[Task a. Provide a detailed description of each dataset, their properties and relationships. 2](#_Toc514321256)

[Explore Data 2](#_Toc591042993)

[Task b. Read data from CSV files to the R environment for processing 4](#_Toc846016009)

[Task c: Clean any outliers and exceptional values from the datasets 5](#_Toc950739507)

[Task d: Normalizations, Scaling 5](#_Toc1086171546)

[Task e. Merge Datasets. 6](#_Toc1180406752)

[Task f: Create training and test datasets, if required 9](#_Toc1383177526)

[Specify the Data Splitting Ratio: 9](#_Toc649426226)

[Split Overview 10](#_Toc419463328)

[Task g: Training a model on the data 11](#_Toc166452534)

[Task h: Apply different Machine Learning approaches and discuss 11](#_Toc779059733)

[Clustering Approach 11](#_Toc1181924609)

[Task I: Accuracy of each different model 11](#_Toc745694959)

[Clustering 12](#_Toc431709133)

[Task J: Alternative ways of normalizations, model building, and their performances 12](#_Toc783777722)

[Strengths 12](#_Toc1431821613)

[Limitations 12](#_Toc818500931)

[Uniqueness 12](#_Toc1316051197)

[Task k: Patterns identified and their visualizations 12](#_Toc1549341334)

[Task l: Describe a detailed comparative analysis between the scaling and Machine Learning approaches – strengths, limitations, uniqueness 13](#_Toc1978787797)

[Clustering (K-Means) 13](#_Toc960852220)

[Strengths 13](#_Toc134344456)

[Limitations 13](#_Toc455616634)

[Uniqueness 13](#_Toc1445776794)

[Linear Regression 13](#_Toc78947250)

[Strengths 14](#_Toc1546850632)

[Limitations 14](#_Toc790596227)

[Uniqueness 14](#_Toc632985361)

[Task m: Comparative analysis should be about integration, transformation, visualization and data mining 14](#_Toc1597748576)

[Integration 14](#_Toc946792649)

[Strengths 14](#_Toc1324544)

[Limitations 14](#_Toc1906608428)

[Uniqueness 15](#_Toc833955838)

[Transformation 15](#_Toc1828882815)

[Strengths 15](#_Toc1394686047)

[Limitations 15](#_Toc1146115247)

[Uniqueness 15](#_Toc763336380)

[Visualization: 15](#_Toc1774419272)

[Strengths 15](#_Toc577379462)

[Limitations 15](#_Toc981234877)

[Uniqueness 16](#_Toc420006713)

[Data Mining 16](#_Toc2037755914)

[Strengths 16](#_Toc84010279)

[Limitations 16](#_Toc984564017)

[Uniqueness 16](#_Toc1064260736)

[Task n: Provide a brief discussion about the knowledge gained 16](#_Toc1738095024)

[Understanding of Data Sources 16](#_Toc260795481)

[Data Integration Insights 16](#_Toc776538202)

[Data Transformation 16](#_Toc133769014)

[Visualization Discoveries 16](#_Toc622863706)

[Data Mining and Modeling Insights 17](#_Toc256296050)

[Practical Applications 17](#_Toc1590284225)

# Task a. Provide a detailed description of each dataset, their properties and relationships.

First, the required libraries like readxl, readr, dplyr, tidyverse are loaded.

## Reading files

Path: Planning data is in the ‘Plan’ directory and production data is in the ‘Production Quantities’ directory

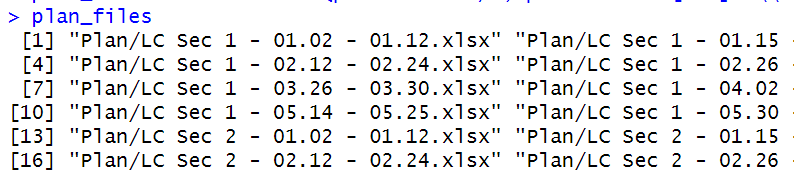


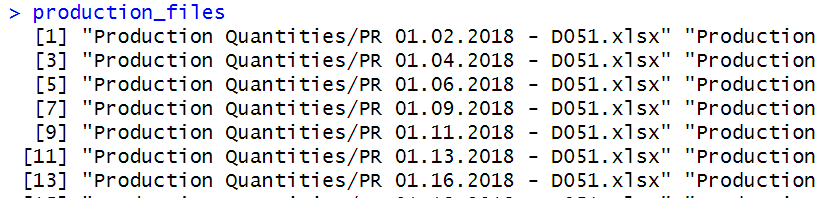


Pattern: This parameter is optional, and a regular expression can be provided. All the .xlsx files are loaded and the files starting with ‘$’ have been excluded. This is due to in Windows file systems, a temporary file is created.

