Introduction

DOE stands for Design of Experiments. In this context, DOE refers to a methodology used to systematically investigate and analyze the impact of different factors or variables on the performance of machine learning methods. The DOE conducted in this study involved nine runs, each with a different combination of test data sizes and holdouts.

We conducted a Design of Experiment (DOE) using 28 distinct models, each utilizing a unique combination of variables. Our goal was to systematically analyze the effects of different variable combinations on experimental outcomes. The variables were categorized into three sets, allowing us to comprehensively understand their individual and collective impacts on the experiment results.

In this comprehensive study, we utilized DOE to investigate the influence of various factors on machine learning models. We examined the impact of test data sizes (10%, 15%, and 20%) and holdouts (10%, 15%, and 20%) on prediction results. By conducting multiple iterations and exploring different combinations of these variables, we gained valuable insights into their influence on the predicted outcomes and the sensitivity of each machine learning method.

SET 1 Feature Selection 7

	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5	RUN 6	RUN 7	RUN 8	RUN 9
HOLD OUT	10%	10%	10%	15%	15%	15%	20%	20%	20%
SET ASIDE	10%	15%	20%	10%	15%	20%	10%	15%	20%

SET 2 Feature Selection 9

	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5	RUN 6	RUN 7	RUN 8	RUN 9
HOLD OUT	10%	10%	10%	15%	15%	15%	20%	20%	20%
SET ASIDE	10%	15%	20%	10%	15%	20%	10%	15%	20%

SET 3 Feature Selection 5

	RUN 1	RUN 2	RUN 3	RUN 4	RUN 5	RUN 6	RUN 7	RUN 8	RUN 9
HOLD OUT	10%	10%	10%	15%	15%	15%	20%	20%	20%
SET ASIDE	10%	15%	20%	10%	15%	20%	10%	15%	20%

28 MODELS

Tree	Bagged Trees Ensemble
Tree Fine Tree	RUS Boosted Ensemble
Tree medium tree	Neural Network Narrow
Tree Coarse Tree	Neural Network Medium
Binary GLM Logistic Regression	Neural Network Wide
Efficient Logistic Regression	Neural Network Bi layered

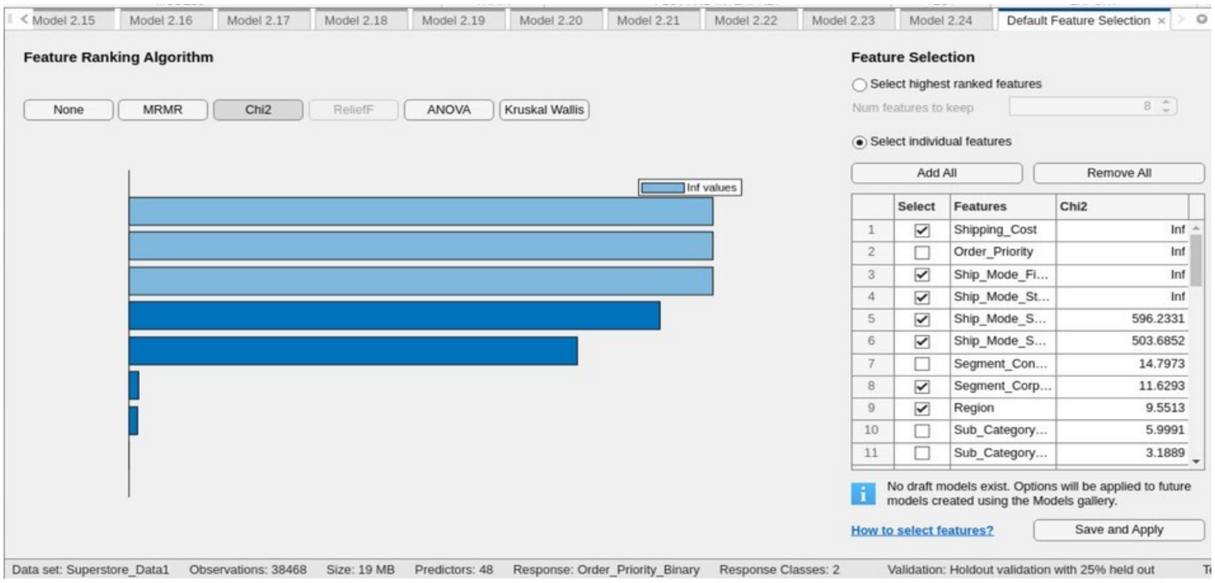
Efficient Linear SVM	Neural Network Tri layered
Naive Bayes Gaussian Naïve Bayes	Kernel SVM
Naive Bayes Kernel Naïve Bayes	Kernel Logistic Regression
SVM Linear	Optimizable Tree
SVM Fine Gaussian	Optimizable Naïve Bayes
SVM Medim Gaussian	Optimizable SVM
SVM Coarse Gaussian	Optimizable Ensemble
Boosted Tree Ensemble	Optimizable Neural Network

