DMBA ASSIGNMENT 1

AY 2023/2024 OCT Semester

Declaration of Originality

I am the originator of this work and I have appropriately acknowledged all other original sources used as my reference for this work.

I understand that Plagiarism is the act of taking and using the whole or any part of another person's work, including work generated by AI, and presenting it as my own.

I understand that Plagiarism is an academic offence and if I am found to have committed or abetted the offence of plagiarism in relation to this submitted work, disciplinary action will be enforced.

☑ I Agree

My Information

Name (as in matriculation card)	JAVEN LAI LE YU
Admin Number	2202934B
Practical Group (e.g. P01)	P03
I am submitting _____ level work	Advanced ▾

Background and Purpose

In the current era of heightened environmental awareness due to climate concerns, businesses are increasingly adopting sustainable practices to minimize their ecological footprint to enhance their reputation with consumers and stakeholders, thereby giving rise to the concept of Corporate Sustainability.

This project focuses on developing predictive models to understand the determinants influencing a company's likelihood in achieving sustainability goals. Unlike conventional existing approaches like merely comply with environmental regulations, this project aims to provide a more comprehensive study by integrating Environmental, Social, and Governance (ESG) principles, as well as green and socially responsible practices in daily operations, to delve deep into discovering overarching insights that could help companies achieve their sustainability goals within the next 5 years.

The dataset provided comprises 20 variables and 10,000 records, encompassing various environmental, management, and stakeholder engagement metrics related to corporate sustainability. The target variable is the "Likelihood of Achieving Sustainability Goals" – a score between 0 and 1.

Problem Statement

Build Predictive Models to understand how factors influence a company's likelihood of achieving their sustainability goals, to construct strategies that can be backed by data to be successful if implemented to **help companies achieve their sustainability goals on schedule.**

Stakeholders of Analysis

Businesses

Companies aiming to achieve their sustainability goals can leverage insights from this project to enhance their practices.

Government and Sustainability Organizations

Entities responsible for regulating and promoting sustainability can use the project's recommendations to enforce robust regulations as industry standards, ensuring companies operate sustainably.

Objectives

From this project, stakeholders can gain:

- Interesting insights about the dataset provided.
- Information on Corporate Sustainability and how it can be incorporated into machine learning.
- Understanding into various predictive models, how to interpret their results, and how the models work.
- Insights on what factors affect a company's likelihood of achieving sustainability goals and how these factors influence the likelihood.
- A chosen predictive model that can be used to determine if a company is likely to achieve their sustainability goals.
- Data-driven recommendations and strategies for a company to increase the likelihood of achieving their sustainability goals.

Data Exploration and Data Pre-processing

Data Dictionary

Field	Description
Record ID	A unique identifier for each record.
Company Name	The name of the company.
Industry	The industry to which the company belongs.
Carbon Emissions (MT)	Annual carbon emissions in metric tons.
Water Usage (m^3)	Annual water usage in cubic meters.
Energy Consumption (MWh)	Annual energy consumption in megawatt-hours.
Waste Generation (MT)	Annual waste generation in metric tons.
Recycling Rate (%)	The percentage of waste that is recycled.
Supply Chain Sustainability Score	A score assessing the sustainability of the company's suppliers.
Employee Transportation Mode	The most common mode of transportation used by employees.
Renewable Energy Use (%)	The percentage of energy consumption from renewable sources.
Sustainability Training	Whether the company provides sustainability training to its employees.
Community Engagement Score	A score indicating the company's engagement with the community on sustainability efforts.
Sustainability Reporting Frequency	The frequency of the company's sustainability reporting.
Eco-friendly Product Lines (%)	The percentage of the company's product lines that are eco-friendly.
Investment in Sustainability (USD)	Annual investment in sustainability projects.
Third-party Sustainability Certification	Whether the company has third-party sustainability certification.
Customer Feedback Score on Sustainability	A score reflecting customer perception of the company's sustainability efforts.
Sustainable Packaging Initiatives	The number of initiatives taken for sustainable packaging.
Likelihood of Achieving Sustainability Goals (TARGET)	A score between 0 to 1 indicating the likelihood of achieving sustainability goals in the next 5 years.

Loading and Examining Data

I have chosen dataset 1 as it is cleaner and has lesser missing values than dataset 2, indicating better quality and reliability.