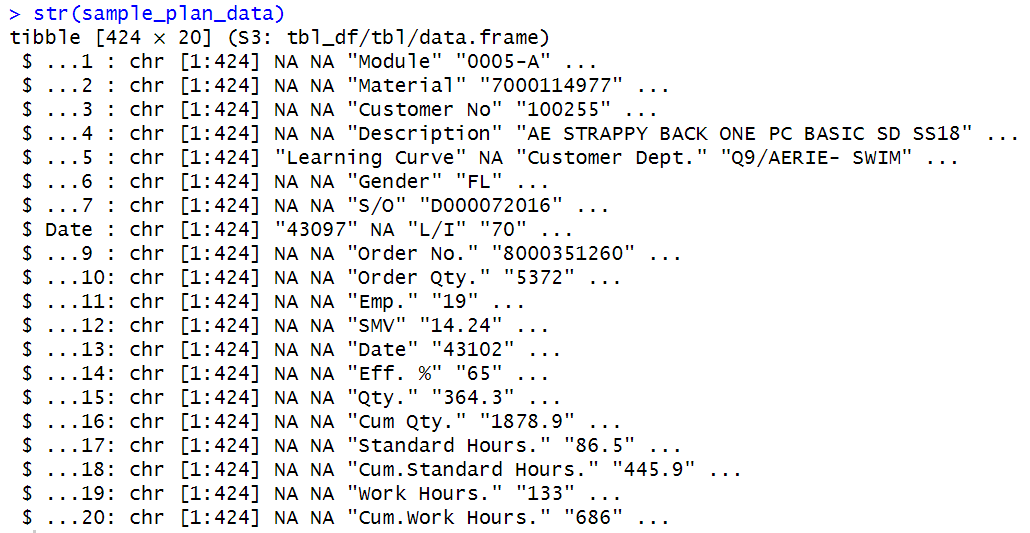
all.files: FALSE means it will load only the names of visible files that are returned. TRUE means all file names will be returned whether it is visible or hidden.

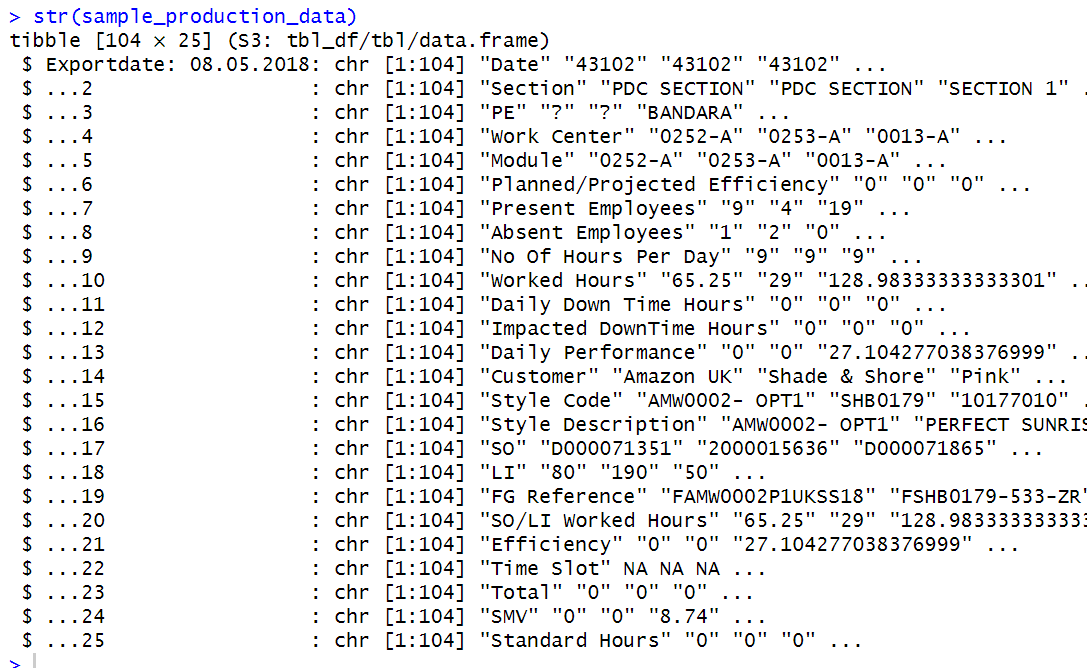
full.names: TRUE means the filename also contains the full path to the directory. FALSE means the filename doesn’t contain the path to the directory.