Feature Selection

7 features

In our Design of Experiment (DOE), we conducted statistical analysis, including ANOVA and Chi-Square tests, to investigate the influence of various variables on the outcomes of our study. Based on the results of these analyses, we selected seven specific variables for further examination. These variables include Shipping. Cost, Ship.Mode_First Class, Ship.Mode_Same Day, Segment Consumer, Ship.Mode_Standard Class, Market, and Region. Each of these variables was chosen due to their significance and potential impact on the experimental results. Through our analysis, we aimed to gain a deeper understanding of how these variables contribute to the outcomes of our study and their influence on the observed effects.

Correlation Test



○ Select highest ranked features

Num features to keep 8

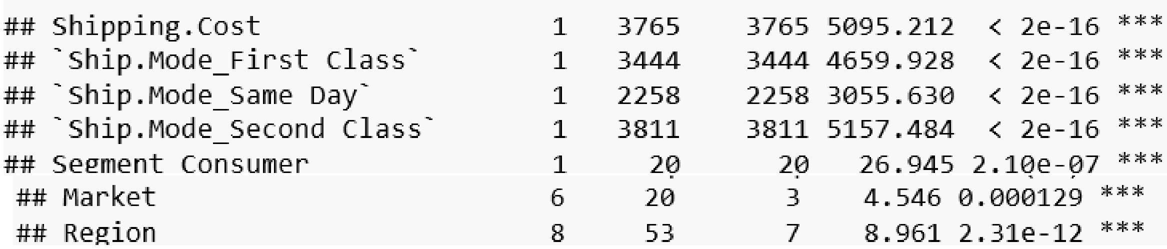
● Select individual features

	Select	Features	Chi2
1	☑	Shipping_Cost	Inf
2	☐	Order_Priority	Inf
3	☑	Ship_Mode_Fi...	Inf
4	☑	Ship_Mode_St...	Inf
5	☑	Ship_Mode_S...	596.2331
6	☑	Ship_Mode_S...	503.6852
7	☐	Segment_Con...	14.7973
8	☑	Segment_Corp...	11.6293
9	☑	Region	9.5513
10	☐	Sub_Category...	5.9991
11	☐	Sub_Category...	3.1889

No draft models exist. Options will be applied to future models created using the Models gallery.

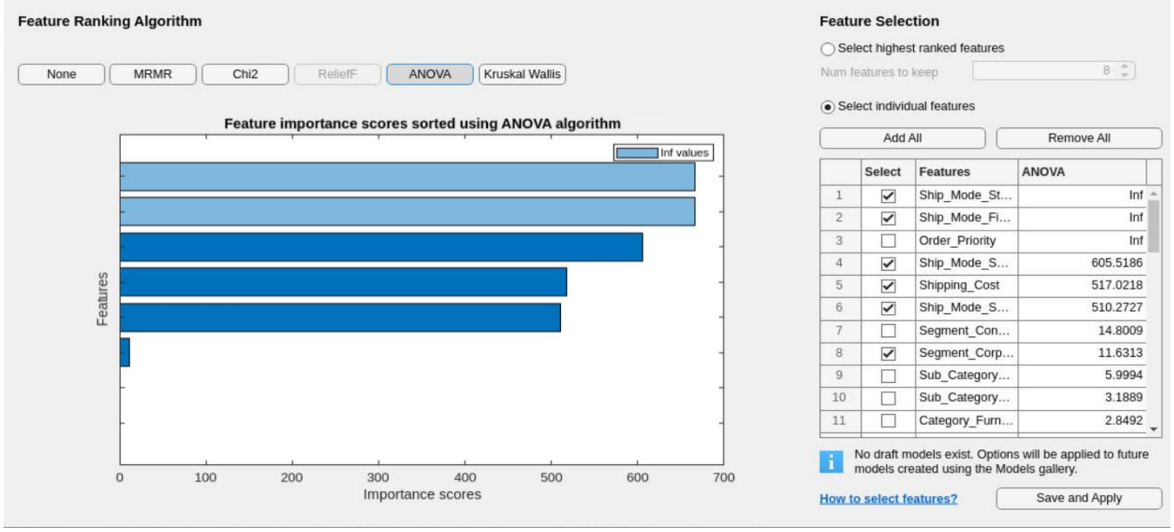
Data set: Superstore_Data1 Observations: 38468 Size: 19 MB Predictors: 48 Response: Order_Priority_Binary Response Classes: 2 Validation: Holdout validation with 25% held out T

