Middle Tennessee State University
DATA 6990 (Fall 2023)

Factors Influencing Increased College Enrollment

Robert Conner
rc5g@mtmail.mtsu.edu

Abstract

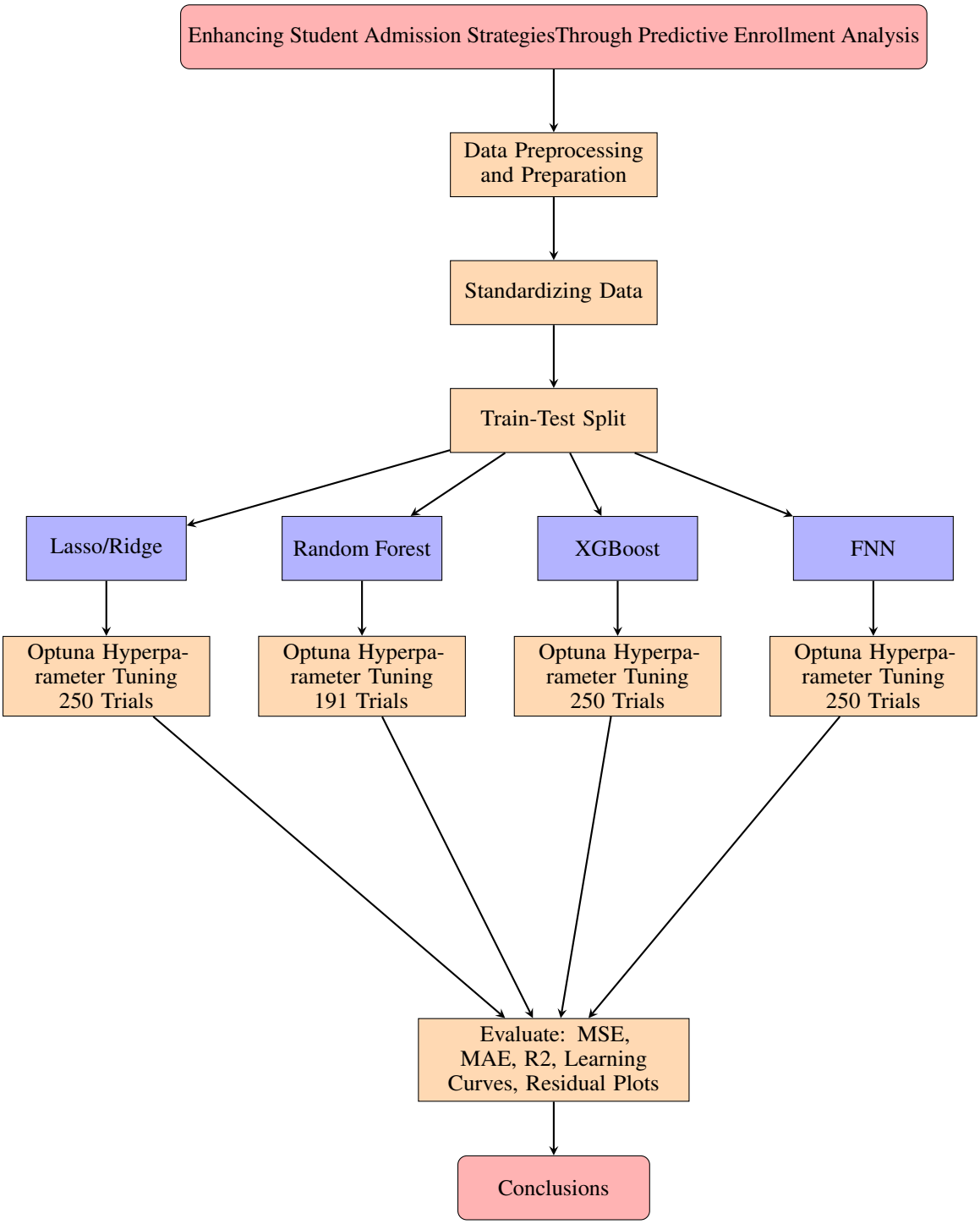
This study addresses the critical challenge of predicting college enrollments, a key concern for universities in planning and resource allocation. Utilizing a dataset of 1,534 colleges in the United States, we employ a suite of advanced machine learning algorithms - LASSO/Ridge Regression, Random Forest, Gradient Boosting, and Feed Forward Neural Network - to explore the relationships between enrollment numbers and various institutional features. Our methodology involves comprehensive data preprocessing, including one-hot encoding of categorical variables, imputation of missing data, and feature engineering. We then optimize each model using Optuna hyperparameter tuning and assess performance using metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and R^2. The study reveals that 'Total Enrollment' and 'Carnegie Classification' are significant predictors of new enrollments, with geographical factors also playing a crucial role. Financial features appear to not have much impact on these predictions. The ExtremeGradientBoosting (XGB) model emerges as the most effective, achieving the highest R^2 score of 0.96. This research not only provides insights for enhanced admission strategies through accurate enrollment forecasting but also contributes to the broader understanding of educational data dynamics.