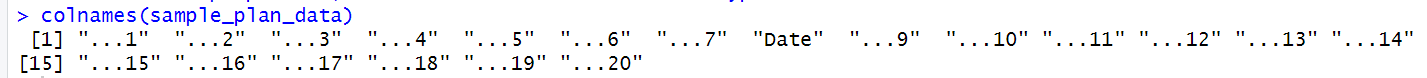


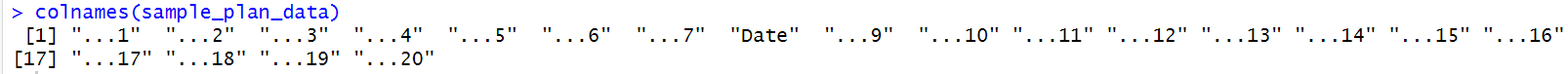
Now a sample from each directory is loaded to explore the structure.





Next, the column names and the data types are examined.



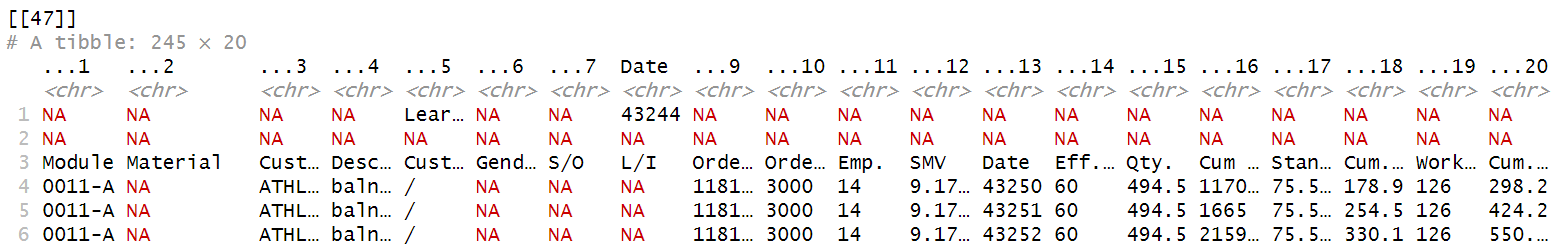


Relationships: Planning dataset has S/O and L/I and these two columns can be linked to the SO and LI in production set.

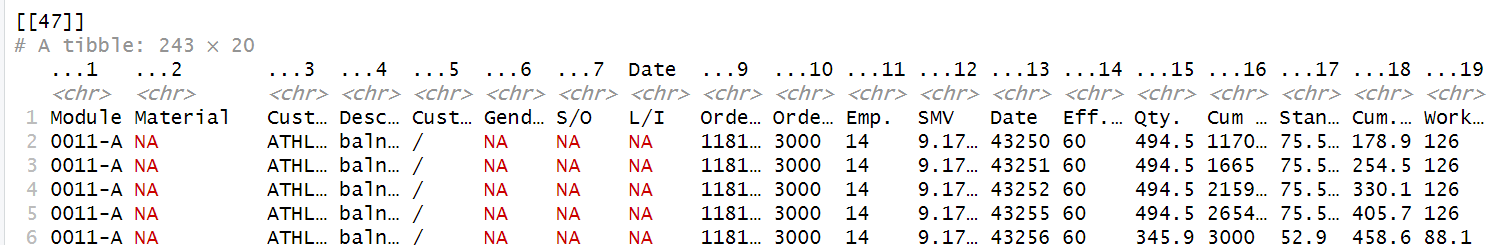
## Simple Explorations

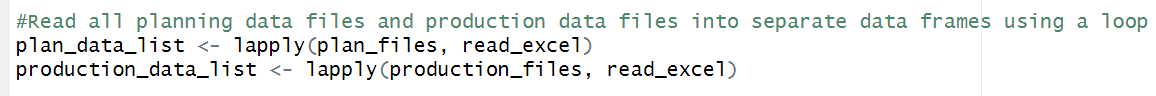
Mean of merged set: 42.00948

# Task b. Read data from CSV files to the R environment for processing

In the planning, files have empty and unwanted rows before the headers. 

A function has been written to remove the unwanted rows before the headers.





Lapply() is used to read list objects and returns a list object of the same length.

# Task c: Clean any outliers and exceptional values from the datasets

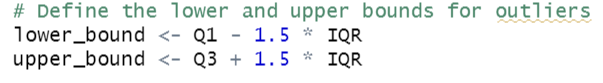
Consideration:

There are two approaches to identifying the outliers. The first method is to combine all the frames of each data set by reading the excels and the next one is to find outliers frame by frame. Here, the second approach is selected because each Excel consists of data from a certain period of the day. If the outliers are identified by looking into the set of combined frames, it will be biased.