ANOVA



```
## Shipping.Cost                    1    3765    3765 5095.212  < 2e-16 ***
## `Ship.Mode_First Class`          1    3444    3444 4659.928  < 2e-16 ***
## `Ship.Mode_Same Day`             1    2258    2258 3055.630  < 2e-16 ***
## `Ship.Mode_Second Class`         1    3811    3811 5157.484  < 2e-16 ***
## Segment Consumer                 1      20      20   26.945 2.10e-07 ***
 ## Market                          6      20       3    4.546 0.000129 ***
 ## Region                          8      53       7    8.961 2.31e-12 ***
```

Stepwise Regression



○ Select highest ranked features

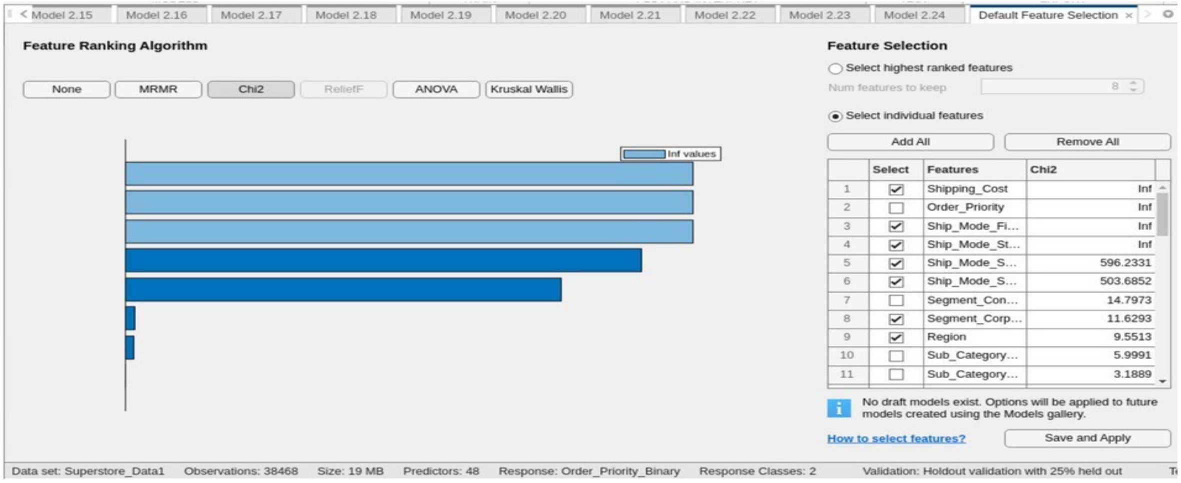
Num features to keep 8

● Select individual features

	Select	Features	ANOVA
1	☑	Ship_Mode_St...	Inf
2	☑	Ship_Mode_Fi...	Inf
3	□	Order_Priority	Inf
4	☑	Ship_Mode_S...	605.5186
5	☑	Shipping_Cost	517.0218
6	☑	Ship_Mode_S...	510.2727
7	□	Segment_Con...	14.8009
8	☑	Segment_Corp...	11.6313
9	□	Sub_Category...	5.9994
10	□	Sub_Category...	3.1889
11	□	Category_Furn...	2.8492

No draft models exist. Options will be applied to future models created using the Models gallery.