1 Introduction

In the realm of higher education, universities are consistently confronted with the complex challenge of managing and predicting student enrollments. These enrollment figures play a pivotal role in guiding various planning and administrative decisions, ranging from academic programming to housing and campus resource allocation. Accurate prediction of enrollment numbers is therefore not only beneficial but essential for effective university administration and planning. This research paper aims to delve into the analysis of enrollment data from 1,534 colleges across the United States. Utilizing a range of machine learning algorithms, including LASSO/Ridge Regression, Random Forest, Gradient Boosting, and Feed Forward Neural Network, the study seeks to examine the relationship between enrollment figures and various institutional characteristics. By doing so, the research aims to uncover patterns and insights that could aid in the enhancement of enrollment prediction methodologies, ultimately contributing to more informed decision-making processes in higher education institutions.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

2 Methodology



2.1 Model Specification

1. **Ridge Regression**: It addresses multicollinearity in linear regression models by introducing a penalty term (L2 regularization). The model optimizes:

$$\text{SSE}_{L_2} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{P} \beta_j^2.$$

Figure 1: Ridge Regression (Hoerl 1970)

Strengths:

- Effective in handling multicollinearity.
- Robust against overfitting due to its regularization term.

Weaknesses:

- Can be sensitive to outliers.
- Not suitable for feature selection as it includes all features in the final model.

2. **Lasso Regression**: Similar to Ridge, but uses L1 regularization, which can shrink some coefficients to zero, effectively performing feature selection.

$$\text{SSE}_{L_1} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{P} |\beta_j|.$$

Figure 2: Lasso Regression (Tibshirani 1996)

Strengths:

- Capable of reducing the number of features by setting some coefficients to zero.
- Helpful in model interpretation and avoiding overfitting.

Weaknesses:

- Can struggle with multicollinearity compared to Ridge.
- Sensitive to outliers.

3. **Random Forest**: An ensemble learning method that operates by constructing multiple decision trees at training time and outputting the mean prediction (for regression) or the majority vote (for classification) of the individual trees.

```
Select the number of models to build, m
for i = 1 to m do
    Generate a bootstrap sample of the original data
    Train a tree model on this sample
    for each split do
        Randomly select k (< P) of the original predictors
        Select the best predictor among the k predictors and
        partition the data
    end
    Use typical tree model stopping criteria to determine when a
    tree is complete (but do not prune)
end
```

Figure 3: Basic Random Forest (Breiman 2001)

Strengths:

- Effective for both classification and regression tasks.
- Robust to outliers and non-linear data.

- Provides feature importance scores.

Weaknesses:

- Can be computationally expensive.
- Less interpretable compared to simple decision trees.

4. **XGBoost**: Extreme Gradient Boosting is an advanced implementation of gradient boosting algorithms, known for its efficiency and effectiveness in both classification and regression tasks. It builds upon the concept of boosting, where decision trees are sequentially added to an ensemble, each correcting the errors of its predecessors, optimizing predictive accuracy through iterative refinements. XGBoost is unique for its capacity to handle various types of differentiable loss functions and employing gradient descent for optimization (Chen et al., 2016)

   Strengths:

   - High performance and fast execution speed.
   - Handles a variety of data types, missing data, and various objective functions.
   - Provides regularization to avoid overfitting, improving model robustness.

