Tom Brady Superbowl Prediction Model

Shashwat Maharjan, Lele Zhang, Brandon Panuccio, and and Satrant Bains

Villanova University

Tom Brady Superbowl Prediction Model

Introduction

American Football is by far the most popular sports in the United States with a Gallup poll revealing 41% of Americans say it is their favorite sport (Jeffrey, 2024). About 100 million Americans watch the Superbowl every year, this is nearly a third of all Americans. This viewership leads to one important thing, a lot of money. The NFL is by far the most profitable sports league in the world with the league generating $20.5 billion in revenue in the last year alone. It is worthwhile then to explore ways in which we may be able to predict whether one player may be able to reveal how strong a team actually is.

Tom Brady is widely regarded as the greatest player of all time with a wide margin between him and number two. He has all of the accolades, including 7 Superbowl wins, the most of all time. Brady also has the most regular season wins, passing yards, touchdowns, yards per game, etc. In trying to predict SB wins or not we decided to use the regular season statistics of Tom Brady. Our goal is to train the model on different windows of Tom Brady's career and predict whether he would win or not win the Superbowl.

Dataset

The dataset used during our training and testing was Pro Football Reference's page on Tom Brady (Reference, 2024). The main reason we chose this specific dataset is the amount of data that is contained on the site. There are dozens of different statistics that represent how well a quarterback has been playing including, Games Started, Completions, Completion %, Yds, TD, TD%, INT, INT%, First Downs.

We decided on removing a few of his seasons from analysis, specifically 2002 and 2008 because of two reasons. In 2002 Tom Brady was a rookie and did not start in any games and simply had no statistics to show for it. In 2008 Tom Brady tore his ACL and was out for almost the entire season besides 1 game. Both games would have affected our models as they showed not significant results.

Logistic Regression

Now that we have this data, we can begin implementing the simplest form of Machine Learning, Logistic Regression. Logistic regression is a foundational statistical method. Its mechanism of action is to understand and predict relationships between variables by finding a line of best fit that best represents the relationship between an independent variable and a dependent variable. In our case, logistic regression was utilized to assess how specific season stats for Brady could help predict whether his team would win the Super Bowl. The technique minimizes the error in the prediction by calculating the best-fit line that represents the relationship between his performance metrics and Super Bowl wins.

Models Used

The models chosen were the following, SkLearn, Pytorch DNN, Tensorflow DNN, and XGBoost Classifier (Abadi et al., 2016; Chen & Guestrin, 2016; Fabian, 2011; Paszke et al., 2019). The primary reason such a wide range of models were chosen is simply due to variety. The goal here was not only to predict wins but to also see if any of these models stands above the others in terms of accuracy. Moreover, we were hoping to see if any of these datasets could reveal anything about the dataset, whether it contained too many Superbowl wins or whether it is generally best suited for the goal we had in mind in the first place.

In addition to this, we chose Pytorch and Tensorflow DNN's because they are the exact same architecture, and we wanted to observe the output to see if they would be the same. XGBoost was specifically chosen due to the fact it is tree-based and on paper it should outperform black box neural networks on tabular data. One more thing to note would be that since DNNs require a lot more data compared to logistic regression and XGBoost which reflects on the prediction capabilities on the test dataset. Also, there is a chance of overfitting on neural networks where it starts to memorize the training dataset and not generalizing enough on the test dataset.

Dependencies/Tools

This project used several dependencies, including:

1. Pandas for data manipulation and preprocessing.
2. Scikit-Learn for implementing logistic regression and logistic regression.
3. PyTorch and TensorFlow for neural network development.
4. XGBoost for gradient boosting classification.
5. Matplotlib and Seaborn for data visualization to assess correlations and model outputs.

These libraries provided essential tools for managing data, building and training models, and evaluating performance.