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

TARGET, file = 'Likelihood of Achieving Sustainability Goals','sustainability_data_set1.csv' # define target, dataset
data = pd.read_csv(file)
```

```python
len(data) # number of rows
```

```
10100
```

```python
data.head() # preview dataset
```

	Record ID	Company Name	Industry	Carbon Emissions (MT)	Water Usage (m^3)	Energy Consumption (MWh)	Waste Generation (MT)	Recycling Rate (%)	Supply Chain Sustainability Score	Employee Transportation Mode	Renewable Energy Use (%)	Sustainability Training	Comm Engage S
0	1.0	Company_1	IT	41648.0	165790.0	41851.0	11163.0	65.846445	55.0	Car	17.492863	Yes	
1	2.0	Company_2	Automobile	40231.0	593567.0	2192.0	741.0	52.904042	90.0	Car	28.862218	Yes	
2	3.0	Company_3	IT	3973.0	450035.0	34379.0	9787.0	27.059690	93.0	Car	15.218815	No	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	5.0	Company_5	Electronics	28881.0	148136.0	6072.0	19969.0	52.535200	44.0	Public Transport	62.066153	No	
10098	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
10099	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N

Analysis:

1. Company Name is created using Company_ + Record ID.
2. There are rows with missing value for every column, likely caused by data logging error where empty rows are created. I must ensure there are no missing values in the dataset as missing values prevent me from using the dataset for most machine learning models.
3. Non-terminating values for percentage columns such as Recycling Rate (%), Renewable Energy Use (%), Eco-friendly Product Lines (%), and Likelihood of Achieving Sustainability Goals.

Action:

I will start by dropping all rows with missing value for the Target column since Target should not be imputed (never impute target for machine learning to ensure ground truth; ensure integrity and accuracy of Target value), hence the missing Target cannot be treated and these rows are unusable.

```
data = data.dropna(subset=[TARGET]) # drop rows with missing Target value as I can't impute Target
```

```
data.isna().sum().sum()          # check for missing values
```
```
0
```

All missing values are gone. This implies that all rows with missing values are problematic since they are missing the target, hence its justified to drop them.

```
len(data)                        # number of rows after cleaning
```
```
9500
```

600 faulty rows dropped due to missing Target. There are no missing values in the dataset now, implying that all the rows dropped were the empty rows discovered earlier.

Next, I rounded off all columns to 2 d.p. for conventional interpretability of values:

Rounding off percentage columns to 2 d.p.

```
data = data.round(2) # round values to 2d.p.
data.head()
```

pply Chain stainability Score	Employee Transportation Mode	Renewable Energy Use (%)	Sustainability Training	Community Engagement Score	Sustainability Reporting Frequency	Eco-friendly Product Lines (%)	Investment in Sustainability (USD)	Third-party Sustainability Certification	Customer Feedback Score on Sustainability	Sustainable Packaging Initiatives	Likelihood of Achieving Sustainability Goals
55.0	Car	17.49	Yes	75.0	Bi-annually	35.94	1649580.0	Yes	67.0	4.0	0.69
90.0	Car	28.86	Yes	92.0	Bi-annually	50.70	3152458.0	Yes	95.0	5.0	0.14
93.0	Car	15.22	No	38.0	Quarterly	13.79	4389876.0	No	54.0	6.0	0.87
44.0	Public Transport	62.07	No	82.0	Quarterly	78.71	1463022.0	No	60.0	4.0	0.51
68.0	Public Transport	61.09	No	49.0	Quarterly	65.22	4879181.0	No	60.0	3.0	0.79

Check for inconsistencies and anomalies

Inconsistencies

```
data.select_dtypes(include=['object']).info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 9500 entries, 0 to 9999
Data columns (total 6 columns):
 #   Column                                    Non-Null Count  Dtype
---  ------                                    --------------  -----
 0   Company Name                              9500 non-null   object
 1   Industry                                  9500 non-null   object
 2   Employee Transportation Mode              9500 non-null   object
 3   Sustainability Training                   9500 non-null   object
 4   Sustainability Reporting Frequency        9500 non-null   object
 5   Third-party Sustainability Certification  9500 non-null   object
dtypes: object(6)
memory usage: 519.5+ KB
```

```
data.nunique()

Record ID                                    9500
Company Name                                 9500
Industry                                        8
Carbon Emissions (MT)                        8543
Water Usage (m^3)                            9412
Energy Consumption (Mwh)                     8628
Waste Generation (MT)                        7884
Recycling Rate (%)                           9500
Supply Chain Sustainability Score              60
Employee Transportation Mode                    4
Renewable Energy Use (%)                     9500
Sustainability Training                         2
Community Engagement Score                     70
Sustainability Reporting Frequency              3
Eco-Friendly Product Lines (%)               9500
Investment in Sustainability (USD)           9492
Third-party Sustainability Certification        2
Customer Feedback Score on Sustainability      50
Sustainable Packaging Initiatives              12
Likelihood of Achieving Sustainability Goals 9500
dtype: int64
```

I need to inspect the values of each column to rectify any problems like inconsistencies or anomalous values to ensure data is truly clean.

I identify categorical columns by finding columns with Object data type. I will investigate categorical columns to identify inconsistencies or issues by checking each unique value in the column, and their frequency.

I will investigate the following columns as they are categorical and I have yet to understand their values:

- Industry
- Employee Transportation Mode
- Sustainability Training
- Sustainability Reporting Frequency
- Third-party Sustainability Certification

Industry:

```
data.Industry.value_counts()
```

```
Pharmaceutical     1634
Automobile         1591
Textile            1588
Food & Beverages   1568
Electronics        1552
IT                 1480
Finannce             49
Healtchare           38
Name: Industry, dtype: int64
```

- Misspelling in Finance and Healthcare industry. Should be corrected for clarity of understanding class.
- Healthcare and Finance have extremely few records compared to other industries (severe imbalance).

Having misspelling and severe class imbalance makes these records' credibility suspicious. Could Healthcare belong to Pharmaceutical and was accidentally separated? Research from several articles like efpia and BIG Language Solutions revealed that these 2 industries are separate, hence I will not combine them.

Investigating if these classes are anomalous:

```
data[data['Industry'] == 'Finannce'].head(2)
```

	Record ID	Company Name	Industry	Carbon Emissions (MT)	Water Usage (m^3)	Energy Consumption (MWh)	Waste Generation (MT)	Recycling Rate (%)	Supply Chain Sustainability Score	Employee Transportation Mode	Renewable Energy Use (%)	Sustainability Training	C Eng
166	167.0	Company_167	Finannce	10020.0	-1000000.0	6460.0	20143.0	64.65	82.0	Public Transport	30.03	No	
363	364.0	Company_364	Finannce	26332.0	-1000000.0	12943.0	21622.0	53.70	66.0	Car	26.66	Yes	

The water usage column values are all ERRORS since a company can't have negative water usage.

```
data[data['Industry'] == 'Healtchare'].head(2)
```

	Record ID	Company Name	Industry	Carbon Emissions (MT)	Water Usage (m^3)	Energy Consumption (MWh)	Waste Generation (MT)	Recycling Rate (%)	Supply Chain Sustainability Score	Employee Transportation Mode	Renewable Energy Use (%)	Sustainability Training	E
1921	1922.0	Company_1922	Healthcare	1000000.0	118745.0	19214.0	19333.0	39.16	64.0	Car	77.13	Yes	
2122	2123.0	Company_2123	Healthcare	1000000.0	970092.0	44243.0	11221.0	60.58	72.0	Walking	67.40	No	

Carbon emission outliers are ERRORS since 1,000,000 metric tons of carbon emissions is impossible as other companies do not exceed 10,000 metric tons.

All Finance and Healthcare rows have the same ilogical value for the highlighted columns. This article shows that 20 oil companies collectively produced a total of 1.35 million **MtCO$_2$e** carbon emissions during 1965 - 2017, suggesting that it's improbable for 1 company alone to produce 1 million MT of carbon emission annually.

Conclusion: Drop Finance and Healthcare Industries

```
# drop Industries Finannce and Healtchare
data = data[data['Industry'] != 'Finannce']
data = data[data['Industry'] != 'Healtchare']
```

Since misspelling, severe class imbalance, and identical errors were found in same column for all rows in these 2 industries, I decided to drop them as having multiple issues suggest a lack of veracity in these records, making it justified to drop them so that the accuracy and reliability of my dataset and analysis will not be compromised. It can be speculated that these rows are adjustments, dummy rows that do not belong in the dataset, or rows that were tampered with. These rows are only 0.9% of the dataset (insignificant), and imputing these faulty columns will introduce ambiguities since assuming the values introduces biases (imputing my own pattern) or skewness (distort true distribution of column). Additionally, it's impossible to 'guess' the actual true value. Hence, keeping these rows by imputing these faulty columns could taint the accuracy and reliability of the analysis.

Employee Transportation Mode, Sustainability Training and Reporting Frequency:

Transportation Mode

```
data['Employee Transportation Mode'].value_counts()
```
```
Bike               2387
Car                2378
Public Transport   2332
Walking            2316
Name: Employee Transportation Mode, dtype: int64
```

No inconsistency in spelling or class imbalance issues.

Sustainability training and sustainability reporting frequency

```
data['Sustainability Training'].value_counts()
```
```
Yes    5742
No     3671
Name: Sustainability Training, dtype: int64
```