   Weaknesses:

   - Can be prone to overfitting if not properly tuned or regularized.
   - Requires setting a number of hyperparameters, which can be complex.
   - Less interpretable compared to simpler models due to its complexity.

5. **Feedforward Neural Network (FNN)**: A type of deep learning model that is structured with an input layer, multiple hidden layers, and an output layer. In this study, our FNN architecture is designed for regression tasks and is tailored through hyperparameter tuning using Optuna. The network consists of:

   - An input layer that matches the dimensionality of the feature space.
   - A variable number of hidden layers (ranging from 1 to 3 in our setup) with units in each layer ranging from 32 to 128. These layers use activation functions such as ReLU, tanh, or sigmoid, as determined by the hyperparameter tuning process.
   - Dropout layers with rates between 0.2 and 0.5 to prevent overfitting.
   - An output layer with a single unit for regression output.

   The network is compiled with optimizers like Adam, SGD, or RMSprop, again selected through the hyperparameter tuning process. The loss function used is Mean Squared Error (MSE).

   Strengths:

   - Flexibility to model complex non-linear relationships in data.
   - Customizable architecture allows for tuning to specific data characteristics.
   - Capable of learning high-level abstractions from data through its deep structure.

   Weaknesses:

   - Requires a large amount of data to train effectively.
   - Prone to overfitting, although this is mitigated by dropout layers and hyperparameter tuning.
   - The "black box" nature of deep learning models can make them less interpretable.

2.2 Data Collection/Preprocessing

In this study we use a dataset provided to us, which includes reported institutional features for admissions in colleges across the country. The dataset is 1,534 rows, each representing a college in the United States. The dataset initially contained 108 features. After preprocessing the dataset was of size: 1,377 colleges and 82 feature columns.

Preprocessing steps taken:

- **Combining Endowment Asset Columns**: Merging columns for endowment assets of public (GASB) and private (FASB) institutions into a single metric, 'Endowment assets (year end) per FTE enrollment', for a consistent analysis.
- **Transformation of Categorical Variables**: One-hot encoding of categorical variables like geographic region, control of the institution, etc., and dropping the first category in each feature to avoid multicollinearity.
- **Computation of Enrollment Percentages**: Calculating new features for percentages of undergraduate and graduate enrollments, and recalculating the percent admitted to correct zero and NaN values.
- **Removal of Irrelevant Features**: Eliminating features with excessive missing values or irrelevance to the study. Identifying potential targets, with attention to preventing data leakage.
- **Imputation of Missing Data**: Employing an XGBoost regressor for predictable imputation and median imputation for columns unsuitable for XGBoost.
- **Further Feature Engineering**: Developing additional normalized features, including percentages of various enrollment types, to streamline the dataset for modeling.
- **Data Normalization**: Implementing StandardScaler for linear regression and FNN models, while tree-based models used the original dataset.

2.2.1 Model Optimization

1. **Linear Regression Models (Ridge and Lasso)**: Ridge and Lasso regression models were tuned using Optuna, with hyperparameters like model_type, alpha, fit_intercept, and tol. The best-performing model type and parameters were selected based on the Optuna study. Evaluation metrics included MSE, MAE, and R^2, and learning curves were plotted to assess model performance.
2. **Random Forest Model**: A Random Forest model was tuned using Optuna with Hyperband pruning. The hyperparameter space included n_estimators, max_depth, min_samples_split, min_samples_leaf, and max_features. The best model was evaluated using MSE, MAE, and R^2, and feature importances were plotted.
3. **XGBoost Model**: The XGBoost model was similarly tuned using Optuna with Hyperband pruning, optimizing parameters like n_estimators, max_depth, learning_rate, subsample, colsample_bytree, gamma, reg_alpha, and reg_lambda. Evaluation metrics and feature importances were analyzed.
4. **Feedforward Neural Network (FNN)**: The FNN model underwent hyperparameter tuning with Optuna, focusing on the number of layers, activation functions, and optimizer types. An EarlyStopping callback was utilized to avoid overfitting. The final model was evaluated on metrics like MSE, MAE, and R2, and both training history and actual vs. predicted values plots were generated.