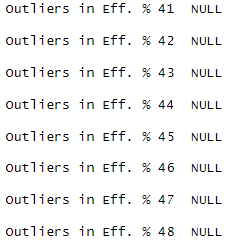
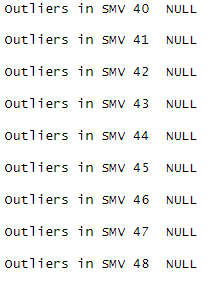
Approaches:

Cleaning outliers is an important step in data preprocessing. Before cleaning outliers, they need to be identified. Common methods for outlier detection include using statistical measures like z-scores or setting an upper bound and a lower bound to IQR. Each frame does not have a large amount of data. Therefore, we have used the second approach.

Setting lower and upper bounds to identify the outliers.



There were no outliers found on SMV and efficiency columns on both production and planning sets.



# Task d: Normalizations, Scaling

The next thing is to identify the columns in the dataset that need normalization or scaling. Typically, numerical features like "SMV", "Efficiency", “Qty”, “Standard Hours”, and “Work Hours” are good candidates.

Considerations:

Efficiency is already scaled to 100%. Therefore, it is not required to normalize.

Approaches:

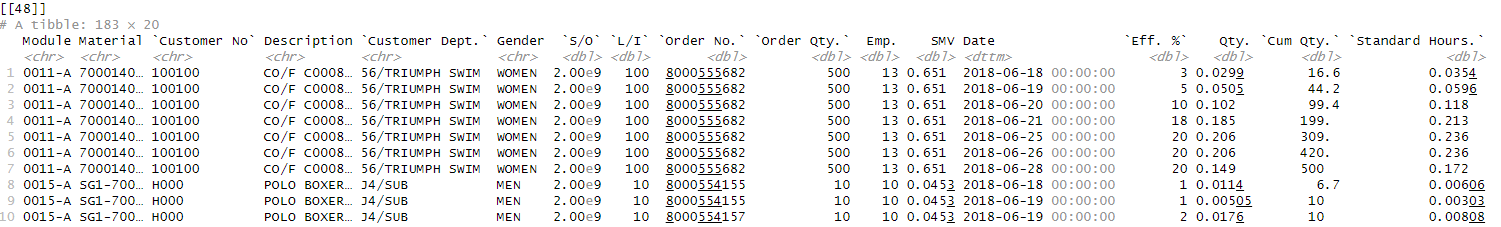
There are several options for normalization and scaling methods. Here are two common methods.

* Min-Max Scaling: Min-max scaling scales the data to a specific range, usually [0, 1].
* Z-Score Standardization: Z-score standardization transforms the data to have a mean of 0 and a standard deviation of 1.

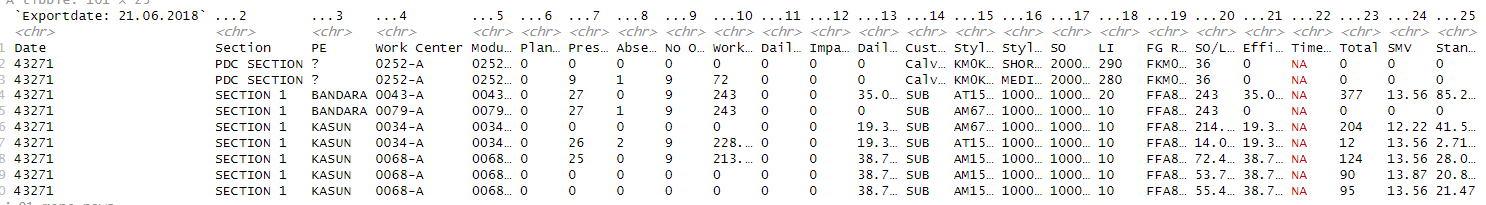
Here, the main max scaling is selected.



A normalized frame in planning looks like this.



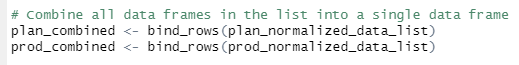
A normalized frame in production looks like this.