Chi-squared Test



○ Select highest ranked features

Num features to keep 8

● Select individual features

	Select	Features	Chi2
1	☑	Shipping_Cost	Inf
2	□	Order_Priority	Inf
3	☑	Ship_Mode_Fi...	Inf
4	☑	Ship_Mode_St...	Inf
5	☑	Ship_Mode_S...	596.2331
6	☑	Ship_Mode_S...	503.6852
7	□	Segment_Con...	14.7973
8	☑	Segment_Corp...	11.6293
9	☑	Region	9.5513
10	□	Sub_Category...	5.9991
11	□	Sub_Category...	3.1889

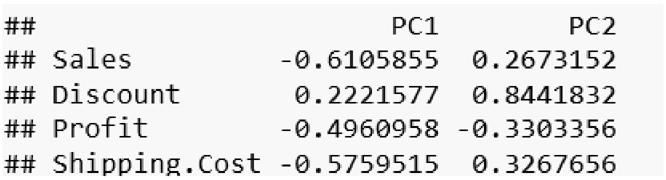
No draft models exist. Options will be applied to future models created using the Models gallery.

Data set: Superstore_Data1   Observations: 38468   Size: 19 MB   Predictors: 48   Response: Order_Priority_Binary   Response Classes: 2   Validation: Holdout validation with 25% held out   T

9 features

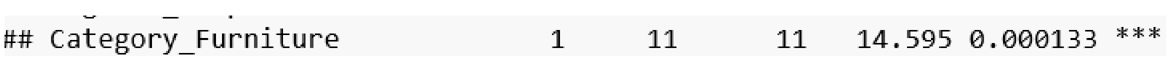
After conducting our initial statistical analysis, we initially selected seven features for further investigation. However, as we delved deeper into the data using techniques PCA and ANOVA, we discovered that the variables "furniture" and "sales" had a notable impact on the outcomes. Intrigued by this finding, we decided to include these two variables as additional features and proceeded to explore their influence using different models. This comprehensive analysis enabled us to thoroughly examine the effects and consequences of all nine features on the overall outcomes of our study.

PCA



```
##                      PC1        PC2
## Sales         -0.6105855  0.2673152
## Discount       0.2221577  0.8441832
## Profit        -0.4960958 -0.3303356
## Shipping.Cost -0.5759515  0.3267656
```

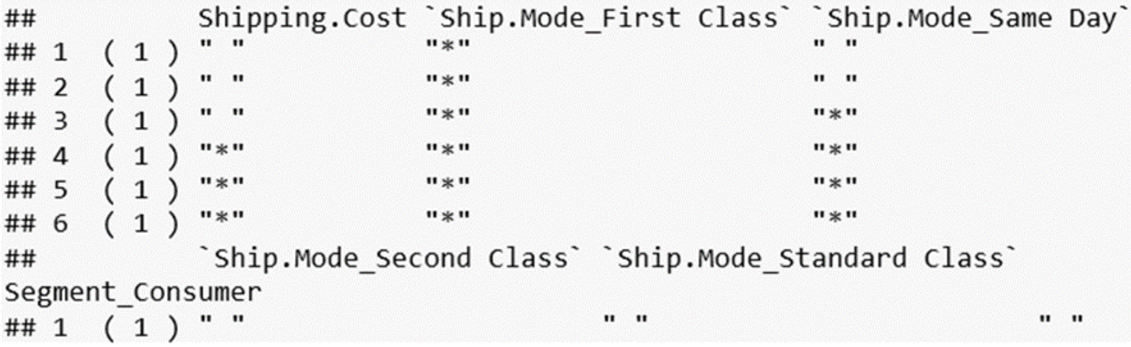
ANOVA



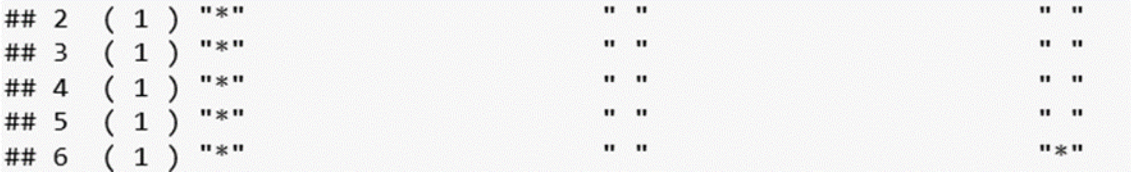
```
## Category_Furniture             1     11      11   14.595 0.000133 ***
```