```
data['Sustainability Reporting Frequency'].value_counts()
```
```
Bi-annually   3190
Annually      3129
Quarterly     3094
Name: Sustainability Reporting Frequency, dtype: int64
```

No inconsistency in spelling or class imbalance issues.

No inconsistency issues like misspelling or illogical values that are implausible found in any other categorical columns. No adjustments needed.

Investigating numerical columns using statistical analysis

**Summary Statistics** (overview of each column) to identify outliers:

```
data.describe().loc[['mean', 'std', 'min', '50%', 'max']]
```

	Record ID	Carbon Emissions (MT)	Water Usage (m^3)	Energy Consumption (MWh)	Waste Generation (MT)	Recycling Rate (%)	Supply Chain Sustainability Score	Renewable Energy Use (%)	Community Engagement Score	Eco-friendly Product Lines (%)	Investment in Sustainability (USD)	Customer Feedback Score on Sustainability	Sustainable Packaging Initiatives	Likelihood of Achieving Sustainability Goals
mean	5001.032508	25333.665038	504106.357909	25693.943376	12776.770318	49.997823	69.532136	44.860917	64.471051	44.629453	2.521404e+06	74.619462	2.978965	0.499767
std	2889.025368	14193.799588	284181.837977	14171.286072	7093.048608	17.438848	17.294060	20.186332	20.193500	20.171707	1.428523e+06	14.413830	1.728423	0.286922
min	1.000000	1000.000000	10028.000000	1008.000000	501.000000	20.000000	40.000000	10.020000	30.000000	10.010000	5.046000e+04	50.000000	0.000000	0.000000
50%	4999.000000	25178.000000	509034.000000	25857.000000	12751.000000	50.030000	70.000000	44.510000	65.000000	44.420000	2.508142e+06	75.000000	3.000000	0.500000
max	10000.000000	49996.000000	999841.000000	49993.000000	24996.000000	80.000000	99.000000	79.990000	99.000000	79.970000	4.999733e+06	99.000000	11.000000	1.000000

Noticed that Sustainable Packaging Initiatives could have outliers as the max is far from median. I will delve deeper to rectify whether these outliers are anomalies that should be removed. No anomalous values (e.g negative spending) in any other columns since every value is plausible and makes sense.

Investigating Sustainable Packaging Initiatives:

```
# vizualize distribution of Sustainable Packaging Initiatives
fig, ax = plt.subplots(figsize=(20, 5))
sns.boxplot(x='Sustainable Packaging Initiatives', data=data, ax=ax, palette='viridis')

<AxesSubplot:xlabel='Sustainable Packaging Initiatives'>
```
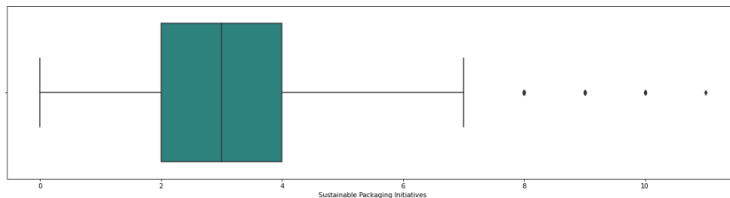
```
data['Sustainable Packaging Initiatives'].value_counts()

2.0     2170
3.0     2139
4.0     1554
1.0     1335
5.0      930
0.0      517
6.0      456
7.0      198
8.0       76
9.0       28
10.0       9
11.0       1
Name: Sustainable Packaging Initiatives, dtype: int64
```

Logically, it's possible for companies to have a few more initiatives than others. The outliers here are not as drastically extreme where it becomes unrealistic. Hence, I will proceed with the assumption that these companies are **not anomalous** and just happen to be outliers because few companies push out as many Sustainable Packaging Initiatives. Thus, I do not drop outliers if the values are accurate.

Check for duplicated rows:

```
In [24]: # check if any companies have identical characteristics
         subset_columns = data.columns[~data.columns.isin(['Record ID', 'Company Name'])]
```

```
In [25]: data.duplicated(subset=subset_columns).sum()

Out[25]: 0
```

I checked for duplicate records (companies with identical characteristics) by excluding unique identifiers like "Record ID" and "Company Name" and compared the remaining features across rows to see if there are companies with identical sets of features. There are **no duplicated records**.

Creating Nominal Target to enable usage of Classification Models:

Binned Target into 3 segments: Low, Medium, and High. This approach clusters likelihood into a meaningful class that can be easily understood by business user (e.g. More intuitive to directly tell a company they have a high likelihood instead of 0.72 likelihood as they do not need to interpret the number to know their company is on the right track). While improving interpretability of target, binning the target also enables the dataset to be used for Classification that directly identifies whether a company is likely/unlikely to achieve their sustainability goals.

