Northeastern University
College of Engineering

EECE5644: Final Project

Alex Montes McNeil
Nishanth Marer Prabhu

04/18/2023

Dataset

- NFL data since the year 2000 containing 30 features of each game played
- Full list of the dataset features is listed in our report
- Reference: https://github.com/ukritw/nflprediction

	schedule_date	schedule_season	schedule_week	schedule_playoff	team_home	score_home	score_away	team_away	team_favorite_id	spread_favorite	...	team_away_current_win_pct
0	2001-09-09	2001	1	0	BAL	17.0	6.0	CHI	BAL	-10.5	...	0.000000
1	2009-12-20	2009	15	0	BAL	31.0	7.0	CHI	BAL	-11.0	...	0.384615
2	2017-10-15	2017	6	0	BAL	24.0	27.0	CHI	BAL	-6.5	...	0.200000
3	2002-09-29	2002	4	0	BUF	33.0	27.0	CHI	BUF	-3.0	...	0.666667
4	2010-11-07	2010	9	0	BUF	19.0	22.0	CHI	CHI	-3.0	...	0.571429

Northeastern University
College of Engineering

NFL Home-Team-Win Classification Problem

Take Aways from Dataset

Number of Games: 4783
Home Straight Up Win Percentage: 57.57%
Away Straight Up Win Percentage: 42.43%
Under Percentage: 49.70%
Over Percentage: 48.55%
Equal Percentage: 1.76%
Favored Win Percentage: 65.96%
Cover The Spread Percentage: 46.96%
Against The Spread Percentage: 49.32%

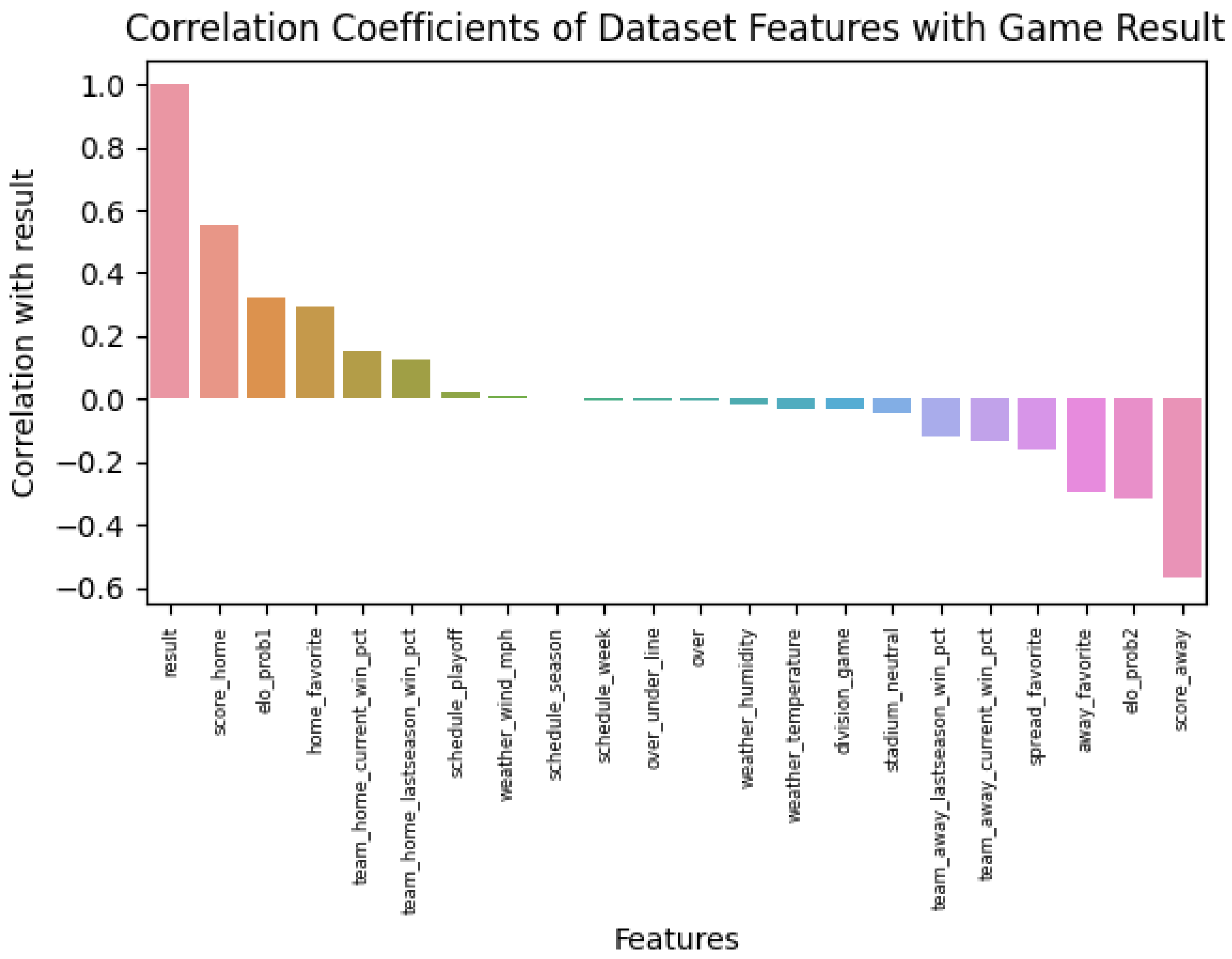
Apply ML Dataset

- Can we train a Multilayer Perceptron (MLP) to predict if the home team or away team will win a game based on this dataset?
  - Does increasing the number of hidden layers improve the model size?
- Use Expected Risk Minimization to determine the theoretical minimum probability of error.

