FINAL REPORT

Master's in Engineering Management- Thayer School of Engineering

Team: RelaxedTherm

Pranav Dharmadhikari, *Roshni Govind*, *Kishore Kumaran*,
Vibha Visvanathan Kamala

1. Introduction

Hypertherm Associates is a leading provider of industrial cutting systems, software, and consumables for various applications. A significant aspect of Hypertherm's operations involves cutting steel parts and providing nesting software to optimize their placement on steel sheets. However, the traditional process of determining part placements and cutting steel sheets is often time-consuming, leading to substantial scrap material and low utilization of steel sheets. This inefficiency increases production costs and contributes to higher carbon emissions and environmental waste. In response to this challenge, Hypertherm has recognized the potential of integrating artificial intelligence (AI) models into their workflow to streamline the cutting process and reduce steel waste. This report explores the methodology employed, the AI-driven solution proposed, and the significant impact it can have on both environmental sustainability and enterprise efficiency. By optimizing the utilization of steel sheets and minimizing waste, this innovative approach aims to lower carbon emissions while enhancing productivity and profitability for Hypertherm and its customers.

1.1 Problem Statement

Hypertherm aims to leverage AI to improve part nesting on steel sheets, exceeding current utilization rates to help customers reduce carbon footprints and cut scrap costs**.**

1.2 Nesting

Nesting is a process used by Hypertherm to cut sheet metal into parts according to the client's specifications while strategically arranging them on a parent material, such as a steel plate, to maximize material utilization and minimize waste. Our team compared the process of Nesting to the classic video game Tetris, where players must carefully position falling tetromino shapes onto a playing field without leaving gaps. At Hypertherm, advanced nesting software and algorithms are employed to optimize the placement of parts on steel plates, considering factors such as part dimensions, sheet dimensions, material usage, etc. Our goal was to identify the most efficient nesting pattern that yields the highest sheet utilization, thereby minimizing material waste and reducing production costs. According to the information provided by Hypertherm, there are 17 different strategies available for nesting parts on a steel plate, each producing a different nesting pattern and varying levels of sheet utilization. Hypertherm's nesting technology explores these strategies, evaluating factors like part orientation, spacing between parts, and material usage to determine the optimal part placements that maximize sheet utilization.

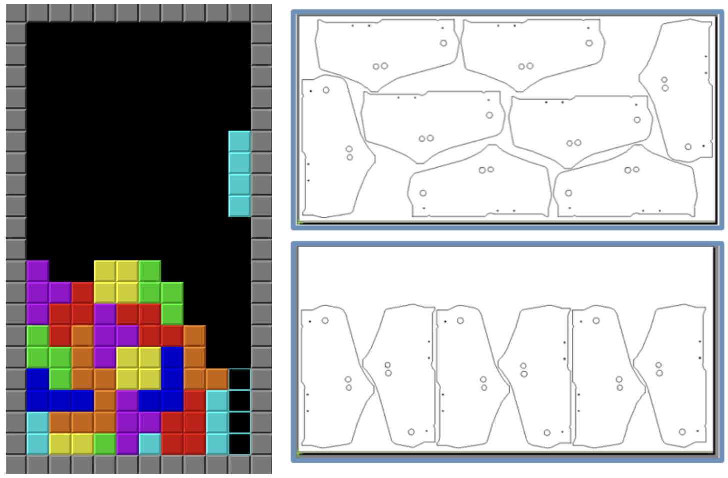


Fig 1: Nesting Process used by the software

1.3 Workflow and Pain Points

Phase 1: Client Requirements

Hypertherm receives three crucial pieces of information from the client:

- **Raw Materials:** For this project's scope, we considered steel plates to be the raw material provided by the client.
- **Time:** The client then provides their availability, logistics, and timeline for the project since time is of the essence when it comes to manufacturing,
- **Order Specifications:** The client then provides Hypertherm with detailed specifications for the order, which include the projected number of parts, type of parent material, part dimensions, and the complexity of the design.

Phase 2: Nesting Strategy Determination

After receiving the client's requirements, Hypertherm moves to the second phase, where they determine the nesting strategy for optimizing steel plate utilization.

There are two methods employed primarily:

- **Pronest Software Auto-Strategy:** Hypertherm uses specialized nesting software to analyze the order and determine the most efficient nesting strategy for the client based on myriad factors. This process is time-consuming as it tests multiple options before finalizing the optimal strategy.
- **Manual Strategy Selection:** During Manual Strategy Selection, clients typically rely on prior experience, personal preference, or external recommendations to determine the best strategy for nesting components onto the sheet. This results in picking a strategy that is not always optimal for the client.

Pain Points:

- **Time Consumption:** The most crucial pain point is the time the nesting software requires to complete the nesting process and run different strategies. It often takes more than 4 hours or even overnight to run a single strategy for an order, which typically consists of about 6 nests. This prolonged computing time for each strategy significantly hampers efficiency and production timelines. Running multiple strategies to evaluate and compare their performance becomes increasingly time-consuming, leading to potential delays and bottlenecks in the overall production process.

- **Randomization Inaccuracy:** Manual Strategy Selection often leads to suboptimal outcomes for clients, as reliance on past experiences, personal preferences, or external advice may not always align with the most efficient or effective nesting strategy, potentially resulting in wasted time and resources.

We aim to address these pain points to improve the efficiency and accuracy of their nesting strategy determination process, ultimately optimizing steel plate utilization, enhancing their manufacturing capabilities, and reducing carbon emissions.

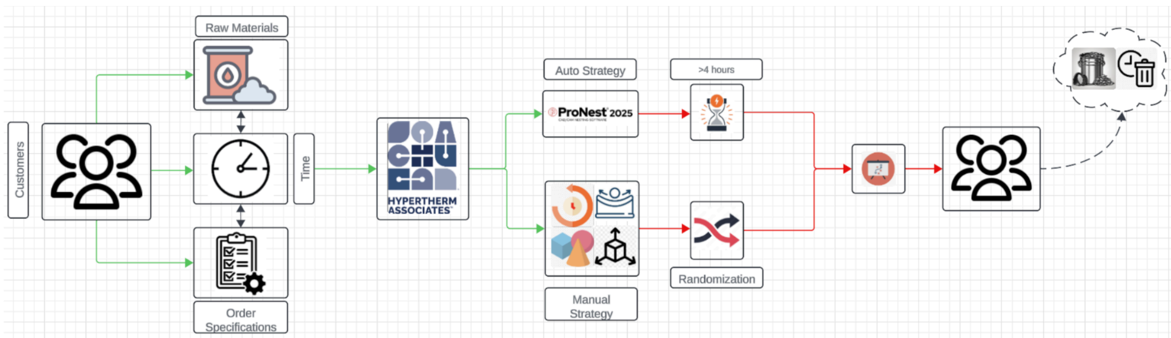


Fig 2: Problem Statement Development Process

2. About the Dataset

Historical data was obtained for this project from the Hypertherm team. This dataset contained all the data collected by Hypertherm's nesting software. In this dataset, the tables of Parts and Nest are the only two pertinent to our project. The schema of these tables is as follows:

2.1 Nest Data

Field	**Description**	Comments
ixJobSummary	Record Index of job containing this nest	
cTimesCut	The number of times the nest will be cut	
fOutput	Has the nest been output?	
cParts	Total number of parts nested	
cSafeZones	Number of safe zones used on the nest	
ixPlateType	Type of plate used	
dNestingTime	Total time spent auto-nesting	
fStrategies	Auto-nesting strategies used	
cMaxTorches	Maximum number of torches on nest	
dMaxTorchSpacing	Maximum torch spacing used on nest	

dLength	Sheet length	
dWidth	Sheet width	
dArea	Sheet area	
ixMaterial	Record index of material used	
dLengthUsed	Length of plate used by nested parts	
dWidthUsed	Width of plate used by nested parts	
dCropUtil	Nested utilization of parts inside of crop (if one exists)	Nested part area / (sheet area – area of remnants saved from nest)
dPartArea	Total area of nested parts	
dTrueArea	Plate area used by nested parts	True area of sheet.