```
# Add a new column for segmentation based on Likelihood of Achieving Sustainability Goals
data['Sustainability_Goals_Segment'] = pd.cut(data['Likelihood of Achieving Sustainability Goals'],
                                    bins=[-1, 0.4, 0.6, 1], labels=['Low', 'Medium', 'High'])
                                    # start -1 as it has to be smaller than 0 for binning to work
```

Concluding Data Cleaning:

Data is now clean with 9413 rows remaining as I have treated missing values, rectified and dropped anomalous values that are implausible, and ensured there are no duplicate records in the dataset.

```
len(data) # number of rows in dataset

9413
```

Before data modelling, I must understand how the data relates to sustainability so that the models I built are relevant and useful in solving the problem statement. Hence, I will perform features engineering and selection.

Features Engineering

To make the data meaningful for machine learning to understand how sustainability goals can be achieved in the next 5 years, I conducted research on Corporate Sustainability to comprehend the essence of this concept so that I can build purposeful models using relevant features that offer meaningful insights to help companies achieve their sustainability goals.

Corporate Sustainability:

According to an article by Forbes, Corporate Sustainability involves a commitment by companies to address Environmental, Social, and Governance sustainability considerations in their operations with the aim of achieving long-term goals aligned with global sustainability objectives. With this knowledge, it's logical to assume that the 'Sustainability Goals' set by the companies revolve around relevant ESG considerations. Therefore, I should focus on including features related to ESG in my predictive models to improve the usefulness and performance.

Based on VelocityEHS, ESG is defined as such:

Environmental	Social	Governance
The E of ESG speaks to a company's impact on the natural world. It covers a wide-range of activities and sustainability concerns, including: • Greenhouse Gas Emissions (GHS) • Air Emissions & Carbon Monitoring • Energy & Utility Usage Tracking • Waste Management • Water Quality • Environmental Reporting (e.g., TRI/Form R)	The S of ESG focuses on how an organization treats people and operates as a member of communities and supply chains. It includes: • Health, Safety & Social Sustainability • Employee Engagement • Diversity, Equity & Inclusion • Privacy, Data Protection & Cyber Security • Product Safety & Stewardship • Labor Standards & Human Rights	The G of ESG covers how well a company is managed and how well it abides by ethical practices at all levels, including items such as: • Business Ethics • Risk Mitigation & Risk Governance • Regulatory Compliance • Tax Transparency • Shareholder Rights • Board Composition & Executive Compensation

Here's an analysis of relevant features in the dataset that can be made to align with ESG pillars.

Environmental:

The dataset already contains useful features like carbon emissions, water usage, energy consumption, and waste generation of a company. Additionally, we can assess a company's actions and commitment towards sustainability by evaluating their recycling rate, renewable energy usage, and percentage of eco-friendly product lines. According to the Forbes article above, the main factors evaluated are greenhouse gases like carbon emission, and how well a company embraces renewable energy. Hence, I could **create a feature that calculates the amount of unrenewable energy used** by the company to assess the company's sustainability instead of calculating amount of renewable energy used since it's more meaningful to measure Renewable energy Use (%).

```
# Calculate the amount of unrenewable energy (in MWh)
df['Unrenewable Energy (MWh)'] = df['Energy Consumption (MWh)'] * (100 - df['Renewable Energy Use (%)']) / 100
```

Social:

We can address the Social Domain using Customer Feedback Score and Community Engagement Score. However, as there are already many Environmental features and having too many features can hinder model performance, I decided to engineer 1 feature to get an overview of a company's social impact. Hence, I created a new feature that utilizes both Customer Feedback and Community Engagement to assess how effective the community engagement efforts carried out by the company impacted customers' impression of the company.

Summarize 2 columns into 1 to reduce dimensionality while making a new meaningful feature to assess the impact of social impacts.