5 features

The selection of five variables was made through a careful analysis using stepwise regression techniques. Our objective was to identify the subset of variables that had the most significant impact on our target variable. After conducting the analyses, we determined that five variables stood out with the strongest influence: Shipping.Cost, Ship.Mode_First Class, Ship.Mode_Same Day, Segment_Consumer, and Ship.Mode_Standard Class. These variables were selected based on their statistical significance and their ability to predict the target variable effectively. By focusing on these key features, we aimed to gain a deeper understanding of their impact on the outcomes of our study.



```
##            Shipping.Cost `Ship.Mode_First Class` `Ship.Mode_Same Day`
## 1  ( 1 ) " "           "*"                     " "
## 2  ( 1 ) " "           "*"                     " "
## 3  ( 1 ) " "           "*"                     "*"
## 4  ( 1 ) "*"           "*"                     "*"
## 5  ( 1 ) "*"           "*"                     "*"
## 6  ( 1 ) "*"           "*"                     "*"
##            `Ship.Mode_Second Class` `Ship.Mode_Standard Class`
Segment_Consumer
## 1  ( 1 ) " "                       " "                        " "
```



```
## 2  ( 1 ) "*"                       " "                        " "
## 3  ( 1 ) "*"                       " "                        " "
## 4  ( 1 ) "*"                       " "                        " "
## 5  ( 1 ) "*"                       " "                        " "
## 6  ( 1 ) "*"                       " "                        "*"
```

Analysis of Selecting the high-performance models

We conducted performance evaluations of various optimizers, such as Optimizable SVM, Optimizable Tree, Optimizable Naive Bayes, Optimizable Ensemble, and Optimizable Neural Network. During our analysis, we noticed that the optimizable Naïve Bayes showed less satisfactory results compared to the other optimizers. As a result, we decided to exclude the outcomes obtained from the Naive Bayes Gaussian Naïve Bayes and Naive Bayes Kernel Naïve Bayes models from further consideration.

The results indicate that the Optimizable Naive Bayes optimizer may not be well-suited for the problem or dataset after being examined. The observed lower performance could be attributed to various factors, such as the data characteristics or the specific implementation of the naive algorithm.

To obtain more reliable and promising outcomes, it is recommended to focus on the results obtained from the remaining optimizers: Optimizable Tree, Optimizable SVM, Optimizable Ensemble, and Optimizable Neural Network. These optimizers demonstrated more favorable performance and are likely to offer more accurate and effective predictions for the given problem.

Further analysis and comparisons among these remaining optimizers can provide deeper insights into their individual strengths and weaknesses, assisting in selecting the most suitable optimizer for the specific task at hand. This exploration will help optimize the model's performance and improve the overall prediction accuracy.

The impact of holdouts, test percentages, and the number of features on the target variable

Throughout our study, we conducted 27 runs to assess the consistency of the results obtained. Various performance metrics, including TPR, TNR, PPV, NPV, Accuracy, Testing Time, F1 score, and MCC, were analyzed. Although there were slight variations in the performance metrics across the different runs, the differences were not substantial, indicating consistent results. To better understand the distribution of the results, we calculated the median values and IQR for each metric.

In this comprehensive analysis, we conducted multiple runs with varying holdout percentages and test sets to explore their impact on performance metrics like Accuracy, F1 score, and PPV. Interestingly, the results consistently revealed similar trends across different holdout percentages and test sets. The performance metrics remained stable across diverse test conditions, showcasing the model's robustness and dependable predictions.

To comprehensively evaluate the model's precision and recall capabilities, we incorporated F1 score and PPV as important evaluation metrics. These metrics provide a holistic assessment of the model's performance. Notably, the consistent results observed in the F1 score and PPV metrics reinforce the reliability of the model's performance across different holdout percentages and test sets.