2.2 Part Data

Field	Description	Comments
ixPart	Part record index	
ixJobSummary	Record Index of job containing this part	
dLength	Part length	
dWidth	Part width	
dArea	Part true area	Area of exterior profile – area of cutouts
cRequired	Number of parts required	
cNested	Number of parts nested	
ixMaterial	Record index of material used	
fExtShape	Shape of the exterior profile	About 50 known shapes
dExtArea	Area of the exterior profile	
dExtBoundaryDist	The maximum distance of any point inside the profile to the nearest point on the exterior profile	Uses a distance transform with distance measured at 45-degree increments

dExtContainedDist	The maximum unbroken distance between any two points on the exterior profile	The line between the two points doesn't intersect the profile anywhere else
dLgIntArea	Area of the largest interior profile	
dLgIntBoundaryDist	The maximum distance of any point inside the profile to the nearest point on the interior profile	Uses a distance transform with distance measured at 45-degree increments
dLgIntContainedDist	The maximum unbroken distance between any two points on the interior profile	The line between the two points doesn't intersect the profile anywhere else
dLgExtConArea	Area of the largest concavity	
dLgExtConBoundaryDist	The maximum distance of any point inside the concavity to the nearest point on the concavity	Uses a distance transform with distance measured at 45-degree increments
dLgExtConContainedDist	The maximum unbroken distance between any two points on the concavity contour	The line between the two points doesn't intersect the profile anywhere else

The Nest table contains values registered after the parts have been nested. It mainly captures data about the area of the parts nested, the sheet used for nesting and its utilization, the number of torches used for cutting, the number of parts nested in the nest, and the dimensions and material of the sheet.

The Part table contains details that capture major trends in the dimensions of the part, like the length, width, and area of the part, info regarding exterior and interior profile (exterior profile is the external boundary of a part, and interior profile is the internal boundary if a part contains any cutouts in it), details regarding concavity of a part (for example, a horseshoe part contains a concave region where other parts could be nested), and the number of parts required to be nested and the number of parts successfully nested by the software.

3. Data Cleaning

Pre-cleaning the Parts and Nest table contained 30 million and 8.7 million records, respectively. After pruning erroneous data, the numbers were brought down to 11 million and 3.8 million records, respectively. This was achieved by pruning the following anomalies from the dataset.

Nest Table:

- Records with dCropUtil = 0
- Records with dPartArea = 0
- Records with dTrueArea = 0
- Records with dLengthUsed = 0 and dLengthUsed < 0
- Records with dWidthUsed = 0 and dWidthUsed < 0

- Records with cParts = 0
- Records with fStrategies outside of 17 strategies defined in the problem statement
- In a few Records, dLengthUsed and dWidthUsed are more than the dLength and dWidth, so we selected a tolerance of 2 and deleted rows where the deviation is more than this, and corrected the values within this tolerance
- Rows where dPartArea is greater than dTrueArea
- dCroputil was calculated based on dTrueArea for over 60% of data and on dArea for the remaining records. Standardized to a calculation based on dTrueArea based on feedback from Hypertherm

Part Table:

- Rows with dLength = 0 and dLength < 0
- Rows with dWidth = 0 and dWidth < 0
- Rows with dArea = 0 and dArea < 0
- Rows with cNested = 0 and cNested < 0

Finally, the cNested (number of parts nested) and dArea (area of part) in a job from the Part table should be used and matched with the cParts (number of parts in the nest) and dPartArea (area of parts in the nest) nests within a job and the area of nested parts within a job to ensure synergy between the Part and Nest tables.

4. Data Preprocessing

For experimenting with ML models, we decided to use the following columns:
[dPartArea_Job, dLength_Avg, dWidth_Avg, dArea_Avg, cNested_Avg, fExtShape_Avg, dExtArea_Avg, dExtBoundaryDist_Avg, dExtContainedDist_Avg, dLgIntArea_Avg, dLgIntBoundaryDist_Avg, dLgIntContainedDist_Avg,dLgExtConArea_Avg,dLgExtConBoundaryDist_Avg, dLgExtConContainedDist_Avg]

All of these columns are obtained from the Part table. The dPartArea_Job is the sum of the area of all the parts in a job, calculated using the fields dArea and cNested of all parts in a job. All the other columns are averaged with other part records in a job. This forms the features to train ML models, the target being the CropUtil obtained from the Nest table. Our aim is to train 17 regression ML models (one for each nesting strategy) to predict output crop utilization for each strategy. Such predictive output will help choose the optimal strategy for a given combination of parts.

We also experimented with taking all records in the Part table individually and using them as a sequence input for each job into an LSTM model. Still, since this approach didn't yield promising results, we decided to drop it.

5. Models

Multiple machine-learning techniques were explored and evaluated to model and predict utilization accurately. These included linear regression, artificial neural networks (ANNs), and an ensemble approach combining autoencoders with random forests. The primary objective was to identify the most suitable modeling approach for each specific strategy present in the dataset.

Strategy-Aware Modeling

A crucial aspect of the modeling process was the development of individual models tailored to each nesting strategy. This strategy-aware approach offered several significant advantages:

Strategy-Specific Learning

By segregating the data and training models independently for each strategy, the algorithms could specialize in understanding the unique nuances, distributions, and intricate patterns inherent to each strategy's data. This tailored learning approach provided several benefits:

- Enhanced Specialization: Models could focus on each strategy's specific characteristics without being influenced by data from other strategies.
- Reduced Cross-Strategy Interference: Data segregation ensured the learning process was free from bias by mixing different strategies.
- Improved Accuracy: Models trained on specific strategies were better equipped to make accurate, strategy-aligned predictions.

After experimenting with various models, including Lasso and Ridge regression and LSTMs, which did not yield optimal results, the linear regression algorithm was chosen as the benchmark due to its superior MAE performance on this dataset. The details of these experiments are discussed in the following section.

5.1 Linear Regression

Linear regression is one of the most commonly used and well-established machine learning algorithms for regression problems. It is easy to implement, interpretable, and serves as a reasonable baseline model to compare against more complex techniques.

5.1.1 Model Architecture

The linear regression modeling process begins by grouping the dataset according to the different strategies, resulting in separate subsets of data for each strategy. Individual linear regression models are then trained independently on these strategy-specific datasets to predict utilization tailored to the unique data distributions within each strategy group.

Linear regression aims to find the best-fitting linear equation ($y = mx + b$) to map the input features (x) to the target utilization variable (y) for the data points corresponding to a given strategy. The linear regression algorithm fits this equation by minimizing the sum of squared residuals between the predicted and actual utilization values during the training process.