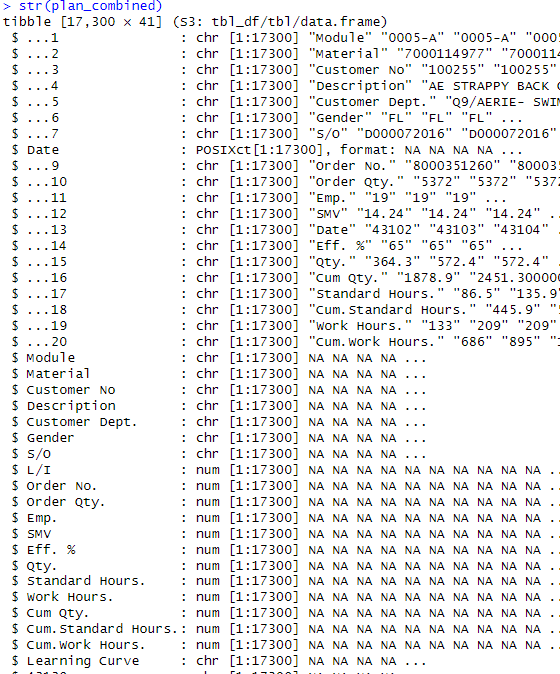
Task e: Merge the datasets

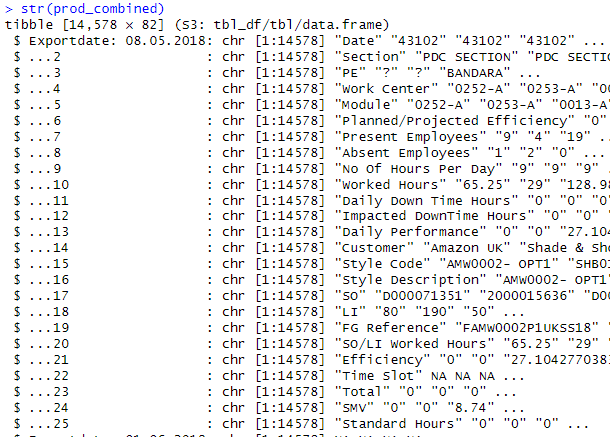
# Task e. Merge Datasets.

Now the planning and production datasets are in frames. These frames need to be combined. First, all the columns are converted into character format. There are three columns in both sets that should be converted otherwise merging is not quite possible. They are the ‘Date’, ‘S/O’, and ‘Material’ columns.

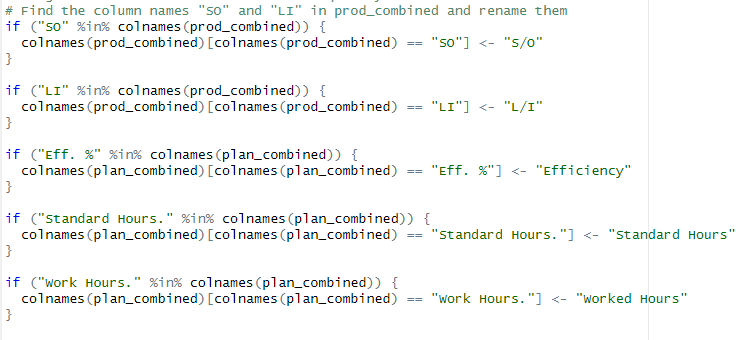
Once they are converted all data frames are combined into a single frame.  


Here is what the combined data looks like.





The column names are altered accordingly in both sets before merging. There we have selected only “S/O”, “L/I”, “Efficiency”, “Standard Hours”, and “Worked Hours”. To keep only the specified columns ("S/O", "L/I", "Efficiency", "Standard Hours", "Worked Hours", and "SMV") from both the plan\_combined and prod\_combined data frames, you can use the select() function from the dplyr package.



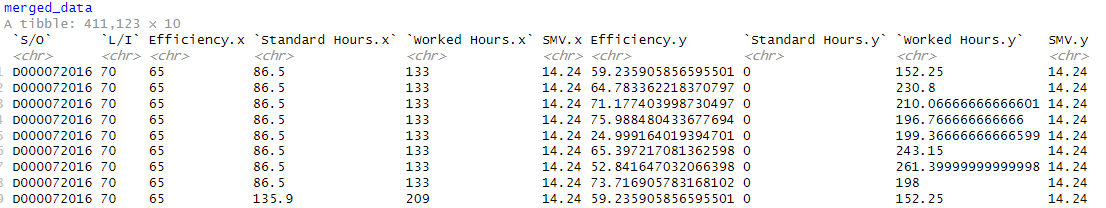
To merge the sets (i.e., plan\_combined and prod\_combined) where you want to combine rows based on a unique key formed by "S/O" and "L/I," you should use an inner join. Here's why:

Inner Join: An inner join combines rows from two data frames based on a common key and includes only the rows where there is a match in both data frames. In your case, you mentioned that "S/O" and "L/I" together act as a unique key that should exist in both data frames. Using an inner join will ensure that only the rows with matching "S/O" and "L/I" values from both datasets are included in the merged result.

For example, if you have a specific sales order ("S/O") and line item ("L/I") in the planning data frame and you want to combine it with the corresponding production data, you need to make sure that there is a matching "S/O" and "L/I" in the production data frame. If there's no match, it means there's no corresponding production data for that planning entry, so it should not be included in the final merged dataset.



The merged dataset looks like this.



