Holly Cohan,Yesh Onipede, Yu Xia

Week 9 Report: Model Evaluation & Selection of Winning Model

The objective of this week's assignment is to evaluate all of the models we have attempted, compare results, and select a winning model. We made our selection with heavy consideration of the bias-variance tradeoff, which represents the balance between the two main types of errors associated with a machine learning model's accuracy and ability to generalize: bias and variance.

Bias refers to errors from assumptions that simplify the model too much, leading to underfitting, where the model cannot capture the data's true patterns. Variance refers to errors from the model's sensitivity to small fluctuations in the training set, often resulting in overfitting, where the model fits noise rather than general patterns. Increasing model complexity reduces bias but can increase variance, so the goal is to find an optimal complexity that minimizes both, allowing the model to generalize well on unseen data.

Model Summary

The first four variations of linear regression models listed in Appendix 1.A were built in week 6.

1. The baseline OLS regression model performed poorly (see Appendix 1.A), with extremely high error metrics (MSE and RMSE) and a negative R^2 on the validation set, indicating that the model does not explain the variance in the data.

2. The OLS model using PCA components demonstrates a significantly improved performance (see Appendix 1.A) based on the significantly lower MSE and RMSE and the reasonable R^2 of 0.245. While this is a reasonable improvement, the RMSE of 89.49 still highlights a significant error.

3. The Lasso regression models with varying regularization strengths (alpha) leveraged 5-fold cross validation to determine the optimal alpha value. The L1 penalty removed some irrelevant features from the model and performed similarly to the OLS model using PCA components.

4. The Ridge regression model also explored various regularization strengths to identify the optimal alpha. Ridge regression adds an L2 penalty to mitigate overfitting and improve generalization by stabilizing the coefficients, providing similar results to the Lasso regression model.

The following three Decision Tree models listed in Appendix 1.A were built in week 7.

5. The basic Decision Tree Regression model achieved near-perfect metrics on the training set, but performed relatively poorly on the validation set, indicating signs of severe overfitting and a failure to generalize well to unseen data.

6. To improve the Decision Tree model's generalization,we performed hyperparameter tuning using a grid search with 2- fold cross-validation and the following three hyperparameters:

    a. Maximum depth: controls how deep the tree is allowed to grow, preventing overfitting

b. Minimum samples split: minimum number of samples required to split a node

c. Minimum samples of leaf: minimum number of samples required at a leaf node

The tuned Decision Tree model showed improvement with its prediction on unseen data, indicating better generalization and reduced overfitting. However, the performance difference between the training and validation metrics suggests that this model is still overfitting slightly.

7. We introduced the pruned Decision Tree model by controlling for the ccp_alpha parameter to control the tree's complexity and reduce overfitting. The pruned model demonstrates significantly improved validation performance while still achieving high training accuracy. Though the R^2 value for this model is the same for the model that did not undergo pruning, the reduced MSE and RMSE values on the validation sets indicate improvement and an overall well-generalized model.

The following three Random Forest models listed in Appendix 1.A were built in week 8.

8. Random Forest 1 used 100 estimators, has a maximum depth of 15, and has no maximum number of features set. This model achieved a strong balance between predictive accuracy and computational efficiency, with relatively low MSE and RMSE values on both training and validation sets.

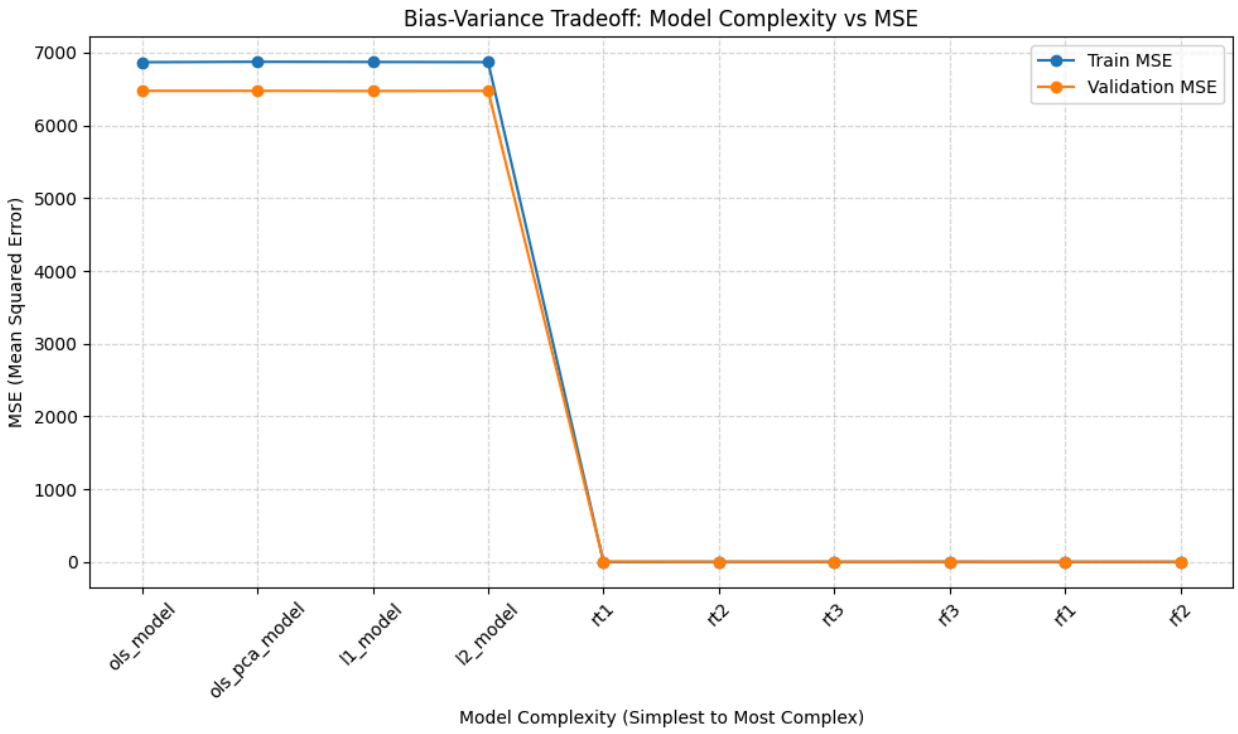
9. Random Forest 2 used 200 estimators, has a maximum depth of 15, and has no maximum number of features set. The increased number of estimators enhanced model accuracy by fully utilizing the available information and allowed the model to capture more complex patterns, though it slightly increased variance. It also showed signs of slight overfitting, with higher accuracy on the training set than the validation set.