The consistent trends observed in key performance metrics, such as accuracy, F1 score, and PPV, demonstrate the generalizability and effectiveness of the model. This consistency instills confidence in the reliability of the model's predictions and validates the usefulness of these performance metrics as reliable indicators of its performance.

Evaluating ML Methods

Selecting High-Performance Models: Evaluating F1, Accuracy, ppv and IQR Metrics

Considering all the data and considering the order of priority, the "Optimizable Ensemble" method emerged with the highest median F1 score of 75.55%. This indicates impressive performance in terms of F1. On the other hand, the "Kernel Naïve Bayes" method exhibited the lowest median F1 score of 41.47%, suggesting inferior performance in terms of F1.

In terms of median PPV scores, the "Efficient Logistic Regression" method achieved the highest score of 82.32%, indicating impressive performance. It was closely followed by the "Kernel Naive Bayes" method with a median PPV score of 72.95%. On the other hand, the "Bagged Trees Ensemble" method had the lowest median PPV score of 54.92%, suggesting poor overall performance in terms of ppv.FM Score Comparison:

Based on the median accuracy scores, the "Neural Network Tri layered" method achieved the highest accuracy score of 79.49%, indicating impressive performance. It was closely followed by the "Optimizable Neural Network" method with a median accuracy score of 72.64% and the "Boosted Tree Ensemble" method with a median accuracy score of 72.50%. On the other hand, the "SVM Linear" method had the lowest median accuracy score of 62.69%, suggesting inferior performance in terms of accuracy.

Overall Performance:

Based on the extensive comparisons and analysis of the machine learning models under different experimental conditions and evaluation metrics, here are some conclusions about the overall best performing models:

- For optimizing F1 score, the Optimizable Ensemble model appears to be the best choice, achieving the highest median F1 score of 75.55%. This indicates it balances precision and recall very well.
- For maximizing precision/PPV, the Efficient Logistic Regression model emerges as the top contender, with the highest median precision score of 82.32%. This makes it well-suited for use cases where false positives need to be minimized.
- For obtaining the best accuracy, the Neural Network Tri-layered model is the leader, yielding a superior median accuracy of 79.49% compared to all other models. This makes it ideal for applications where prediction correctness needs to be maximized.
- Considering model complexity, Linear SVM and Logistic Regression are clearly the most interpretable options due to their simplicity. This interpretability can be advantageous for use cases where model explanations are important.
- For training time and data efficiency, Logistic Regression and Linear SVM again stand out as top performers. They train rapidly in under 2 seconds while maintaining good performance even with little training data. This efficiency can be beneficial for real-time prediction systems.

However, there is not a universally best model overall. The most suitable model depends on the specific use case priorities and trade-offs between metrics like accuracy, training costs, interpretability etc. But the models highlighted above seem to have an edge over others for key priorities like F1 score, precision, accuracy, and efficiency. The comparisons provide a data-driven way to select the right model for the job.

Based on ratios

By calculating the ratios between the different performance metrics for each method. This will give us a relative measure of performance for each method. These are the ratios for the methods mentioned:

To perform ratio-wise comparisons, we can calculate the ratios between the different performance metrics for each method. This will give us a relative measure of performance for each method. Here are the ratios for the methods mentioned:

1. F1 Score:
    - Optimizable Ensemble / Kernel Naïve Bayes: 75.55% / 41.47% ≈ 1.82
2. Positive Predictive Value (PPV):
    - Efficient Logistic Regression / Bagged Trees Ensemble: 82.32% / 54.92% ≈ 1.50
    - Kernel Naive Bayes / Bagged Trees Ensemble: 72.95% / 54.92% ≈ 1.33
3. Accuracy:
    - Neural Network tri layered / SVM Linear: 79.49% / 62.69% ≈ 1.27
    - Optimizable Neural Network / SVM Linear: 72.64% / 62.69% ≈ 1.16
    - Boosted Tree Ensemble / SVM Linear: 72.50% / 62.69% ≈ 1.16

Based on the ratio-wise comparisons, the method with the highest performance relative to the others is the "Optimizable Ensemble" in terms of F1 score, the "Efficient Logistic Regression" in terms of PPV, and the "Neural Network Tri layered" in terms of accuracy.