2.3 Model Evaluation

In assessing the performance of our models, we adopt an approach that not only considers traditional regression metrics, but also incorporates additional analyses to ensure robustness and reliability of our predictions. Our evaluation strategy is as follows:

1. **Primary Performance Metrics**: The core metrics for evaluating model performance include:
    - *Mean Squared Error (MSE)*: Measures the average of the squares of the errors, i.e., the average squared difference between the estimated values and the actual value.
    - *Mean Absolute Error (MAE)*: Represents the average of the absolute differences between predicted values and observed values.
    - *Coefficient of Determination (R^2)*: Indicates the proportion of the variance in the dependent variable that is predictable from the independent variables.

2. **Cross-Validation**: We employ K-Fold cross-validation (10 splits), particularly useful in assessing the effectiveness of the models in handling unseen data and reducing the potential for overfitting.
3. **Learning Curves**: By plotting learning curves, we assess how well the model learns as the size of the training set increases. This helps in understanding the model's behavior with respect to its capacity to generalize from the training data to unseen data.
4. **Feature Importance Analysis**: For tree-based models like Random Forest, and XGBoost, we evaluate the feature importance to understand which features contribute most to the predictive power of the model. This is crucial for interpretability and for understanding the dynamics of the educational data.
5. **Actual vs. Predicted Values Plot**: Visualization of actual versus predicted values provides a direct comparison of the model's predictions with the actual data, offering an intuitive assessment of model accuracy.
6. **Optuna Hyperparameter Tuning**: We use Optuna for optimizing the hyperparameters of our models. The best hyperparameters are selected based on the lowest MSE achieved during the tuning process, ensuring that the models are well-optimized for the task.
7. **Early Stopping Mechanism**: For models like the Feedforward Neural Network, an early stopping mechanism is employed to cease training when the validation loss ceases to decrease. This approach aids in preventing overfitting and enhances the model's ability to generalize.
8. **Handling of Missing Data**: Imputation strategies, including XGBoost-based imputation for certain columns and median imputation for others, are employed to handle missing data, which is a common challenge in educational datasets.