5.1.2 Performance Evaluation

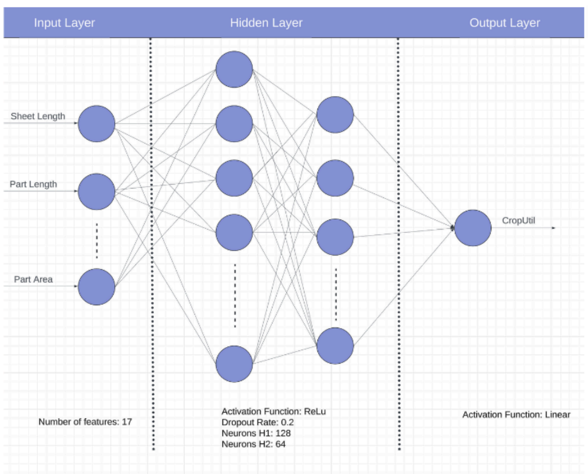
The average mean absolute error (MAE) for the linear regression models across all strategies is 11.36%.

5.2 Artificial Neural Networks (ANN)

ANNs can automatically learn and extract significant features from the raw input data through their hidden layers, reducing the need for manual feature engineering. Their flexibility and ability to model non-linear relationships make them an excellent choice for this regression problem.

5.2.1 Model Architecture

The ANN modeling process begins by standardizing the input data using sklearn's StandardScaler. The standardized data is then split into training and validation sets. A randomized search is performed over key hyperparameters to optimize the ANN architecture, including the learning rate, dropout rates for the hidden layers, and the number of neurons in the two hidden layers. This hyperparameter tuning utilizes 5-fold cross-validation on the training set to evaluate a grid of hyperparameter combinations.



The optimal configuration is selected from the set of hyperparameters that results in the lowest average mean absolute error (MAE) across the cross-validation folds. This optimized architecture is then used to create an ANN model consisting of an input layer with neurons equal to the number of features, two hidden layers with the tuned number of neurons and dropout rates, and an output layer with a single neuron for the regression output.

This optimized ANN model is trained on the training data using the Adam optimizer. During training, callbacks like EarlyStopping and ReduceLROnPlateau help prevent overfitting and dynamically adjust the learning rate based on validation loss. The performance of the trained model is evaluated on the held-out validation set using MAE as the metric. Once trained, the ANN model can predict new, previously unseen data after standardizing the inputs.

5.2.2 Performance Evaluation

The average MAE across the different ANN models is 8.17%, lower than the regression model.

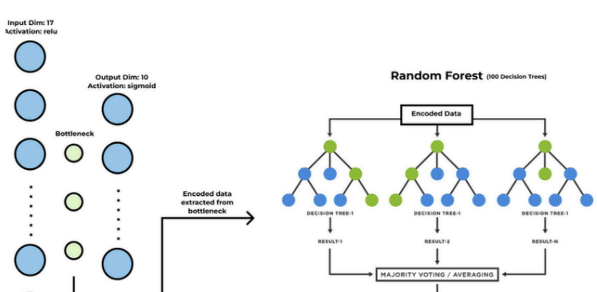
5.3 Autoencoder-Random Forest Model

In predictive analytics, combining different models enhances prediction accuracy and efficiency. Based on preliminary research, leveraging lower-dimensional feature learning alongside a random forest regressor seemed highly effective for Hypertherm's use case and the dataset provided. The autoencoder reduces dimensionality and extracts key features, while the Random Forest model makes robust, unbiased predictions. This ensemble creates a precise and reliable output for optimizing steel utilization.

5.3.1 Model Architecture

The hybrid model architecture comprises two main components: the Autoencoder and the Random Forest regressor.

Autoencoder:



The autoencoder is an artificial neural network designed for unsupervised learning. It consists of an input layer, an encoder layer, a bottleneck (latent space), and a decoder layer. In this setup, the input layer has 17 units, corresponding to the 17 columns of the dataset.

The encoder compresses this input data into a lower-dimensional space with 10 units, effectively capturing the most significant features while discarding noise and redundancy. The bottleneck layer represents this compressed data. Although the decoder layer typically reconstructs the input data from the encoded form, in this model, the focus remains on the encoded data passed to the next component.

Random Forest Regressor:

After the encoding process, the compressed data is fed into a Random Forest regressor. This ensemble learning method comprises 100 decision trees, providing a well-rounded mechanism without biases for making predictions. The Random Forest regressor uses the encoded data to learn patterns and relationships, enabling it to predict steel sheet utilization effectively.

Data Flow and Prediction

The process begins with the dataset, which includes 17 features related to steel sheet utilization strategies. The autoencoder first processes this data, reducing it to a 10-dimensional encoded form. The Random Forest regressor then uses this encoded data to predict the utilization of each strategy.

5.3.2 Performance Evaluation

The average MAE across the models built on different strategies is 6.15%, the best of the models we tested for this use case. Hence, the AE-RF is the proposed model for making steel utilization predictions.

6. Final Proposed Solution and Result:

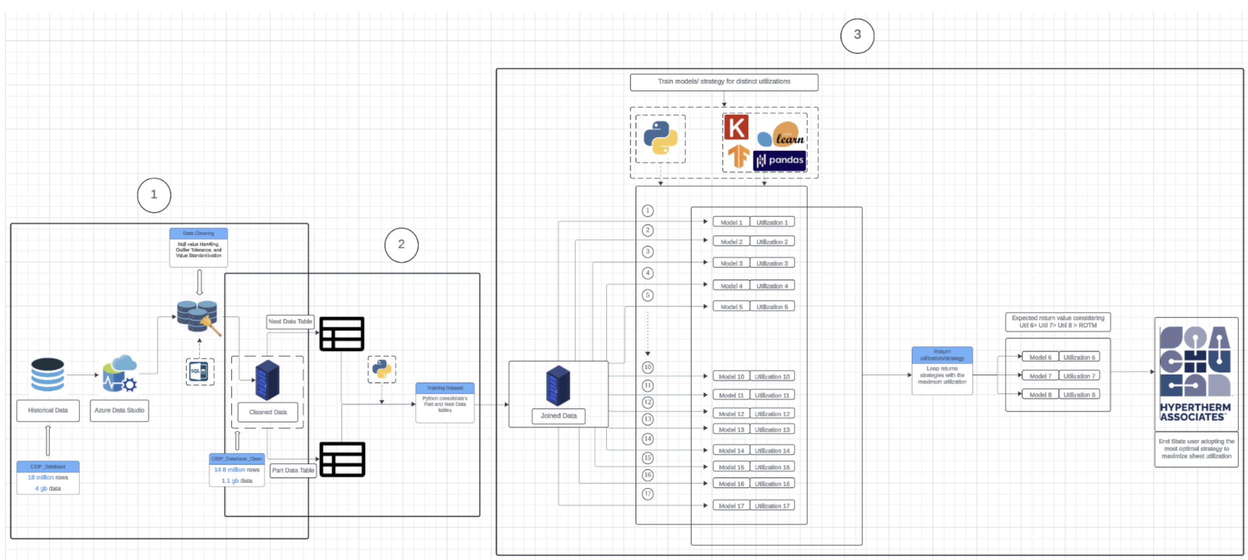


Fig 3: Final Solution Architecture

Following data cleaning, preprocessing, and modeling stages, the ultimate solution delivers utilization predictions for all strategies. A consolidated function incorporating all models facilitates the prediction of utilization for new inputs and outputs of the top three strategies with the highest anticipated utilization.

Hypertherm receives this output, enabling them to select the optimal strategy among the three according to their preferences while considering the raw material usage.