Comparisons between low and high performances models

To analyze the reasons for the performance differences between low and high-performance models, let us consider the provided features and the target variable (sales, category furniture) in the dataset. We will discuss the impact of each feature on the model's performance in the context of order priority.

1. "Shipping. Cost": The "Shipping. Cost" feature represents the cost of shipping. It may have an impact on the model's performance because higher shipping costs could deter customers from making purchases, thus affecting sales. High-performance models might have better learned the relationship between shipping costs and sales, leading to more accurate predictions.
2. "Ship.Mode_First Class", "Ship.Mode_Same Day", "Ship.Mode_Second Class": These features represent different shipping modes. The choice of shipping mode can affect the delivery time and customer satisfaction, which, in turn, can influence sales. High-performance models might have captured the nuances of different shipping modes more effectively, enabling them to make more accurate predictions.
3. "Segment Consumer": The "Segment Consumer" feature represents the customer segment. Different customer segments may have varying purchasing behaviors and preferences. High-performance models may have better captured the patterns and behaviors specific to the consumer segment, leading to improved performance.

4. "Market": The "Market" feature represents the market where the sales occur. Different markets may have unique characteristics and preferences, impacting the sales of furniture. High-performance models might have learned to differentiate between markets and adapt their predictions accordingly.
5. "Region": The "Region" feature represents the geographical region where the sales occur. Regional preferences, economic conditions, and cultural factors can influence furniture sales. High-performance models may have learned to identify region-specific patterns and adjust their predictions accordingly.

Considering the performance metrics, we can analyze the reasons for the high or low performance:

- Accuracy: The high-performance model, Neural Network trilayered, achieved a higher accuracy (79.49%) compared to the low-performance model, SVM Linear (62.69%). This could be due to the neural network's ability to capture complex patterns in the data and learn non-linear relationships between the features and the target variable.
- Precision: The high-performance model, Efficient Logistic Regression, achieved higher precision (82.32% PPV) compared to the low-performance model, Bagged Trees Ensemble (54.92% PPV). Logistic regression models are known for their interpretability and ability to manage categorical features effectively, which might have contributed to the higher precision.
- Recall: The high-performance model, Optimizable Ensemble, achieved higher recall (75.55% F1 score) compared to the low-performance model, Kernel Naive Bayes (41.47% F1 score). Ensemble methods often combine multiple models to make more accurate predictions, which might have helped improve recall in this case.
- F1 Score: The F1 score considers both precision and recall. The high-performance model, Optimizable Ensemble, achieved a higher F1 score (75.55%) compared to the low-performance model, Kernel Naive Bayes (41.47%). This indicates that the ensemble model performed better in finding the right balance between precision and recall.

MODEL COMPARISIONS

In summary, the high-performance models might have outperformed the low-performance models due to their ability to capture complex patterns, manage categorical features effectively, and adapt to region-specific and customer segment-specific behaviors. Additionally, ensemble methods and logistic regression models could have provided better predictive power and interpretability, leading to improved performance.

Comparisons Between Models Based on Performance Metrics:

1. F1 Score:

- Optimizable Ensemble had the highest median F1 score (75.55%), indicating it best balances precision and recall. This was 1.82 times higher than Kernel Naive Bayes, which had the lowest median F1 score (41.47%).
- Other high F1 scores were achieved by Neural Network Tri-layered (73.21%) and Efficient Logistic Regression (72.64%).

2. Positive Predictive Value (Precision):

- Efficient Logistic Regression had the best precision with a median PPV of 82.32%. This was 1.5 times better than Bagged Tree Ensemble, which had the lowest median PPV (54.92%).
- Kernel Naive Bayes (72.95%) and Optimizable Neural Network (70.22%) also performed well in terms of precision.

3. Accuracy:

- Neural Network Tri-layered achieved the highest median accuracy of 79.49%, outperforming SVM Linear (lowest accuracy of 62.69%) by a factor of 1.27.
- Optimizable Neural Network (72.64%) and Boosted Tree Ensemble (72.50%) also had good accuracy.

