WSABI Summer Research Lab **Summer 2025**

Lab 18: Random Forests and Boosting

Instructor: Jonathan Pipping *Author: Ryan Brill*

18.1 Estimating In-Game NFL Win Probabilities

18.1.1 The Data

We will now implement boosting to estimate play-by-play NFL win probabilities, like Ben Baldwin did in [BB]. Our data (from the `nflfastR` package) contains the following information:

- `i`: the index of the i^{th} play
- `y_i`: the outcome of the game, which is 1 if the team in possession won and 0 otherwise
- `x_i`: the game state, which is a vector of the yard line, the down, the distance to the first down, the score differential, the seconds remaining in the game, the number of timeouts remaining for each team, and the pre-game point spread relative to the team in possession

18.1.2 Your Task

1. Fit a win probability model $\widehat{\mathrm{WP}}(x_i) = \widehat{\mathbb{P}}(\mathrm{win} \mid x_i)$ in `R` using either a Random Forest (`ranger`) or XGBoost (`xgboost`). Tune the parameters yourself using validation log-loss. Instructions for this are provided in [LN] for Random Forests and [BB] for XGBoost.
2. Visualize estimated win probabilities using **partial dependence plots (PDPs)**.
   (a) First, make a line plot of $\widehat{\mathrm{WP}}$ (y axis) vs yard line (x axis) for various values of score differential (color) and time remaining (facet), holding other covariates fixed.
   (b) Next, make a heatmap of $\widehat{\mathrm{WP}}$ (color) as a function of score differential (x axis) and time remaining (y axis), holding other covariates fixed.
   (c) Finally, make a line plot of $\widehat{\mathrm{WP}}$ (y axis) vs yard line (x axis) for various values of point spread (color) and time remaining (facet), holding other covariates fixed.
3. Quantify uncertainty in win probability point estimates by **bootstrapping**. Fit $B = 100$ bootstrapped WP models $\{\widehat{\mathrm{WP}}^{(b)}\}_{b=1}^{B}$. Note that due to the dependence structure in the data (all plays in a game have the same outcome), y_i is independent across games, but not within the same game. Therefore, to correctly mimic the data-generating process, you must sample **games** with replacement rather than individual rows (plays). This is known as **block bootstrapping**.
4. Create 95% confidence intervals for each play in the dataset. What is the distribution of confidence interval widths? How does this vary by time remaining and down? What does this tell you about the uncertainty in win probability estimates?
5. Create a line plot of $\widehat{\mathrm{WP}}$ (y axis) vs time remaining (x axis) for various values of score differential (color) and yardline (facet), holding other covariates fixed. Create shaded colored regions for the 95% confidence intervals using `geom_ribbon` in `ggplot2`. How wide are these confidence intervals?

References

[BB] Baldwin, B., *NFL Win Probability from Scratch using XGBoost in R*, April 16, 2021.

[BL] Breiman, L., *Arcing Classifiers*, The Annals of Statistics, Vol. 26, No. 3, pp. 801-824, 1998.

[BYW] Brill, R. S., Yurko, R., & Wyner, A. J., *Analytics, Have Some Humility: A Statistical View of Fourth-Down Decision Making*, The American Statistician, 2025.

[LN] Lock, D., & Nettleton, D., *Using Random Forests to Estimate Win Probability Before Each Play of an NFL Game*, Journal of Quantitative Analysis in Sports, Vol. 10, No. 1, pp. 1-14, 2014.