3 Experiments and Results

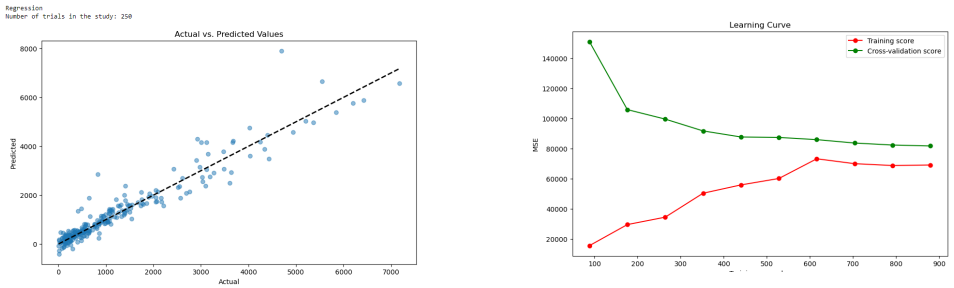


Figure 4: Residuals and Learning Curve: LASSO

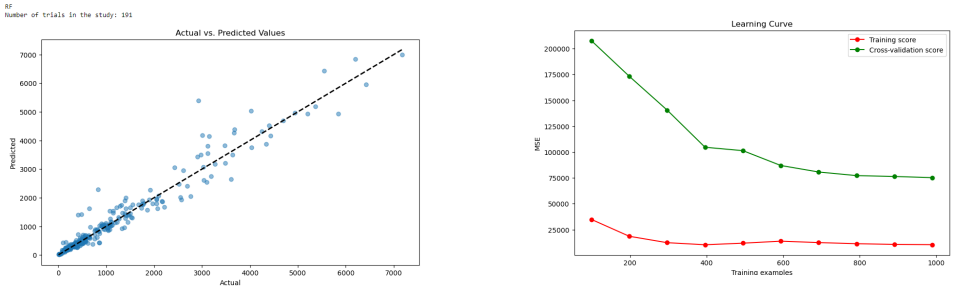


Figure 5: Residuals and Learning Curve: RandomForest

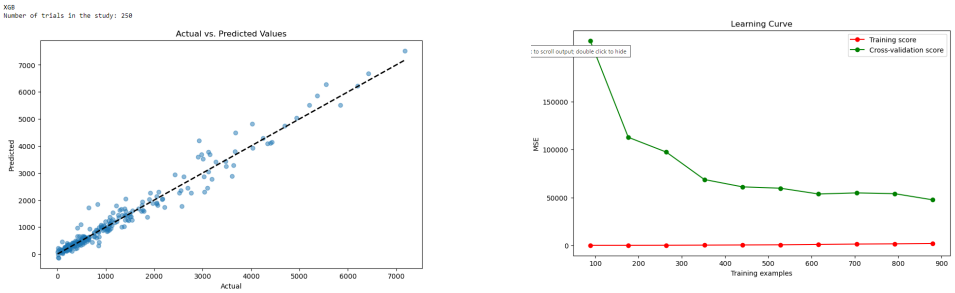


Figure 6: Residuals and Learning Curve: ExtremeGradientBoosting

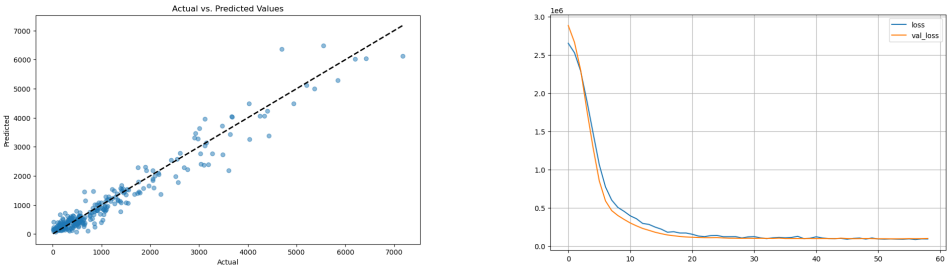


Figure 7: Residuals and Learning Curve: RandomForest

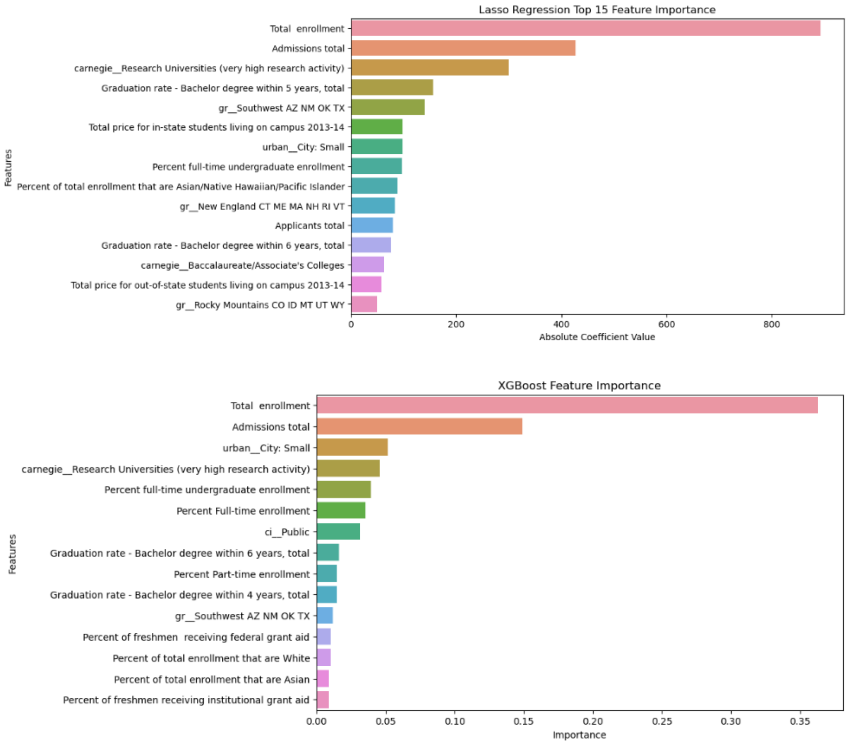


Figure 8: Feature Importance: LASSO and XGB

	model	mse	mae	r2
0	FNN	97398.09	213.03	0.94
0	XGB	62783.47	153.72	0.96
0	RF	101864.72	172.33	0.94
0	LASSO	149954.89	223.15	0.91

Figure 9: Final Modeling Results

4 Discussion of Results

4.1 Model 1: LASSO

The LASSO regression model shows good results from the training and testing of the college admissions data. The residuals plot shows that the large majority of predictions are close to the line denoting perfect predictions. The learning curve shows that the validation set begins with a much higher MSE than the training set, but this gap decreases significantly. The small gap between the validation and training scores shows that the model, while possibly slightly overfit, still generalizes well to the dataset. The feature importance ranking from the LASSO model shows that the most influential feature was 'Total Enrollment', which is expected as the size of the school would greatly influence the number of enrollments they can achieve. Interestingly, the 'Carnegie Classification' of a 'very high research activity' school was also one of the more influential features.

The following hyperparameters were chosen for the model: 'alpha': 0.74, 'fit_intercept':True, 'tol':1.19e-05

This model achieved the following metrics: MSE - 149,955; MAE - 223.15; R2 - 0.91. The scores achieved show that the model performs well and is able to generalize to the unseen test data.

4.2 Model 2: RandomForest

The Random Forest model shows slightly better results than the LASSO regression model. The residuals are very similiar to the previous model and show that the predictions are close to the perfect prediction line. The learning curves show that this model is most likely more overfit to the training data than the previous model as well. The gap between the training and cross-validation scores is larger, but not overly extreme.