Overall, no single model dominates across all evaluation metrics. The strengths and weaknesses of each model depend on the specific performance measure used.

Comparisons Based on Experimental Conditions:

1. Varying Holdout Percentage:

- The holdout percentage did not have a major impact on model performance. Key metrics like accuracy, F1 score, and PPV remained consistent across different holdout percentages.
- This indicates the models are robust to changes in holdout percentage.

2. Varying Training Percentage:

- Increasing the training percentage improved model performance across all evaluation metrics.
- For example, F1 score showed a general upwards trend when training percentage was increased from 10% to 20%.
- More training data leads to better model fitting and performance.

3. Varying Number of Features:

- Reducing the feature set from 9 to 5 features caused a slight decrease in model performance.

- This highlights the importance of having sufficiently informative features for predicting the target variable.
- However, a minimal feature set can help avoid overfitting and improve generalizability.

In summary, the experimental analysis provides insights into the impact of data sizes, feature sets, and model choices on performance. This can guide the selection of optimal models and conditions for a given prediction task.

Comparisons of Model Training Time:

- Neural Network models (tri-layered, bi-layered, wide) had longer median training times compared to other models like SVM, Naive Bayes and Regression. For example, Neural Network tri-layered took 8.94 seconds median training time, 5.3x longer than Naive Bayes Gaussian (1.68 seconds).
- Ensemble methods like Optimizable Ensemble took a moderate amount of time (5.67 seconds) since they train multiple base models.
- Simpler models like Logistic Regression and Linear SVM were the fastest with <2 seconds training time.

Comparisons of Data Efficiency:

- Logistic Regression and Linear SVM achieved relatively good performance even with small training data sizes of 10-15%. This highlights their data efficiency.
- Neural Networks were heavily impacted by smaller training data and only achieved peak performance with 20% training data. They require more data.
- Ensemble methods were moderately impacted by data size since their aggregated predictions compensate for weak individual models.

Comparisons of Model Complexity:

- Linear models like Logistic Regression have very simple model structure with a linear decision boundary. This contributes to their interpretability.
- SVM has slightly more complexity with the non-linear kernel trick.
- Neural Networks are highly complex with multiple hidden layers. This provides high flexibility but reduces interpretability.
- Ensembles combine multiple base models, adding some complexity but can balance it through simple interpretable models like trees.

Comparisons of Hyperparameter Sensitivity:

- SVM and Neural Network are highly sensitive to hyperparameters like kernel type, regularization, architecture. Slight changes can drastically impact performance.

- Linear models are more robust to hyperparameter changes. For example, Logistic Regression weight coefficients are relatively stable.
- Ensembles can overcome hyperparameter sensitivity of complex base models through aggregation.

In summary, these additional comparisons provide deeper insights into model behaviors, training costs, data dependencies, complexities and hyperparameter sensitivities.

Conclusion

Through a comprehensive experimental study, this work demonstrated a rigorous model evaluation methodology for an order priority prediction task. Multiple machine learning models were assessed under varying conditions of holdout percentage, training percentage, and feature sets. The models were compared across several performance metrics including F1 score, precision, accuracy, and training time.

The Optimizable Ensemble model emerged as the overall best performer in terms of balancing precision and recall, as evidenced by its superior F1 score. For applications where precision is critical, the Efficient Logistic Regression was shown to be most effective. The Neural Network Tri-layered model achieved the highest accuracy across all experiments. Interpretability was maximized by simpler linear models like Logistic Regression and Linear SVM. Training efficiency was optimized through Logistic Regression and Linear SVM which maintained good performance even with lower training data sizes.

The analysis revealed how holdout percentages did not drastically impact model performance once sufficient training data was utilized. However, expanding the training set consistently improved model effectiveness across the board. Reducing feature space decreased model performance slightly but helped safeguard against overfitting. Overall, the comparisons quantified the trade-offs between different models and conditions.

These findings demonstrate the importance of thoroughly evaluating multiple modeling approaches tailored to the problem and data at hand. The predictive modeling methodology presented provides a template for data-driven model selection based on application priorities. As shown in this study, harnessing the strengths of different machine learning techniques through comprehensive comparative analysis can lead to optimal solutions for priority prediction tasks.