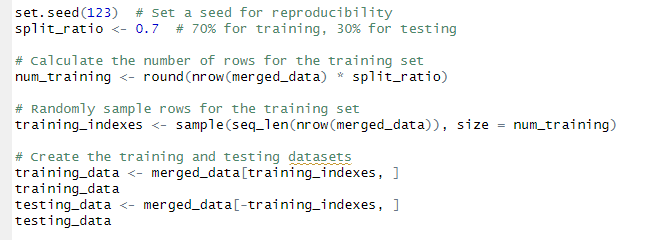
# Task f: Create training and test datasets, if required

involves creating training and testing datasets if they are required for your machine learning analysis. To create training and test datasets, you typically split your data into two parts: one for training your machine learning model and the other for evaluating its performance. In R, you can achieve this using various methods, but one of the most common approaches is to use the sample() function to randomly split the data.

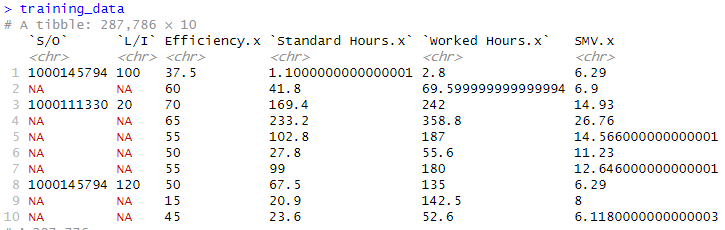
## Specify the Data Splitting Ratio:

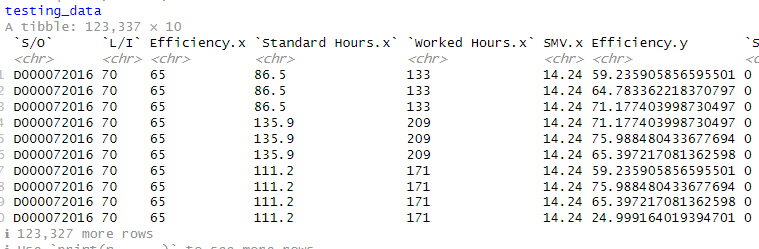


Decide on the ratio in which you want to split your data into training and test sets. A common split is 70-30 or 80-20, where 70% or 80% of the data is used for training, and the rest is used for testing. Here, we have used 70%-30%.



Training and testing data look like this.





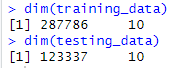
## Split Overview



The dim(training\_data) and dim(testing\_data) functions are used to check and display the dimensions (number of rows and columns) of the training and testing datasets, respectively.

* dim(training\_data) will display the number of rows and columns in the training\_data dataset.
* dim(testing\_data) will display the number of rows and columns in the testing\_data dataset.

This is a common practice when working with datasets to ensure that the data-splitting process has been performed correctly and that you have the expected number of observations in your training and testing sets. Checking the dimensions can help to verify that the data split ratio and random sampling were executed as intended.



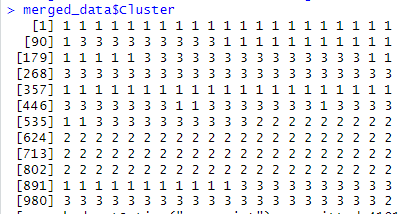
# Task g: Training a model on the data

Selecting an appropriate machine learning algorithm is based on the type of problem we are trying to solve (e.g., regression, classification, clustering) and the nature of the data. In this line, I am using the lm() function to create a linear regression model. The formula Efficiency ~ . specifies that "Efficiency" is the target variable to be predicted, and represents that all other columns in the training\_data dataset are used as predictor variables.

# Task h: Apply different Machine Learning approaches and discuss

## Clustering Approach

Clustering is typically applied to discover patterns and group similar data points together. Linear regression, on the other hand, is a supervised learning method used for predicting continuous numerical values.



Here is how data is scattered.



This line of code fits the linear regression model to the training data (training\_data), making it learn the relationships between the predictor variables and the target variable ("Efficiency").

# Task I: Accuracy of each different model

## Clustering

In clustering, there is no straightforward concept of "accuracy" as would be in supervised learning, where have a known target variable to compare predictions against.

**Silhouette Score**: The silhouette score measures how similar an object is to its cluster (cohesion) compared to other clusters (separation). A higher silhouette score indicates better-defined clusters. Its value ranges from -1 to 1. 1: This means clusters are well apart from each other and distinguished. The silhouette score for this model is 0.69.



# Task J: Alternative ways of normalizations, model building, and their performances

Scaling is another approach to this.

### Strengths

* Normalization of Variables: Scaling, especially using techniques like Z-score scaling or Min-Max scaling, helps ensure that variables are on the same scale. This can be beneficial for machine learning algorithms, as it prevents certain features from dominating others due to differences in magnitude.
* Stability and Convergence: Scaling can improve the stability and convergence of various machine learning algorithms, especially gradient-based optimization methods.