By leveraging historical data as input across all 17 models developed for various strategies, the predictions generated exhibit an average 10% increase in utilization compared to existing methods, thereby enhancing the efficiency and effectiveness of strategies for Hypertherm and their clientele.

	fStrategies	CropUtil	Best Strategy Predicted	Predicted CropUtil
0	0	0.022530	64	0.662942
2	8	0.636089	128	0.739873
3	8	0.634186	16384	0.709342
4	0	0.768649	256	0.769121
5	128	0.413703	256	0.606014
...	...	...	...	...
1790931	0	0.262695	4	0.702255
1790932	4	0.640358	2048	0.811541
1790933	8	0.617906	64	0.819938
1790934	8	0.509921	64	0.739062
1790935	0	0.648776	128	0.826359

Fig 4: Output of the Model based on Historical Data

6.1 Impact

There were 3 impact factors that we considered:

6.1.1 Common Assumptions

Pre-Implementation:

- **Number of Orders/year:** This indicates that Hypertherm received 302,000 orders annually before integrating RelaxedTherm's implementation into its workflow.
- **Feet of Metal Cut/year:** The company cuts 302,000 feet of metal each year to fulfill the orders. The team considered that each order is 1 foot long and has 1 nest.
- **Lbs of scrap/foot of cut:** For every foot of metal cut, 3.72 pounds of scrap metal was generated as waste.

Post-Implementation:

- **Number of Orders/year:** After integrating RelaxedTherm's implementation in their workflow, the number of orders remained unchanged at 302,000 per year.
- **Feet of metal Cut/year:** The total footage of metal cut remained the same at 302,000 feet. The team considered that each order is 1 foot long and has 1 nest.

- **Lbs of scrap/foot of cut:** The amount of scrap metal generated per foot of cut decreased to 2.74 pounds due to the 10% increased utilization.

1. **Human Health Impact:**

To assess the human health impact of excessive carbon emissions, we calculated the number of years lived with disability and life years lost/ year, quantified in the Disability-Adjusted Life Years (DALYs) metric.

Pre-Implementation:

- **Current Utilization:** Hypertherm's nesting process had a utilization rate of 62%, suggesting room for improvement in efficiency.
- **Years cut short (DALY):** The health impact of the metal-cutting process resulted in a loss of 0.00000209 years of healthy life per year per order due to premature death or disability.
- **Days cut short/year:** Due to excessive carbon emissions, the metal-cutting process led to 230.38 days of healthy life lost per year for 302,000 orders for one person.

Post-Implementation:

- **Current Utilization:** The utilization rate improved to 72%, suggesting better efficiency in the metal-cutting process.
- **Years cut short (DALY):** The health impact of the metal-cutting process was significantly reduced, with only 2.0273E-06 years of healthy life lost per year per order.
- **Days cut short/year:** The number of days of healthy life lost per year decreased to 223.47 days, a reduction compared to before the implementation.

The integration saved 6.91 days of healthy life per year for 302,000 orders for one person, indicating an improvement in the overall health impact of the metal-cutting process.

2. **Ecosystem impact:**

This factor indicates the product of the number of species facing extinction due to excessive carbon emissions, the geographic area they inhabit, and the impact duration quantifies the overall biodiversity loss.

Pre-Implementation:

- **Number of species that may disappear/ year/ order due to excessive carbon emissions** is 0.00000001. This suggests that before RelaxedTherms implementation was integrated, the excessive carbon emissions were causing several species to go extinct per order per year.
- **Number of species that may disappear per year** due to the impact is 0.0112344 for 302000 orders. This indicates that before mitigation, the impact was causing the potential extinction of approximately 0.0112344 species per year while Hypertherm ran at full capacity.
- **Number of species that may disappear in the next 50 years** due to the impact is 0.56172 for 302000 orders which essentially means that over a longer time frame of 50 years, the pre-implementation impact was projected to cause the potential extinction of 0.56172 species, which is a more substantial number.

Post-Implementation:

- **Number of species that may disappear per year per order due to excessive carbon emissions** is 9.7E-09. This suggests that after integrating RelaxedTherms, the number of species potentially going extinct per order per year due to the impact was reduced to an even smaller number of 9.7E-09, a significant reduction compared to the pre-implementation value.

- **Number of species that may disappear per year** due to the impact is 0.00802964 which means that the overall number of species potentially going extinct per year due to the impact was also reduced to 0.00802964 after integrating RelaxedTherms implementation.
- **Number of species that may disappear in the next 50 years** due to the impact would be 0.401481979 which means over the next 50 years, the post-implementation impact is projected to cause the potential extinction of 0.401481979 species.

By subtracting the post-implementation value (0.401481979) from the pre-implementation value (0.56172), we can see that our implementation would potentially save 0.160238021 species from extinction over the next 50 years due to the impact.

3. **Resource Impact:**

The resource impact represents the future value of resources unavailable due to their current utilization.

Pre-implementation:

- **Number of resources impacted/order/year:** On average, $6/ order was impacted for every order placed within a year, indicating a considerable strain on resource availability.
- **Number of resources impacted/year:** The total number of resources impacted annually was $1,812,000 for 302000 orders, highlighting the substantial resource consumption before the RelaxedTherms solutions were implemented.

Post-implementation:

- **Number of resources impacted/order/year:** After the team's implementation, the average amount saved per order placed annually decreased to $5.82/ order, reflecting a noticeable improvement in resource utilization efficiency.
- **Number of resources impacted/year:** The total number of resources impacted yearly saw a significant reduction to $1,757,640 for 302000 orders which annually underscores the positive impact of our integration in terms of resource conservation and monetary benefits.

This hints that the future value of resources that would be unavailable due to their utilization in the present is more pre-implementation than post-implementation, which essentially suggests that Hypertherm would save $181200/year for 302000 orders.

	Human Health Impact		Ecosystem impact		Resources impact ($/yr)	
	Pre implementation	Post Implementation	Pre implementation	Post Implementation	Pre implementation	Post Implementation
Number of Orders/year	302,000	302,000	302,000	302,000	302,000	302,000
Feet of metal Cut/ year	302,000	302,000	302,000	302,000	302,000	302,000
lbs of scrap/foot of cut	3.72	2.741052632	3.72	2.741052632	3.72	2.741052632
Current Utilization	62.00%	72.00%	62.00%	72.00%	62.00%	72.00%
Years cut short (DALY)	0.00000209	2.0273E-06				
Days cut short/ year	230.3807	223.469279				
Number of Days Saved	6.911421					
Number of species that may disappear/year/order due to the impact			0.00000001	9.7E-09		
Number of species that may disappear/year due to the impact			0.0112344	0.00802964		
Number of species that may disappear in the next 50 years to the impact			0.56172	0.401481979		
Number of species that would be saved in the next 50 years			0.160238021			
Number of Days Saved						
Resources($) impacted/order/year					6	5.40
Resources($) impacted/year					1812000	1630800
Resources($) Saved/ year					181200	

Fig 5: Calculation for Impact

6.1.2 Realistic Impacts:

1. **Resource Impacts:**

The resource impact represents the future value of resources unavailable due to their current utilization.

Considerations:

- Hypertherm's medium to large customers might cut 300,000 ft to 1.3 million ft per year. Using this data (provided by the client), we consider the average order to be 800,000 ft per year per customer.
- We believe that Hypertherm has 1000 customers, and each customer gives them 1 order of 800,000 ft per year.