Assessment of Data

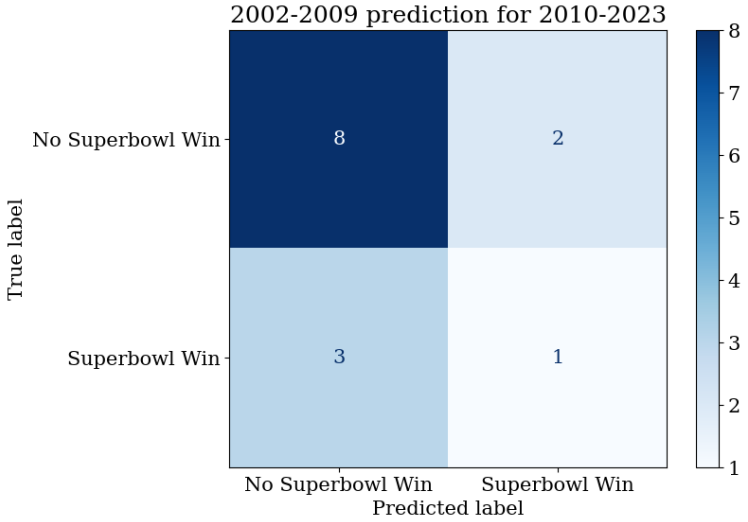
Before training the models, we conducted an assessment of the dataset to understand key patterns, trends, and potential biases. This included removing columns that would hinder the regression, such as outliers, and strings. By visualizing various statistics like Completion Percentage, Passing Yards, and Touchdowns per season, we identified correlations with Super Bowl wins. This initial analysis informed which features would likely be the strongest indicators of season success, helping guide our feature selection process for each model.

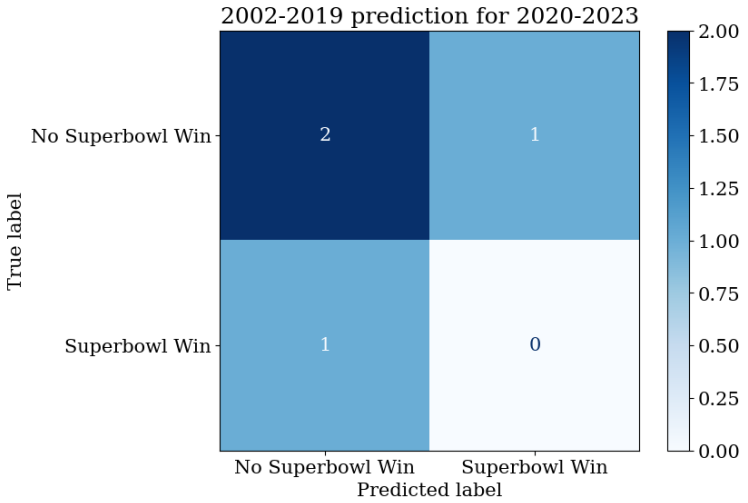
Conclusion

Our findings indicate that certain performance metrics for Tom Brady, particularly passing efficiency and completion rates, were more predictive of Super Bowl success than others. However, model performance varied, with the neural networks showing some improvement over linear methods but also requiring larger data segments for effective training. More outliers are present than just Brady's stats, however. As football is a team