```
# Create the Engagement Impact Score
df['Engagement Impact Score'] = round(df['Customer Feedback Score on Sustainability'] / df['Community Engagement Score']
                                      , 2) * 100 # round to 2d.p. so that values are to nearest whole number
```

```
# vizualize distribution of Engagement Impact Score to check for outliers and illogical values like negative values
fig, ax = plt.subplots(figsize=(20, 5))
sns.boxplot(x='Engagement Impact Score', data=df, ax=ax, palette='viridis')
```

```
<AxesSubplot:xlabel='Engagement Impact Score'>
```
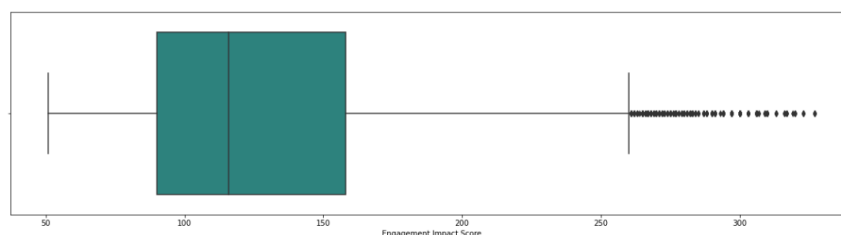
Governance:

Are government/environmental organisations taking any actions to enforce sustainability practices on companies? Based on weforum, taxes are imposed to regulate carbon emissions and energy consumption. With this knowledge, I will find out the tax rates imposed, to integrate into the dataset.

Carbon emission:

According to the National Environment Agency (NEA), Singapore imposed a carbon tax of $5/tCO2e from 2019-2023. With this info, I will calculate the tax each company paid for their carbon emission.

Energy Consumption:

After thorough research, I found a dataset from Eurostat [Download>Full dataset] that provides information on tax percentages on energy consumption from 1995 to 2021 for the European Union (EU) and other countries, categorized by various industry sectors. Due to limited information available, I can only make use of Industry and Services [Sheet 6,7] sectors for my project, and I also have to proceed with the **assumption that the sustainability dataset is from 2021, and the EU Tax regulations apply** to the mentioned sectors.

Preview of Dataset in Excel:

Data extracted on 03/01/2024 12:32:05 from [ESTAT]
Dataset: Energy taxes by paying sector [env_ac_taxener]
Last updated:

Time frequency: Annual
Unit of measure: Percentage
Industry (except construction)

TIME	2015	2016	2017	2018	2019	2020	2021
GEO (Labels)							
European Union - 27 countries (from 2020)	17.46	16.74	16.92	18.05	19.13	19.87	22.18
Belgium	14.33	16.17	17.37	17.31	18.56	19.56	18.93
Bulgaria	17.26	21.34	19.07	21.5	30.3	34.16	32.48
Czechia	24.99	24.2	24.04	25.27	31.26	32.43	34.69

[Sheet 6, showing 22.18% tax on energy consumption for all industrial industries during 2021 under EU]

Data extracted on 03/01/2024 12:32:05 from [ESTAT]
Dataset:
Last updated:

Time frequency: Annual
Unit of measure: Percentage
Services (except wholesale and retail trade, transportation and storage)

TIME	2014	2015	2016	2017	2018	2019	2020	2021
GEO (Labels)								
European Union - 27 countries (from 2020)	10.29	10.17	10.43	10.44	10.22	10.14	9.96	9.79
Belgium	12.48	13.7	15.27	16.52	15.91	16.66	16.39	16.59
Bulgaria	2.61	2.72	2.58	2.71	2.6	2.26	2.11	2.13
Czechia	8.07	8.36	8.87	8.88	8.6	7.88	7.96	7.23

[Sheet 7, showing 9.79% tax on energy consumption for all service industries during 2021 under EU]

Integrating Taxed amount in dataset:

Feature 2: Governance

Include Taxes for carbon emissions and unrenewable energy usage.

```python
# Define tax rates of energy consumption for each industry
tax_rates = {
    # SERVICE:
    'Food & Beverages': 9.79,

    # INDUSTRY:
    'Electronics': 22.18,
    'IT': 22.18,
    'Pharmaceutical': 22.18,
    'Automobile': 22.18,
    'Textile': 22.18
}

# Apply tax rates based on the 'Industry' column
df['Tax Percentage for Energy'] = df['Industry'].map(tax_rates) # energy tax
df['Tax Percentage for Carbon'] = 5                             # carbon tax
```