Results:

- Considering the factors considered above (Number of resources impacted/order/year & Number of resources impacted/year), the team concluded that Hypertherm's $480,000,000 worth of resources would be saved.

2. **Carbon Emission Impacts:**

The resource impact represents the future value of resources unavailable due to their current utilization.

Considerations:

- Hypertherm's medium to large customers might cut 300,000 ft to 1.3 million ft per year. Using this data (provided by the client), we consider the average order to be 800,000 ft per year per customer.
- We believe that Hypertherm has 1000 customers, and each customer gives them 1 order of 800,000 ft per year.

Results:

- Considering the above factors (KgCO2e emitted/order/year & KgCO2e emitted/order/year), the team concluded that Hypertherm would emit less than 25263157.89 KgCO2e/ year.

	Carbon Emission impact (KgCO2e/year)	
	Pre Implementation	Post Implementation
KgCO2e carbon emitted/ foot of cut	0.12	0.09
KgCO2e carbon emitted/ year	96000000.00	70736842.11
	25263157.89	

Realistic Resources impact ($/yr)	
Pre implementation	Post Implementation
1000	1000
800000000	800000000
3.72	2.741052632
62.00%	72.00%
6	5.40
4800000000	4320000000
480000000	

Fig 6&7: Realistic Calculation for Impact

7. Conclusion:

In conclusion, through this project, we have performed extensive data exploration, analysis, and cleaning resulting in over 60% of data being pruned. Multiple ML and neural network-based models were evaluated, with the Linear Regression model serving as a baseline model with an average MAE of 11% and the Autoencoder - Random Forest Regression model giving the best performance with an average MAE of 6%. With the help of the Autoencoder - Random Forest model, we ran a regressive analysis on the historical dataset. We found that if the predicted strategy was used for all the previous nesting jobs, an average of 10% increase in utilization could have been achieved, showcasing the potential of integrating our project into Hypertherm's nesting software.

8. Future Scope

- Integrating the Model into a Website/Application: To facilitate broader accessibility and user-friendly interactions, the modeling pipeline could be integrated into a web-based platform or application. This would involve developing a user interface where users can input relevant data, obtain utilization predictions, and recommend top strategies.
- Scaling the Model with Different Input Metrics: The current implementation utilizes part averages as input features. However, exploring and incorporating different metrics or input data aggregations could improve prediction accuracy and provide more comprehensive insights.
- Testing Out Other Models: While the current implementation is an ensemble of autoencoders with random forests, machine learning is rapidly evolving. Exploring and evaluating other state-of-the-art models could yield improved performance or uncover new insights.

Appendix

Mean Absolute Error for all Models

MAE - LR

Strategy -2147483648 - LR MAE: 0.1724384731017937
Strategy 0 - LR MAE: 0.17007085061048358
Strategy 1 - LR MAE: 0.12087709037506887
Strategy 2 - LR MAE: 0.07587781897573141
Strategy 4 - LR MAE: 0.10831671082297097
Strategy 8 - LR MAE: 0.11130536659789568
Strategy 16 - LR MAE: 0.10089318489413998
Strategy 32 - LR MAE: 0.11463965895750547
Strategy 64 - LR MAE: 0.10674503232528745
Strategy 128 - LR MAE: 0.11334468477553149
Strategy 256 - LR MAE: 0.09056337047297233
Strategy 512 - LR MAE: 0.11708796531329728
Strategy 1024 - LR MAE: 0.12435057973973052
Strategy 2048 - LR MAE: 0.12670126339086446
Strategy 4096 - LR MAE: 0.12620449579268
Strategy 8192 - LR MAE: 0.1336700162174162
Strategy 16384 - LR MAE: 0.1184001536864068

MAE - ANN

Strategy -2147483648 - ANN MAE: 0.1097107682471816
Strategy 0 - ANN MAE: 0.1004535755441663
Strategy 1 - ANN MAE: 0.0740200648623339
Strategy 2 - ANN MAE: 0.03738127748469544
Strategy 4 - ANN MAE: 0.07874878821943923
Strategy 8 - ANN MAE: 0.06088854574044074
Strategy 16 - ANN MAE: 0.06740776591310316
Strategy 32 - ANN MAE: 0.08809113846669805
Strategy 64 - ANN MAE: 0.09188961986987759
Strategy 128 - ANN MAE: 0.07266957436902598
Strategy 256 - ANN MAE: 0.08451832628498428
Strategy 512 - ANN MAE: 0.08299802368316325
Strategy 1024 - ANN MAE: 0.07359543907081563
Strategy 2048 - ANN MAE: 0.06385327250228227
Strategy 4096 - ANN MAE: 0.07603518800236708
Strategy 8192 - ANN MAE: 0.07218181656478946
Strategy 16384 - ANN MAE: 0.07263756478868126

MAE - Autoencoder Random Forest Ensemble

Strategy -2147483648 - Random Forest MAE: 0.09305657531236997
Strategy 0 - Random Forest MAE: 0.07994441079713896
Strategy 1 - Random Forest MAE: 0.0627151977764201
Strategy 2 - Random Forest MAE: 0.027509728272255507
Strategy 4 - Random Forest MAE: 0.0616838621334025
Strategy 8 - Random Forest MAE: 0.050668070367750534
Strategy 16 - Random Forest MAE: 0.05650405574998458
Strategy 32 - Random Forest MAE: 0.06422683678323825
Strategy 64 - Random Forest MAE: 0.07012840373321899
Strategy 128 - Random Forest MAE: 0.0579891598706541
Strategy 256 - Random Forest MAE: 0.054372319788147924
Strategy 512 - Random Forest MAE: 0.06566009398828265
Strategy 1024 - Random Forest MAE: 0.05673849683360188
Strategy 2048 - Random Forest MAE: 0.04805366126191594
Strategy 4096 - Random Forest MAE: 0.06200144543337612
Strategy 8192 - Random Forest MAE: 0.07299470779275806
Strategy 16384 - Random Forest MAE: 0.060406469730801014

Metrics used to calculate the impact

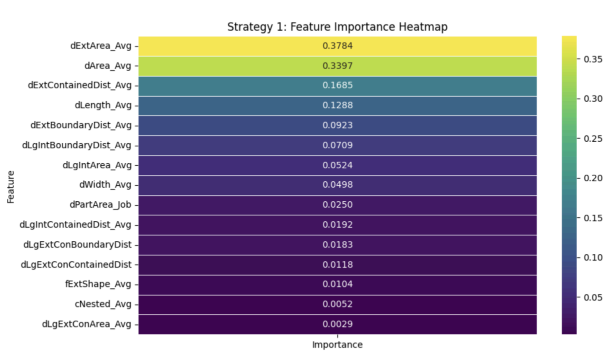
Impact Assessment Method

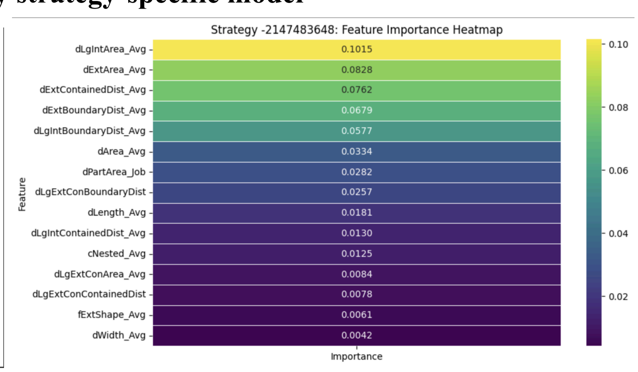
Damage Category	Units	From	Comments
Human health	DALY	ReCiPe Endpoint (H)	Accounts for years lived disabled as well as life cut short
Ecosystems	Species * yr	ReCiPe Endpoint (H)	Assessed in units of species * yr, or the number of species that may disappear due to the impact times the area over which they are affected times the duration that the species are affected
Resources	Economic units	ReCiPe Endpoint (H)	Puts a future value on resources which will be unavailable since we are using them today
Climate change	kg CO_2 eq.	IPCC 100a	Same method used by most GHG accounting programs
Water	m^3	ReCiPe Midpoint (H)	Counts the amount of water consumed. **Does not show impact. Used for benchmarking only.**
CED	MJ	Cumulative Energy Demand	Adds up different categories of energy