### Limitations

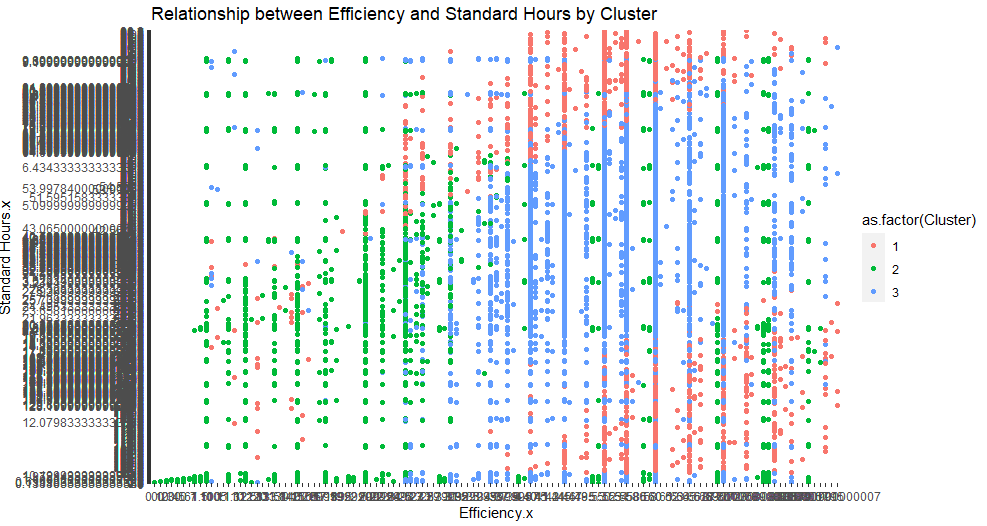
* Loss of Interpretability: Scaling can sometimes make the interpretation of coefficients or feature importance less intuitive, particularly when dealing with Min-Max scaling.
* Not Always Necessary: Scaling may not always be necessary, depending on the algorithm and the data. Some algorithms, like tree-based methods (e.g., decision trees, random forests), are not sensitive to feature scaling.

### Uniqueness

* Scaling is not a machine learning algorithm per se but a preprocessing step. Its uniqueness lies in its role in ensuring that variables are on a consistent scale, which is essential for the proper functioning of many machine learning algorithms.

# Task k: Patterns identified and their visualizations

After clustering, you can create visualizations to explore relationships within clusters. For example, you can create scatterplots or other visualizations that show the relationships between specific pairs of variables within each cluster.



# Task l: Describe a detailed comparative analysis between the scaling and Machine Learning approaches – strengths, limitations, uniqueness

## Clustering (K-Means)

Strengths

* Unsupervised Learning: Clustering is an unsupervised learning technique, making it useful when you don't have labelled data.
* Pattern Discovery: It helps discover hidden patterns or structures in data by grouping similar data points.
* Flexibility: It can handle a wide range of data types and is versatile for various applications, including customer segmentation, anomaly detection, and more.

### Limitations

* Sensitivity to Initializations: K-Means clustering can produce different results based on the initial random cluster centroids. It's important to run it multiple times with different initializations.
* Assumption of Spherical Clusters: K-Means assumes that clusters are spherical and equally sized, which may not always be the case in real-world data.
* The Need to Specify K: You need to specify the number of clusters (K) in advance, which can be challenging if you don't have prior knowledge of the data's underlying structure.

### Uniqueness

* Clustering is a unique machine machine-learning that focuses on grouping data points based on their similarity or distance from each other, without relying on a target variable for guidance.

## Linear Regression

### Strengths

* Interpretability: Linear regression models provide straightforward interpretability. You can analyze coefficients to understand the relationships between independent variables and the target.
* Predictive Power: Linear regression can work well when the relationship between independent and dependent variables is approximately linear.
* Easily Extendable: It serves as a foundation for more complex regression techniques, allowing for easy extension to handle non-linear relationships (e.g., polynomial regression, ridge regression, or lasso regression).

### Limitations

* Assumptions: Linear regression assumes a linear relationship between predictors and the target, independence of errors, homoscedasticity (constant variance), and normally distributed errors. Violation of these assumptions can lead to inaccurate results.
* Overfitting: In cases where the true relationship is not linear, linear regression models can lead to overfitting if not properly regularized.
* Sensitivity to Outliers: Linear regression can be sensitive to outliers, affecting model performance.