```python
# Calculate taxed amount
df['Energy Tax'] = df['Unrenewable Energy (MWh)'] * df['Tax Percentage for Energy'] / 100
df['Carbon Tax'] = df['Carbon Emissions (MT)'] * df['Tax Percentage for Carbon'] / 100
```

Energy tax rates vary between service industries and manufacturing industries based on Eurostat's dataset. I classified the industries into service or industry using contextual knowledge.

Tax on energy only applies on unrenewable energy, since it's logical to **assume companies will not get taxed on the electricity usage if they make a conscious effort to go sustainable**. Carbon Emission receives $5 tax for each metric Tonne emitted.

Although all ESG Pillars have been addressed, the features could still benefit from further improvements.

Balancing comparison between various industries:

Various **industries may inherently require different levels of resources based on the nature of their operations**, making direct comparisons of carbon emissions, water usage, energy usage, and waste generation across diverse sectors inequitable. Hence, I will **represent the company's emissions as a percentage relative to the industry average**. Similarly for tax, renewable energy percentage, recycling rate percentage, and Sustainable packaging initiatives, I will express these values as the difference compared to the industry average values respectively. This enables a nuanced examination of how usage/emission of carbon, water, energy, waste, and efforts to reduce impacts relative to the industry's average influences a company's ability to achieve its sustainability goals. For investment in sustainability, I assume that it doesn't vary among different industries, so it's simply compared with the average investment amount for overall comparison into how much more the company spent on sustainability investment than an average company.

```python
# Calculate industry averages for each emission metric
industry_avg = df.groupby('Industry').mean()[['Carbon Emissions (MT)', 'Water Usage (m^3)',
                                               'Energy Consumption (MWh)', 'Waste Generation (MT)']]
```

```python
# Merge industry averages back to the original DataFrame
df = pd.merge(df, industry_avg, left_on='Industry', right_index=True, suffixes=('', '_industry_avg'))
```

```python
# Create new columns for percentage above or below industry average for each metric
df['Carbon Emissions relative to industry average (%)'] =(
    (df['Carbon Emissions (MT)'] - df['Carbon Emissions (MT)_industry_avg'])
                / df['Carbon Emissions (MT)_industry_avg']) * 100

df['Water Usage relative to industry average (%)'] = (
    (df['Water Usage (m^3)'] - df['Water Usage (m^3)_industry_avg'])
                    / df['Water Usage (m^3)_industry_avg']) * 100

df['Energy Consumption relative to industry average (%)'] = (
    (df['Energy Consumption (MWh)'] - df['Energy Consumption (MWh)_industry_avg'])
                        / df['Energy Consumption (MWh)_industry_avg']) * 100

df['Waste Generation relative to industry average (%)'] = (
    (df['Waste Generation (MT)'] - df['Waste Generation (MT)_industry_avg']) / df['Waste Generation (MT)_industry_avg']) * 100
```

```python
# Calculate industry averages for each emission metric
industry_avg2 = df.groupby('Industry').mean()[['Energy Tax', 'Carbon Tax', 'Renewable Energy Use (%)', 'Recycling Rate (%)',
                                                'Eco-friendly Product Lines (%)', 'Sustainable Packaging Initiatives']]
```

```python
# Merge industry averages back to the original DataFrame
df = pd.merge(df, industry_avg2, left_on='Industry', right_index=True, suffixes=('', '_industry_avg'))
```

```python
# Create new columns for how much more or less Energy Tax and Carbon Tax relative to industry average
df['Energy Tax difference from industry average ($)'] = (df['Energy Tax'] - df['Energy Tax_industry_avg'])
df['Carbon Tax difference from industry average ($)'] = (df['Carbon Tax'] - df['Carbon Tax_industry_avg'])
```

```python
# Create new columns for how much more 'sustainable' the company is relative to industry average
df['Renew-Energy difference from industry (%)'] = (df['Renewable Energy Use (%)'] - df['Renewable Energy Use (%)_industry_avg'])
df['Recycling Rate difference from industry (%)'] = (df['Recycling Rate (%)'] - df['Recycling Rate (%)_industry_avg'])
df['Eco-friendly Product Lines difference from industry (%)'] = (
    df['Eco-friendly Product Lines (%)'] - df['Eco-friendly Product Lines (%)_industry_avg'])

df['Sustainable Packaging Initiatives difference from industry'] = (
    df['Sustainable Packaging Initiatives'] - df['Sustainable Packaging Initiatives_industry_avg'])
```

```python
df['Investment amount difference from industry average ($)'] = (
    df['Investment in Sustainability (USD)'] - df['Investment in Sustainability (USD)'].mean())
```

Data is ready to be used for Modelling:

Clear useless columns made while creating new features

```python
df = df.drop(columns=['Tax Percentage for Carbon', 'Tax Percentage for Energy'])  # drop tax percentage columns
df = df.loc[:, ~df.columns.str.endswith('_industry_avg')]                         # drop all columns ending with '_industry_avg'
```