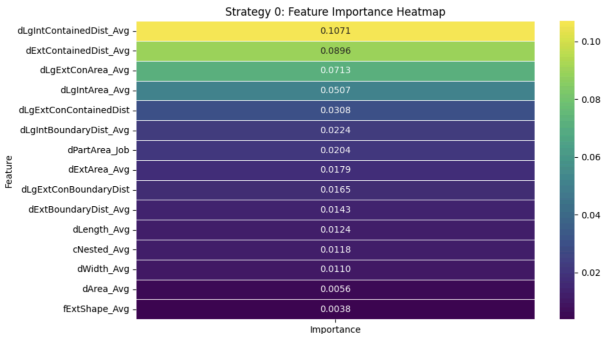
Impact Assessment Method: ES Method

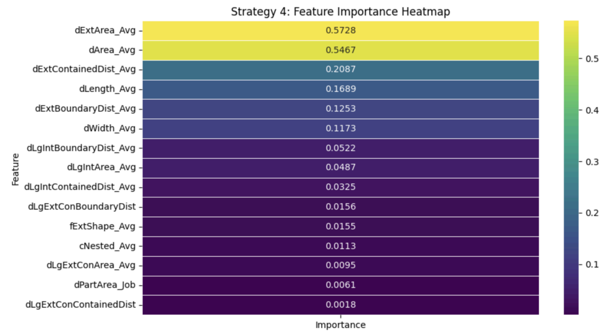
EarthShift
Building pathways to sustainability

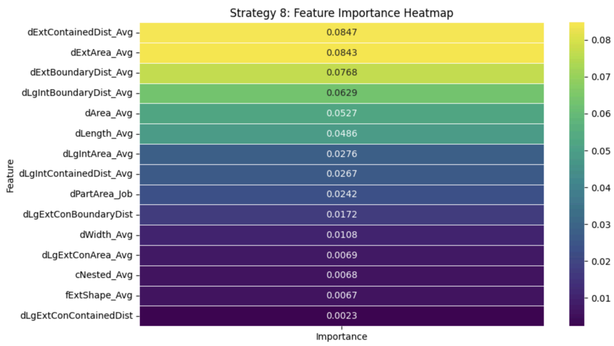
Every strategy has unique correlations with various features, which is why we've incorporated all these features, and it is also why we have strategy strategy-specific model

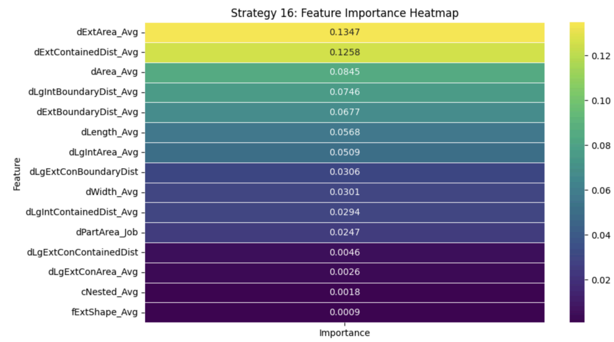


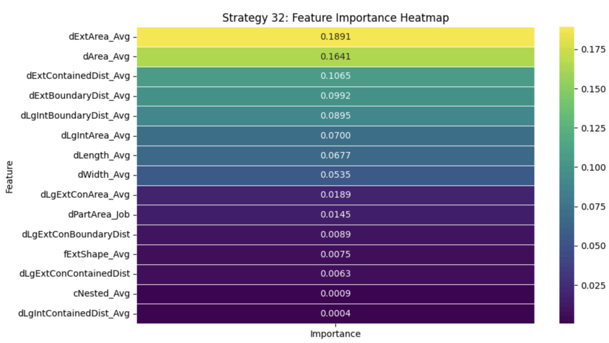


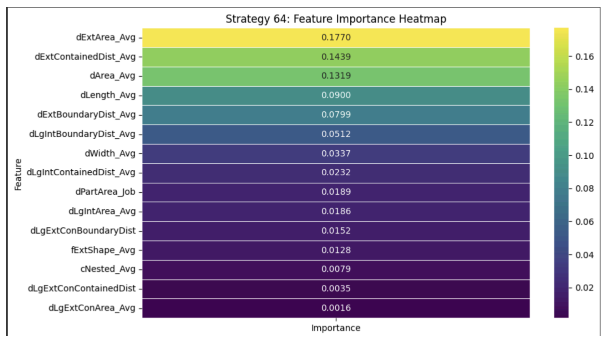

Strategy 0: Feature Importance Heatmap
dLgIntContainedDist_Avg 0.1071
dExtContainedDist_Avg 0.0896
dLgExtConArea_Avg 0.0713
dLgIntArea_Avg 0.0507
dLgExtConContainedDist 0.0308
dLgIntBoundaryDist_Avg 0.0224
dPartArea_Job 0.0204
dExtArea_Avg 0.0179
dLgExtConBoundaryDist 0.0165
dExtBoundaryDist_Avg 0.0143
dLength_Avg 0.0124
cNested_Avg 0.0118
dWidth_Avg 0.0110
dArea_Avg 0.0056
fExtShape_Avg 0.0038
Feature
Importance

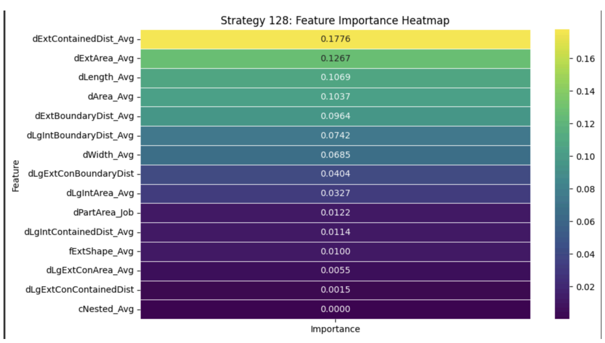

Strategy 4: Feature Importance Heatmap
dExtArea_Avg 0.5728
dArea_Avg 0.5467
dExtContainedDist_Avg 0.2087
dLength_Avg 0.1689
dExtBoundaryDist_Avg 0.1253
dWidth_Avg 0.1173
dLgIntBoundaryDist_Avg 0.0522
dLgIntArea_Avg 0.0487
dLgIntContainedDist_Avg 0.0325
dLgExtConBoundaryDist 0.0156
fExtShape_Avg 0.0155
cNested_Avg 0.0113
dLgExtConArea_Avg 0.0095
dPartArea_Job 0.0061
dLgExtConContainedDist 0.0018
Feature
Importance

