 2005.
- Shiladitya Sinha, Chris Dyer, Kevin Gimpel, and Noah A. Smith. In Proceedings of the ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics, Prague, Czech Republic, September 2013.
- snscrape. https://github.com/JustAnotherArchivist/snscrape
- Pro-Football-Reference. https://www.pro-football-reference.com/
- Our Github. https://github.com/zjiang2/601_project

The End