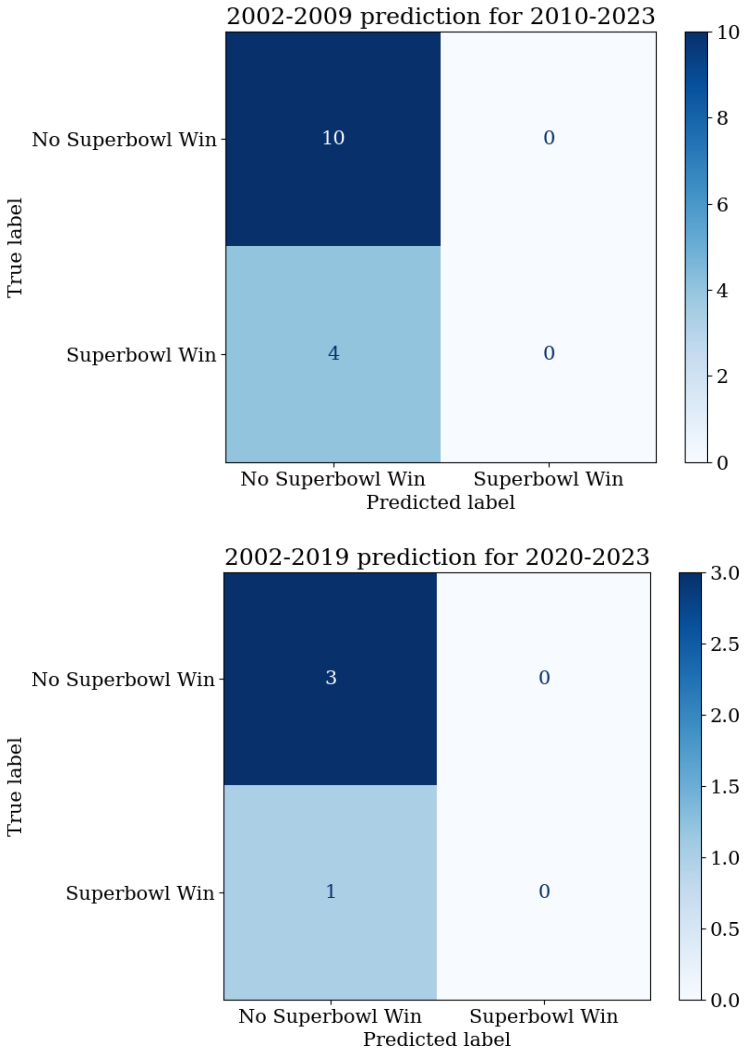
sport, the performance of Offense, Defense, and Special Teams can dictate how successful a team or player is in winning. Additionally, football is a one game, winner takes all sport. Other sports, like baseball and basketball play series of games which offer more data to support which team is the best. This logic holds especially true in the 2007 season, where Brady and the Patriot's went undefeated up until the Superbowl game itself, where he was defeated by Eli Manning and the New York Giants. Future directions may include expanding the dataset to include other team statistics, and applying model tuning to improve prediction accuracy. The python implementation, results, and the figures can be found on this https://github.com/shashwatmaharjan/tom-brady-superbowl-success.