10. Random Forest 3 used 200 estimators, has a maximum depth of 10, and has no maximum number of features set. This model sought to find a middle ground and provided a tradeoff between variance and bias, with reasonably low MSE and RMSE on both the training and validation sets. The reduced depth helped control overfitting, making this model effective for capturing key data patterns without high risk of overfitting, enhancing its generalizability.

Winning Model Selection

The metric we used to select the winning model is MSE for our regression model because it penalizes large errors, making it easier to compare models, especially those with similar performance.

Chart Depicting the Bias-Variance Tradeoff



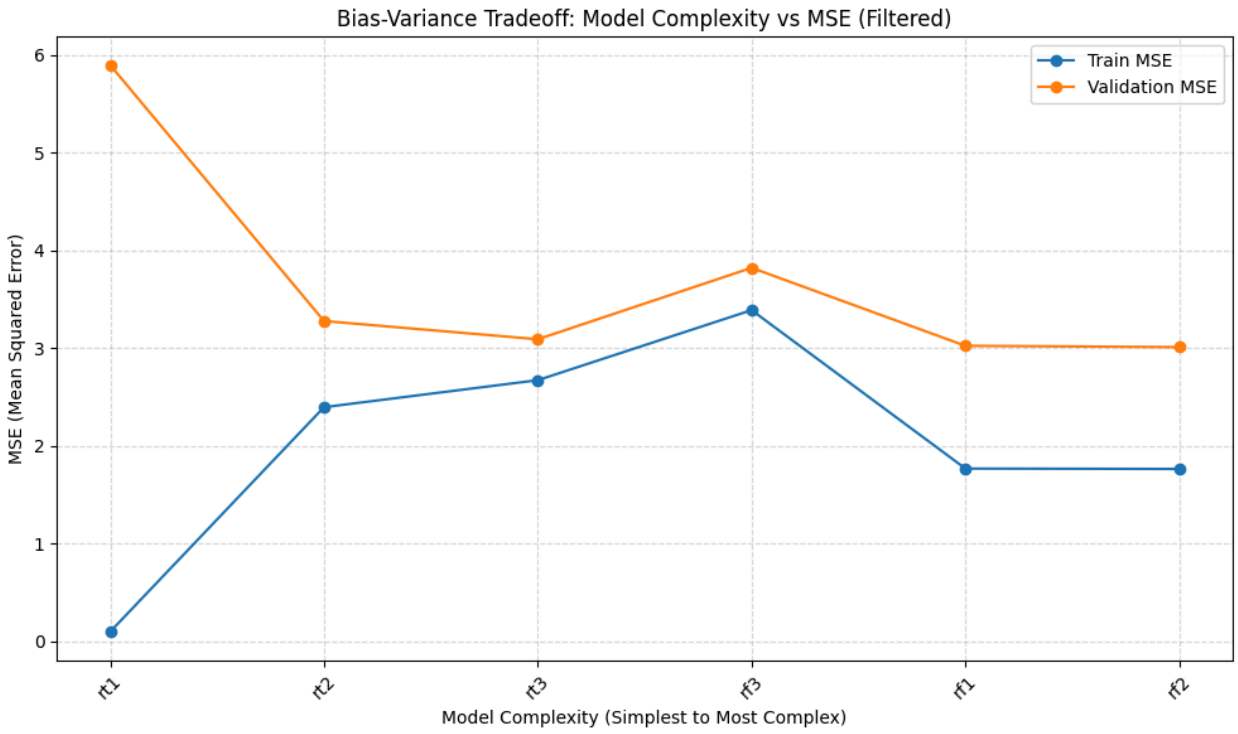
The chart above provides a clear visual summary of the bias-variance tradeoff as it displays the Mean Squared Error (MSE) for each model on both the training and validations sets, ordered from simplest to most complex.