The following hyperparameters were chosen for the model: 'n_estimators': 945, 'max_depth': 16, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': None

This model achieved a MSE of 101,865, MAE of 172.33, and a R2 of 0.94. These scores show that the model, while possibly more overfit than the LASSO model, still performs better on the test set. This model achieved a lower MSE and MAE, and increased the R2 score by 3 percent.

4.3 ExtremeGradientBoosting (XGB)

The XGB model performed better than both the LASSO and RandomForest models. The residuals for this model are similiar to the previous models and shows no pattern among the residuals besides falling near the perfect prediction line. The learning curves for this model show that it is also less overfit than the RF model. The curves display that the cross-validation scores, while beginning with a much higher MSE value, gradually come close to the scores from the training set. The training set scores stay very close to 0 (in relation to the cross-validation MSE) for the entire training process.

The feature importance rankings produced by the XGB model shows similarities with the LASSO model feature importance rankings. The 'Total Enrollment' is still the most influential and we can see that the 'Carnegie Classification' of 'very high research activity' is also still one of the top features.

The feature 'urban_City: Small' is prominently shown as rank 3, but none of the other urbanization categories appear in the rankings.

The following hyperparameters were chosen for the model: 'n_estimators': 926, 'max_depth': 3, 'learning_rate': 0.04, 'subsample': 0.5, 'colsample_bytree': 0.98, 'gamma': 0.58, 'reg_alpha': 5.12, 'reg_lambda': 0.68

This model achieved the metrics of the lowest MSE and MAE values, 62,783 and 153.72 respectively. This model also achieved the highest R2 score of 0.96. This shows that the model is able to accurately predict the new enrollment value based on this dataset.