### Uniqueness

* Linear regression is a well-established supervised learning algorithm used primarily for predicting continuous numeric values. Its uniqueness lies in its simplicity, interpretability, and foundational role in regression analysis.
* In summary, each approach—scaling, clustering, and linear regression—has its strengths, limitations, and uniqueness. The choice of which approach to use depends on the specific goals of your analysis, the nature of your data, and the insights you seek to gain. It's often beneficial to explore multiple approaches and select the one(s) that best aligns with your objectives and the characteristics of your dataset.

# Task m: Comparative analysis should be about integration, transformation, visualization and data mining

## Integration

### Strengths

* Data Consolidation: Integration allows you to combine data from multiple sources into a unified dataset. This is useful for gaining a holistic view of the production planning process.
* Improved Data Quality: Integrating data can help identify and rectify data quality issues, such as missing values or inconsistencies, during the consolidation process.
* Enhanced Analysis: A single integrated dataset simplifies subsequent analysis, making it easier to derive insights and patterns.

### Limitations

* Data Compatibility: Integrating data from diverse sources may require dealing with varying data formats, structures, and quality. Ensuring compatibility can be challenging.
* Data Privacy and Security: Combining data sources may raise privacy and security concerns, especially when dealing with sensitive information.

### Uniqueness

* Data integration is a critical data preprocessing step that enables comprehensive analysis by bringing together data from disparate sources into a unified format.

## Transformation

### Strengths

* Data Enhancement: Transformation allows you to enhance data quality by addressing issues like missing values, outliers, and inconsistencies.
* Feature Engineering: You can create new features or modify existing ones to better represent the underlying patterns in your data.
* Normalization: Scaling and normalization transform data to a consistent scale, which is essential for certain machine learning algorithms.

### Limitations

* Subjectivity: Decisions regarding how to handle missing data or outliers can be subjective and may impact the analysis outcome.
* Information Loss: Aggressive transformations or feature selection may lead to information loss if relevant data is discarded.
* Computational Cost: Complex transformations can be computationally expensive, particularly for large datasets.

### Uniqueness

* Data transformation is a preprocessing step that focuses on enhancing data quality and preparing it for analysis. It involves various techniques like imputation, scaling, encoding, and feature engineering.

## Visualization:

### Strengths

* Insight Generation: Visualization provides a powerful way to explore data, identify trends, and gain insights that may not be evident through numerical analysis alone.
* Communication: Visualizations are an effective means of communicating complex findings to stakeholders and decision-makers.
* Exploratory Data Analysis: Visualization aids in exploratory data analysis by helping you understand data distributions and relationships.

### Limitations

* Subjectivity: Interpretation of visualizations can be subjective, and different individuals may derive varying insights from the same visualization.
* Data Complexity: For high-dimensional data, creating informative visualizations can be challenging, as it may require reducing dimensionality through techniques like dimensionality reduction.

### Uniqueness

* Visualization is a crucial tool for understanding data and presenting findings. It offers a unique way to explore and communicate complex information effectively.

## Data Mining

### Strengths

* Pattern Discovery: Data mining techniques, including clustering, regression, and classification, enable the discovery of patterns and relationships in data.
* Prediction: Data mining can be used for predictive modelling, allowing you to forecast future production quantities or make data-driven decisions.
* Automation: Many data mining algorithms can automatically uncover insights from data, reducing the need for manual analysis.

### Limitations

* Model Assumptions: Data mining models often make assumptions about the data, and if these assumptions are violated, the model's accuracy may suffer.
* Overfitting: Complex data mining models can overfit the training data, leading to poor generalization of new data.
* Data Preprocessing: Data mining success heavily depends on data preprocessing steps such as cleaning, integration, and transformation.

Uniqueness

* Data mining encompasses a wide range of techniques for extracting valuable information from data, making predictions, and uncovering hidden patterns. Its uniqueness lies in its ability to automate knowledge discovery.

# Task n: Provide a brief discussion about the knowledge gained

## Understanding of Data Sources

In this project, a deep understanding of diverse data sources was gained, including planning data and actual production records. These sources presented integration and analysis challenges.

### Data Integration Insights

Data integration played a pivotal role in merging disparate sources into a coherent dataset, enabling a comprehensive view of production planning processes.

## Data Transformation

Data transformation enhanced data quality by addressing missing values, outliers, and scaling variables. Feature engineering improved the dataset's suitability for analysis.

## Visualization Discoveries

Visualization uncovered hidden patterns and trends in the data, revealing insights such as cyclical production patterns and variable correlations.

## Data Mining and Modeling Insights

Data mining techniques, including clustering and linear regression, provided valuable insights, aiding in the identification of production patterns and quantifying relationships between variables.

## Practical Applications

The project's knowledge has practical applications, including enhanced production forecasting accuracy, more efficient resource allocation, reduced lead times, cost savings, and improved strategic decision-making in production planning processes.