SKLearn Confusion Matrix

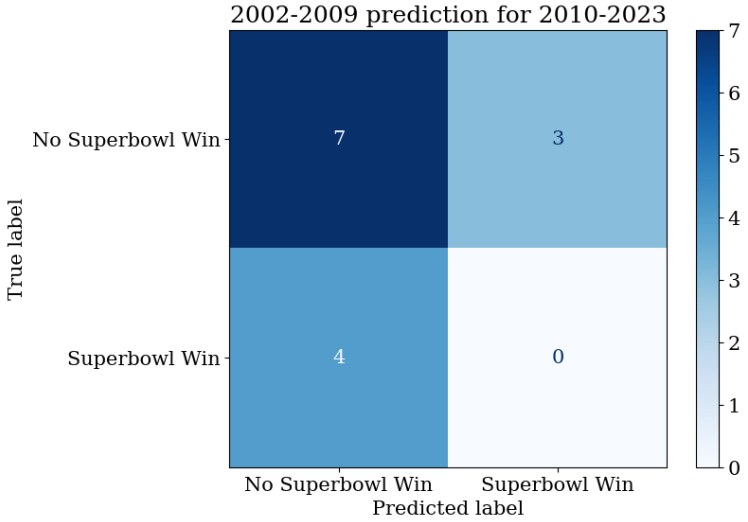


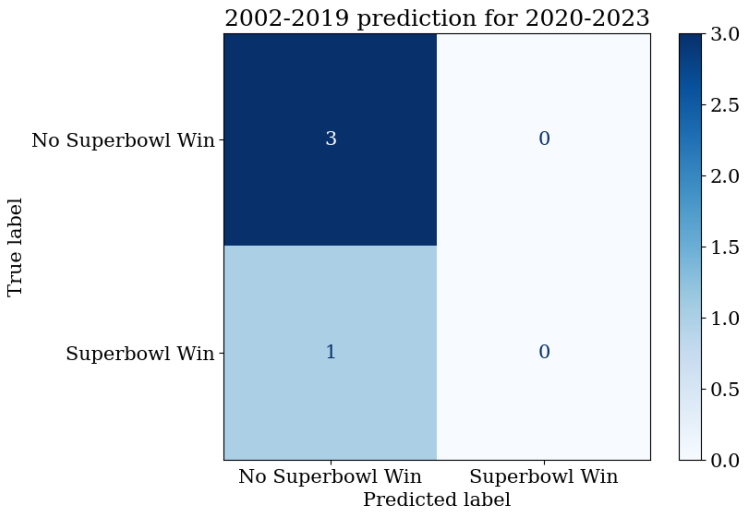


XGBoost Confusion Matrix

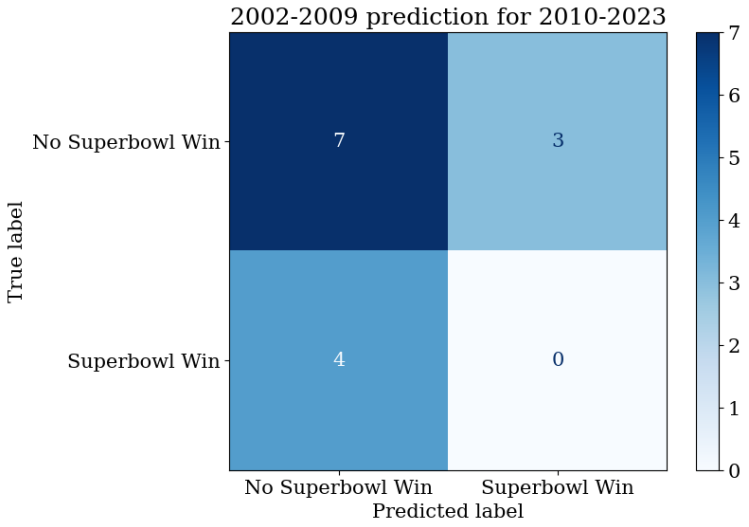


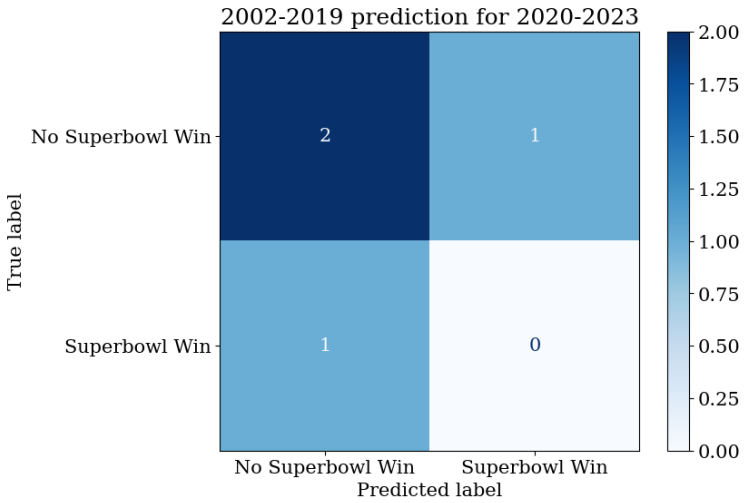
Pytorch DNN Confusion Matrix





Tensorflow DNN Confusion Matrix





References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). {Tensorflow}: A system for {large-scale} machine learning. *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, 265–283.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Fabian, P. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research 12*, 2825.

Jeffrey, J. (2024). Pro football reference. https://news.gallup.com/poll/610046/football-retains-dominant-position-favorite-sport.aspx.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, *32*.

Reference, P. F. (2024). Pro football reference. https://www.pro-football-reference.com/players/B/BradTo00.htm.