4.4 FeedFowardNeuralNetwork (FNN)

The final model used was the FNN model. This model performed well on the dataset, resulting in a residual plot that was largely similiar to the previous models. There appears to be no pattern to the residuals and they stick close to the perfect prediction line. The learning curve for this model is different than the previous learning curves. Due to differences in how the models were created, this last learning curve was produced differently. The values on the y-axis show the loss from the training as a normalized value (not raw MSE such as in the previous learning curves). The X-axis denotes the number of epochs. From this plot, we can see that the 'loss' (training MSE) and 'val_loss' (validation MSE) begin apart but quickly converge and stabilize for the duration of training. This lack of a gap between the lines shows that the model does a great job generalizing and is likely not overfit to the training data.

The following hyperparameters were chosen for the model: 'n_layers': 2, 'activation': relu, 'optimizer': rmsprop, 'units_layer_0': 94, 'units_layer_1': 121, 'dropout_layer_1': 0.42

The final resulting metrics from this model showed it achieved a MSE of 97,398, a MAE of 213.03, and a R2 of 0.94. These results are comparable to the previous models.

5 Conclusion

In this study, we explored predictive modeling in the context of college admissions through the creation of various machine learning models (LASSO/Ridge Regression, RandomForest, ExtremeGradientBoosting, and FeedForwardNeuralNetwork). Each model presented their own strengths and weaknesses. Overall, all the created models demonstrated good predictive abilities with R2 scores over 91% for all the models. The RandomForest model and the LASSO model both performed well albeit with slight overfitting. The FNN model showed similiar results as the RandomForest model, achieving a R2 of 94%. The XGB model outperformed all the previous models, achieving the highest R2 of 96%, indicating superior predictive accuracy. The LASSO models strength lies in its simplicity and ability to generalize while capturing linear patterns. The RandomForest and XGB tree-based models offer more complex but slightly overfitted predictions. FNN, on the other hand, strikes a balance between complexity and generalization.

In terms of Mean Squared Error (MSE) and Mean Absolute Error (MAE), the XGB model showed the lowest values, indicating its sophisicated predictive capabilities. The learning curves of the various models suggested varying degrees of overfitting, with RandomForest and XGB displaying more signs than LASSO and FNN.

Across the models, 'Total Enrollment' emerged as a consistently influential feature, underscoring its significant role in prediciting new enrollments. This is expected, as discussed, due to the size of the school influencing the number of new enrollments. Similiarly, the feature 'Admission total' is consistently in the top of the feature importance rankings. This is another expected result, since the number of admissions greatly influences the final number of new enrollments to the school. The 'Carnegie Classification' showing that high research activity greatly influenced the prediction process was interesting. This feature was the only Carnegie Classification that appears in the top 10 for both of the feature importance rankings. I assume this is due to a college that conducts a large amount of research may also encourage greater enrollment, or symbolize greater opportunities for these students, thus resulting in higher enrollments. Lastly, the geographical region is another prominent feature in the importance rankings. The geographical location of the US Southwest appears higher in the rankings, followed by the Northeast. I find this interesting since trends show that population centers

across the Southwest have grown substantially since 2010 and the Northeast is still a very populated region.

Considering the very high predictive abilities shown by these models in predicting the new enrollments, we can assume that these models could be used to estimate future enrollments. These models offer valuable insights for college admissions strategies, enabling more accurate forecasting of enrollment numbers. These models can aid in resource allocation, marketing strategies, and policy-making by providing data-driven predictions of student enrollment. Future research could explore the feature rankings and attempt to identify the reasons that these features are influential. Additional predictive features can be added to this dataset, potentially providing further insights.

References

[1] Chen, T., & Guestrin, C. (2016, August). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). ACM. https://doi.org/10.1145/2939672.2939785

[2] Hoerl, A. E., & Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics, 12(1), 55–67. https://doi.org/10.2307/1267351

[3] Kuhn, M., & Johnson, K. (2013). Applied Predictive Modeling. Springer New York, NY. https://doi.org/10.1007/978-1-4614-6849-3

[4] Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society. Series B (Methodological), 58(1), 267–288. http://www.jstor.org/stable/2346178