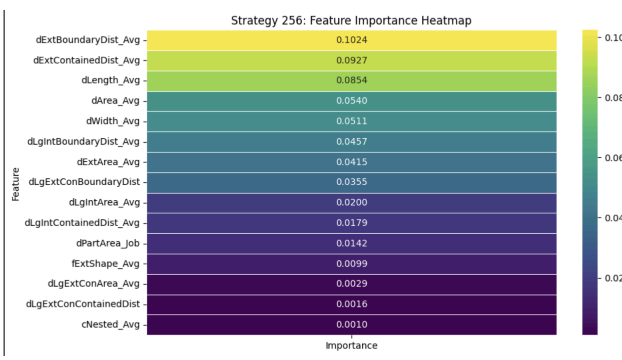

Strategy 8: Feature Importance Heatmap
dExtContainedDist_Avg 0.0847
dExtArea_Avg 0.0843
dExtBoundaryDist_Avg 0.0768
dLgIntBoundaryDist_Avg 0.0629
dArea_Avg 0.0527
dLength_Avg 0.0486
dLgIntArea_Avg 0.0276
dLgIntContainedDist_Avg 0.0267
dPartArea_Job 0.0242
dLgExtConBoundaryDist 0.0172
dWidth_Avg 0.0108
dLgExtConArea_Avg 0.0069
cNested_Avg 0.0068
fExtShape_Avg 0.0067
dLgExtConContainedDist 0.0023
Feature
Importance

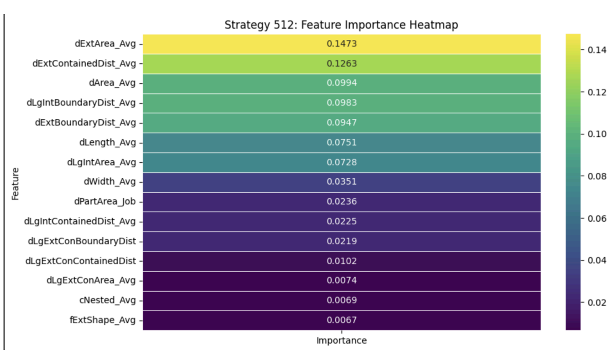

Strategy 16: Feature Importance Heatmap
dExtArea_Avg 0.1347
dExtContainedDist_Avg 0.1258
dArea_Avg 0.0845
dLgIntBoundaryDist_Avg 0.0746
dExtBoundaryDist_Avg 0.0677
dLength_Avg 0.0568
dLgIntArea_Avg 0.0509
dLgExtConBoundaryDist 0.0306
dWidth_Avg 0.0301
dLgIntContainedDist_Avg 0.0294
dPartArea_Job 0.0247
dLgExtConContainedDist 0.0046
dLgExtConArea_Avg 0.0026
cNested_Avg 0.0018
fExtShape_Avg 0.0009
Feature
Importance

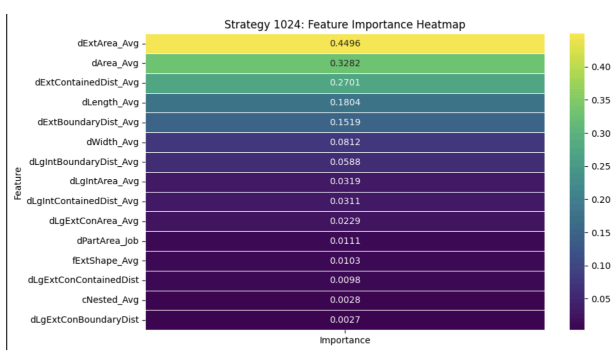

Strategy 32: Feature Importance Heatmap
dExtArea_Avg 0.1891
dArea_Avg 0.1641
dExtContainedDist_Avg 0.1065
dExtBoundaryDist_Avg 0.0992
dLgIntBoundaryDist_Avg 0.0895
dLgIntArea_Avg 0.0700
dLength_Avg 0.0677
dWidth_Avg 0.0535
dLgExtConArea_Avg 0.0189
dPartArea_Job 0.0145
dLgExtConBoundaryDist 0.0089
fExtShape_Avg 0.0075
dLgExtConContainedDist 0.0063
cNested_Avg 0.0009
dLgIntContainedDist_Avg 0.0004
Feature
Importance

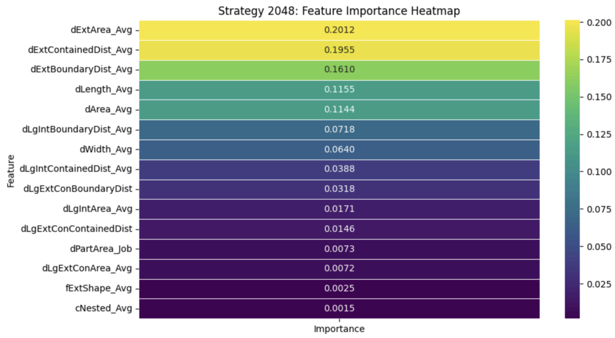

Strategy 64: Feature Importance Heatmap
dExtArea_Avg 0.1770
dExtContainedDist_Avg 0.1439
dArea_Avg 0.1319
dLength_Avg 0.0900
dExtBoundaryDist_Avg 0.0799
dLgIntBoundaryDist_Avg 0.0512
dWidth_Avg 0.0337
dLgIntContainedDist_Avg 0.0232
dPartArea_Job 0.0189
dLgIntArea_Avg 0.0186
dLgExtConBoundaryDist 0.0152
fExtShape_Avg 0.0128
cNested_Avg 0.0079
dLgExtConContainedDist 0.0035
dLgExtConArea_Avg 0.0016
Feature
Importance

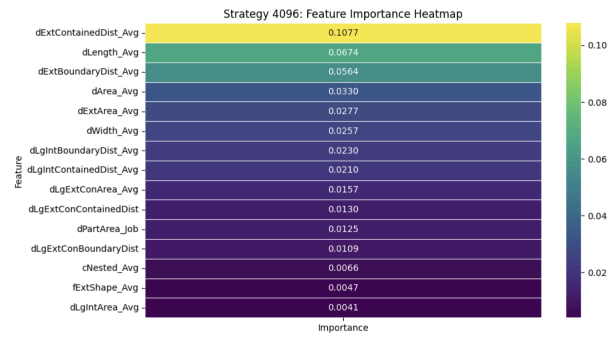

Strategy 128: Feature Importance Heatmap
dExtContainedDist_Avg 0.1776
dExtArea_Avg 0.1267
dLength_Avg 0.1069
dArea_Avg 0.1037
dExtBoundaryDist_Avg 0.0964
dLgIntBoundaryDist_Avg 0.0742
dWidth_Avg 0.0685
dLgExtConBoundaryDist 0.0404
dLgIntArea_Avg 0.0327
dPartArea_Job 0.0122
dLgIntContainedDist_Avg 0.0114
fExtShape_Avg 0.0100
dLgExtConArea_Avg 0.0055
dLgExtConContainedDist 0.0015
cNested_Avg 0.0000
Feature
Importance