Northeastern University
College of Engineering

Reduce the Size of the Dataset

- Calculated the correlation coefficients of all numerical features with result
- Chose to keep features with coefficient above +/- 0.05
    - Further reduction required to remove features the give away the result of the game
- Feautures used for classification:
    - elo_prob1/2
    - team_current_win_pct
    - team_lastseason_win_pct
    - home/away_favorite
    - spread_favorite



Expected Risk Minimization (ERM)

ERM Confusion Matrix	
0.50	0.23
0.50	0.77

- We will determine the mean and covariance matrix for each of the label set
- Assuming the underlying PDF as Gaussian, we will evaluate the PDF
- In this case we will use a 0-1 loss matrix and multiply it with the class posterior

$$P(X) = P(X|L=0) * P(L=0) + P(X|L=1) * P(L=1)$$

$$P(X|L=l) * P(L=l) \quad where\ l = 0,1$$

- The class posterior is given by the below equation:

$$P(L=l|X) = \frac{P(X|L=l) * P(L=l)}{P(X)} \quad where\ l = 0,1$$

- To determine the Risk Matrix, we need to multiply the Class Posterior $P(L=l|X) \quad where\ l = 0,1$ with the loss matrix λ_{01}

$$R(D=l|X) = \lambda_{01} * P(L=l|X) \quad where\ l = 0,1$$

- Finally, we take the argmin of the risk matrix

Northeastern University
College of Engineering

Multilayer Perceptron Model (MLP)

- MLPClassifier from scikit-learn python package
  - Number of hidden layers: 1
    - Find optimal number of neurons during cross validation
  - Activation function: relu
  - 10k Iterations (convergence was reached before the end of the iterations)
- N-fold cross-validation with GridSearchCV used to estimate number of neurons per layer
  - Varied number of folds to see the impact on the error of the model
    - [2, 5, 10, 15, 20]

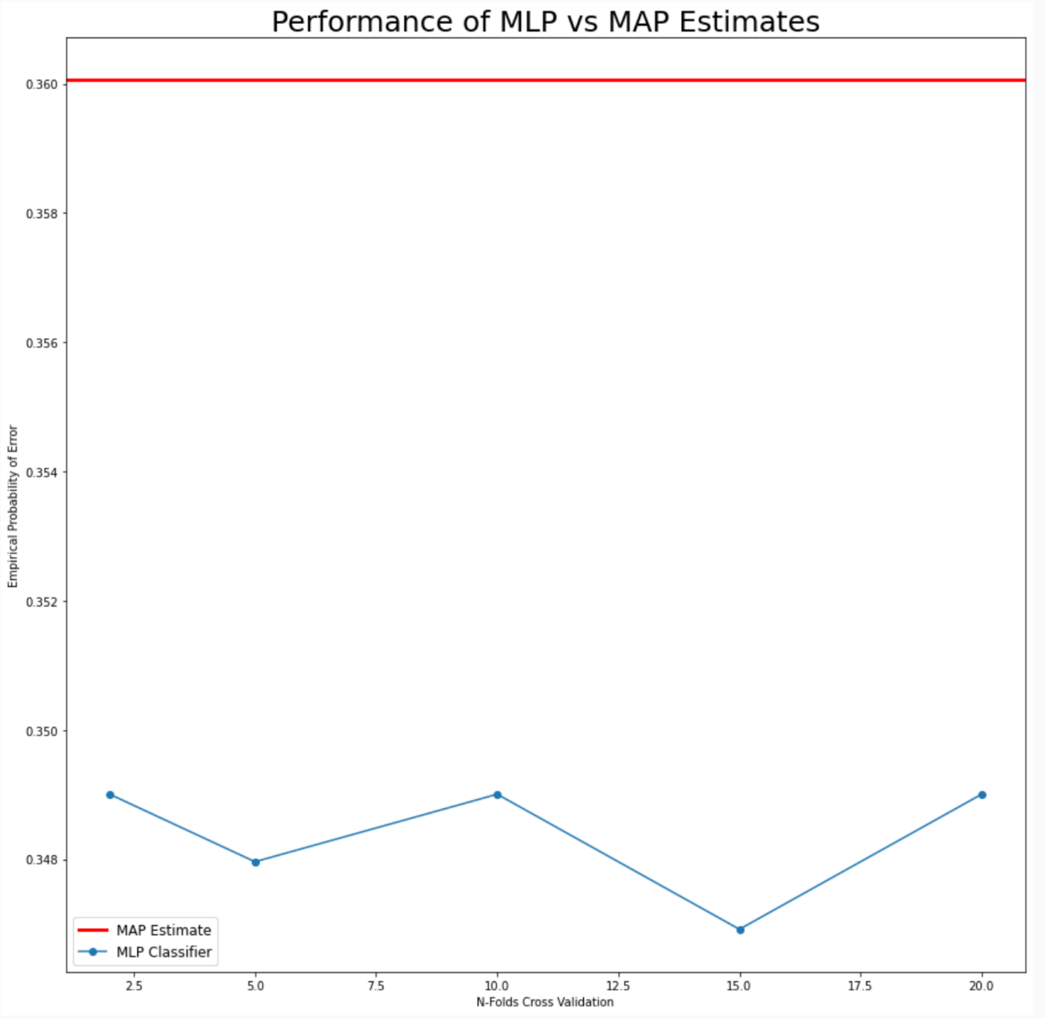
Northeastern University
College of Engineering

Results and Analysis

- MLP outperforms MAP estimate
  - This could be due to the shape of the data
- Our dataset is < 5000 samples

Future Work

- Better understanding of the data set
- This dataset only contains games from after the year 2000
  - In the future we could include a large set NFL games



Github Repository

All code and data used in this project can be found here:

https://github.com/nishanthmarer/ITMLPRFinalProjectAlex_Nishanth

Northeastern University
College of Engineering