High Bias in Simple Models

The simplest models, Ordinary Linear Regression and Linear Regression with PCA, show high MSE values on both the training and validation sets, especially in the validation MSE. This indicates high bias. These models fail to capture the underlying patterns in the training data, which resulted in them having poor performance and underfitting the data as indicated by the high training MSEs. These simpler models have similar MSE values across training and validation sets, but both are high, reflecting their inability to fit the data accurately.

Increasing Complexity and Reduced Bias

As the models become more complex, the training MSE drops significantly. This reduction in the training errors shows that the models are becoming more flexible and capturing more of the patterns in the training data. In order to further evaluate which model performs the best among these three, we zoom into the tree-based methods in the chart.



The pruned Decision Tree, Random Forest 2, and Random Forest 3 (referred to as 7, 9, and 10 above, respectively) emerged as our strongest models. As shown in the graph, Random Forest 2 has the lowest training MSE, showing a strong fit to the training data, but its larger gap in validation MSE suggests some overfitting and higher variance. Random Forest 3 has the highest MSE values of the three but maintains a close alignment between training and validation errors, suggesting that it generalizes well while potentially underfitting slightly. The pruned Decision Tree shows a smaller gap between training and validation MSE, indicating a good balance with lower variance and a solid generalization capability. Due to the strong balance between complexity and generalization that it achieved, the pruned Decision Tree is our best model.

What Does this Tell Us About Our Models and Prediction Task?

Simple models are ineffective for this task

The high MSE for Ordinary Linear Regression and Linear Regression with PCA on both the training and validation sets indicates that these simple models cannot capture the underlying patterns in our data. This suggests that our prediction task is complex, likely involving non-linear relationships that these basic linear models fail to capture. These models are underfitting the data, which points to the need for greater model flexibility in order to improve prediction accuracy.

Overfitting with Very Complex Models

We observed that overfitting becomes more pronounced as model complexity increases, particularly with random forest models. While random forests are more capable of capturing patterns in the training data compared to decision trees, they fail to generalize well to the validation dataset for this task, indicating

that they may be learning noise from the training set. As we observed that random forests can easily overfit to our dataset, it is reasonable to suspect that more complex models, such as those leveraging boosting techniques and neural networks, are even more susceptible to overfitting. For our prediction task, which requires models to generalize across varying conditions, these overly complex models are not suitable as they struggle to adapt to unseen data.

Optimal Performance with Moderate Complexity
Models like the Pruned Decision Tree and Random Forests (especially with hyperparameter tuning) achieve low MSE on both the training and validation sets, indicating that they effectively capture patterns without overfitting. This balance suggests that our prediction task benefits from models that are complex enough to capture non-linear relationships but controlled enough to avoid fitting to noise in the training data.

Performance Metrics for the Winning Model

Model Type	Metrics	Training Set Values	Validation Set Values	Testing Set Values
Pruned Decision Tree Model	**MSE**	2.6747	3.0939	**3.1037**
	RMSE	1.6354	1.7589	**1.7617**
	R^2	0.9997	0.9996	**0.9997**

Appendix 1.A Metrics Table

Model Type	Metrics	Training Set Values	Validation Set Values
Ordinary Linear Regression	**MSE**	6867.24	6474.64
	RMSE	82.87	80.47
	R^2	0.22	0.21
Ordinary Linear Regression with PCA	**MSE**	6874.08	6474.86
	RMSE	82.91	80.466520
	R^2	0.21	0.21
Linear Regression with Lasso (L1)	**MSE**	6871.35	6472.41
	RMSE	82.89	80.45
	R^2	0.21	0.21
Linear Regression with Ridge (L2)	**MSE**	6869.46	6474.39
	RMSE	82.88	80.46
	R^2	0.21	0.21
Initial Regression Tree Model (rt1)	**MSE**	0.1008	5.8987
	RMSE	0.3175	2.4287
	R^2	0.999988	0.9993
Regression Tree Model with Hyperparameter Tuning (rt2)	**MSE**	2.3140	3.3827
	RMSE	1.5212	1.8392
	R^2	0.9997	0.9996
Pruned Decision Tree Model (rt3)	**MSE**	2.6747	3.0939
	RMSE	1.6354	1.7589
	R^2	0.9997	0.9996
Random Forest 1(rf1)	**MSE**	1.7703	3.0264
{'n_estimators': 100,	**RMSE**	1.3305	1.7397

'max_depth': 15, 'max_features': None}	**R^2**	0.9998	0.9996
Random Forest 2 (rf2) **{'n_estimators': 200, 'max_depth': 15, 'max_features': None}**	**MSE**	1.7664	3.0131
	RMSE	1.3290	1.7358
	R^2	0.9998	0.9996
Random Forest 3 (rf3) **{n_estimators': 200, 'max_depth': 10, 'max_features': None}**	**MSE**	3.3915	3.8244
	RMSE	1.8416	1.9556
	R^2	0.9996	0.9995