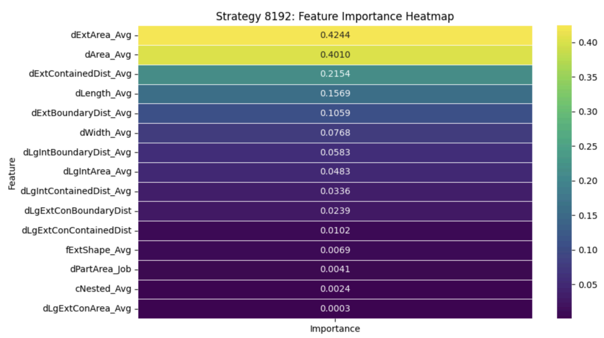

Strategy 256: Feature Importance Heatmap
dExtBoundaryDist_Avg 0.1024
dExtContainedDist_Avg 0.0927
dLength_Avg 0.0854
dArea_Avg 0.0540
dWidth_Avg 0.0511
dLgIntBoundaryDist_Avg 0.0457
dExtArea_Avg 0.0415
dLgExtConBoundaryDist 0.0355
dLgIntArea_Avg 0.0200
dLgIntContainedDist_Avg 0.0179
dPartArea_Job 0.0142
fExtShape_Avg 0.0099
dLgExtConArea_Avg 0.0029
dLgExtConContainedDist 0.0016
cNested_Avg 0.0010
Feature
Importance

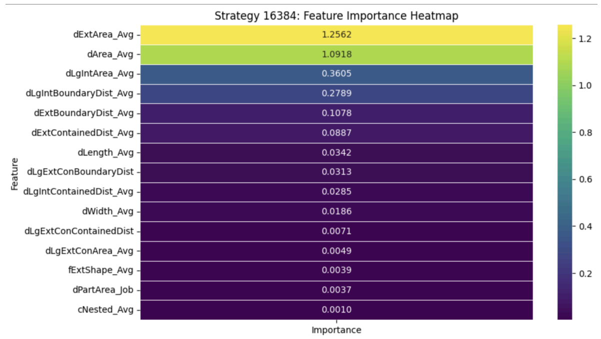

Strategy 512: Feature Importance Heatmap
dExtArea_Avg 0.1473
dExtContainedDist_Avg 0.1263
dArea_Avg 0.0994
dLgIntBoundaryDist_Avg 0.0983
dExtBoundaryDist_Avg 0.0947
dLength_Avg 0.0751
dLgIntArea_Avg 0.0728
dWidth_Avg 0.0351
dPartArea_Job 0.0236
dLgIntContainedDist_Avg 0.0225
dLgExtConBoundaryDist 0.0219
dLgExtConContainedDist 0.0102
dLgExtConArea_Avg 0.0074
cNested_Avg 0.0069
fExtShape_Avg 0.0067
Feature
Importance

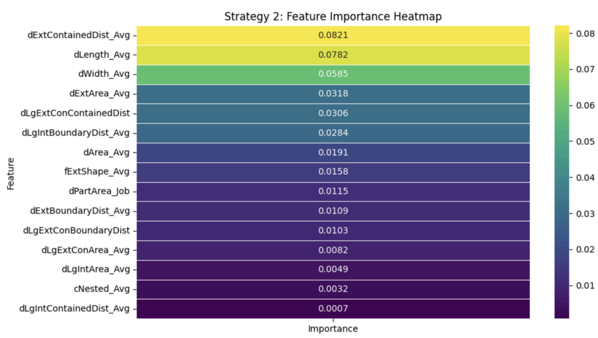

Strategy 1024: Feature Importance Heatmap
dExtArea_Avg 0.4496
dArea_Avg 0.3282
dExtContainedDist_Avg 0.2701
dLength_Avg 0.1804
dExtBoundaryDist_Avg 0.1519
dWidth_Avg 0.0812
dLgIntBoundaryDist_Avg 0.0588
dLgIntArea_Avg 0.0319
dLgIntContainedDist_Avg 0.0311
dLgExtConArea_Avg 0.0229
dPartArea_Job 0.0111
fExtShape_Avg 0.0103
dLgExtConContainedDist 0.0098
cNested_Avg 0.0028
dLgExtConBoundaryDist 0.0027
Feature
Importance


Strategy 2048: Feature Importance Heatmap
dExtArea_Avg 0.2012
dExtContainedDist_Avg 0.1955
dExtBoundaryDist_Avg 0.1610
dLength_Avg 0.1155
dArea_Avg 0.1144
dLgIntBoundaryDist_Avg 0.0718
dWidth_Avg 0.0640
dLgIntContainedDist_Avg 0.0388
dLgExtConBoundaryDist 0.0318
dLgIntArea_Avg 0.0171
dLgExtConContainedDist 0.0146
dPartArea_Job 0.0073
dLgExtConArea_Avg 0.0072
fExtShape_Avg 0.0025
cNested_Avg 0.0015
Feature
Importance
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025


Strategy 4096: Feature Importance Heatmap
dExtContainedDist_Avg 0.1077
dLength_Avg 0.0674
dExtBoundaryDist_Avg 0.0564
dArea_Avg 0.0330
dExtArea_Avg 0.0277
dWidth_Avg 0.0257
dLgIntBoundaryDist_Avg 0.0230
dLgIntContainedDist_Avg 0.0210
dLgExtConArea_Avg 0.0157
dLgExtConContainedDist 0.0130
dPartArea_Job 0.0125
dLgExtConBoundaryDist 0.0109
cNested_Avg 0.0066
fExtShape_Avg 0.0047
dLgIntArea_Avg 0.0041
Feature
Importance
0.10
0.08
0.06
0.04
0.02


Strategy 8192: Feature Importance Heatmap
dExtArea_Avg 0.4244
dArea_Avg 0.4010
dExtContainedDist_Avg 0.2154
dLength_Avg 0.1569
dExtBoundaryDist_Avg 0.1059
dWidth_Avg 0.0768
dLgIntBoundaryDist_Avg 0.0583
dLgIntArea_Avg 0.0483
dLgIntContainedDist_Avg 0.0336
dLgExtConBoundaryDist 0.0239
dLgExtConContainedDist 0.0102
fExtShape_Avg 0.0069
dPartArea_Job 0.0041
cNested_Avg 0.0024
dLgExtConArea_Avg 0.0003
Feature
Importance
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05


Strategy 16384: Feature Importance Heatmap
dExtArea_Avg 1.2562
dArea_Avg 1.0918
dLgIntArea_Avg 0.3605
dLgIntBoundaryDist_Avg 0.2789
dExtBoundaryDist_Avg 0.1078
dExtContainedDist_Avg 0.0887
dLength_Avg 0.0342
dLgExtConBoundaryDist 0.0313
dLgIntContainedDist_Avg 0.0285
dWidth_Avg 0.0186
dLgExtConContainedDist 0.0071
dLgExtConArea_Avg 0.0049
fExtShape_Avg 0.0039
dPartArea_Job 0.0037
cNested_Avg 0.0010
Feature
Importance
1.2
1.0
0.8
0.6
0.4
0.2


Strategy 2: Feature Importance Heatmap
dExtContainedDist_Avg 0.0821
dLength_Avg 0.0782
dWidth_Avg 0.0585
dExtArea_Avg 0.0318
dLgExtConContainedDist 0.0306
dLgIntBoundaryDist_Avg 0.0284
dArea_Avg 0.0191
fExtShape_Avg 0.0158
dPartArea_Job 0.0115
dExtBoundaryDist_Avg 0.0109
dLgExtConBoundaryDist 0.0103
dLgExtConArea_Avg 0.0082
dLgIntArea_Avg 0.0049
cNested_Avg 0.0032
dLgIntContainedDist_Avg 0.0007
Feature
Importance
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01