```python
df.head() # preview prepared dataset
```

Supply Chain Sustainability Score	Employee Transportation Mode	...	Water Usage relative to industry average (%)	Energy Consumption relative to industry average (%)	Waste Generation relative to industry average (%)	Energy Tax difference from industry average ($)	Carbon Tax difference from industry average ($)	Renew-Energy difference from industry (%)	Recycling Rate difference from industry (%)	Eco-friendly Product Lines difference from industry (%)	Sustainable Packaging Initiatives difference from industry	Investment amount difference from industry average ($)
55.0	Car	...	-67.376460	64.072112	-13.579389	4555.098068	828.370574	-27.386378	16.207169	-8.304554	0.977703	-8.718245e+05
93.0	Car	...	-11.443786	34.778981	-24.231970	3360.761871	-1055.379426	-29.656378	-22.582831	-30.454554	2.977703	1.868472e+06
85.0	Public Transport	...	23.513127	-89.814835	-6.000263	-2653.145687	-67.479426	-23.106378	-18.502831	35.045446	-1.022297	-1.702705e+06
88.0	Public Transport	...	-33.899305	71.019038	-56.290355	1104.942487	1203.270574	11.623622	-2.962831	-7.584554	-0.022297	2.172012e+06
87.0	Walking	...	-95.813191	-59.561209	16.373254	-1360.809555	1120.020574	-21.066378	-21.622831	17.865446	-1.022297	1.778306e+06

```python
# save to csv
df.to_csv('data_modelling.csv', index=False) # don't follow index number because some Record IDs were dropped
```

To use this cleaned and prepared dataset in SAS EM, I saved it as a csv file.

Exploratory Data Analysis (EDA):
Uncover patterns between features and Target to know which features could be useful predictors.

Pearson's correlation with Target

Extremely weak and insignificant correlation between numerical columns and Regression Target, interpreted by low R-Squared value. Highest R-Square of 0.019 means Waste Generation is 1.9% related to Target. This indicates the absence of a direct relationship between Target and features, suggesting that Regression Target will perform poorly due to lack of direct patterns with predictors.

To investigate if there is a discernible, underlying pattern between features and Classification Target, I will analyze the relationship between Regression and Classification Target.

```
df.corr()['Likelihood of Achieving Sustainability Goals'].sort_values(ascending=False)
```

```
Likelihood of Achieving Sustainability Goals              1.000000
Waste Generation (MT)                                     0.019063
Waste Generation relative to industry average (%)         0.018984
Energy Consumption relative to industry average (%)       0.016558
Energy Consumption (MWh)                                   0.016435
Renew-Energy difference from industry (%)                 0.008547
Renewable Energy Use (%)                                  0.008329
Carbon Tax                                                0.007683
Carbon Emissions (MT)                                     0.007683
Carbon Tax difference from industry average ($)           0.007664
Energy Tax difference from industry average ($)           0.007626
Carbon Emissions relative to industry average (%)         0.007582
Unrenewable Energy (MWh)                                  0.006805
Eco-friendly Product Lines (%)                            0.006597
Recycling Rate (%)                                        0.006473
Recycling Rate difference from industry (%)               0.006305
Energy Tax                                                0.006229
Eco-friendly Product Lines difference from industry (%)   0.006143
Community Engagement Score                                0.003227
Sustainable Packaging Initiatives                        -0.002379
Sustainable Packaging Initiatives difference from industry -0.002561
Investment in Sustainability (USD)                       -0.004727
Investment amount difference from industry average ($)   -0.004727
Engagement Impact Score                                  -0.007032
Customer Feedback Score on Sustainability                -0.009422
Water Usage relative to industry average (%)             -0.010580
Water Usage (m^3)                                        -0.010784
Supply Chain Sustainability Score                        -0.011623
Record ID                                                -0.016990
Name: Likelihood of Achieving Sustainability Goals, dtype: float64
```

No class imbalance for Low and High as the frequency of these 2 classes seem closely equal. However, there is class imbalance for Medium as there are fewer rows within this range.

To use the classification target, I can either group Low and Medium as 0 and High as 1 (to identify companies with high success rate and learn factors that lead to their success), or Low as 1 (to identify and alert companies at risk of not achieving sustainability goals).

```python
import seaborn as sns
import matplotlib.pyplot as plt

# Set the color palette based on Sustainability_Goals_Segment
ordinal_palette = {'Low': 'red', 'Medium': 'orange', 'High': 'green'}

plt.figure(figsize=(18, 6))
sns.countplot(x='Likelihood of Achieving Sustainability Goals', data=df,
              hue='Sustainability_Goals_Segment', palette=ordinal_palette)
plt.title('Countplot of Likelihood of Achieving Sustainability Goals with Color Coding')
plt.show()
```

[Distribution of Likelihood of achieving sustainability, color-coded using Segments (Classification Target)]

Pattern in Target & categorical features
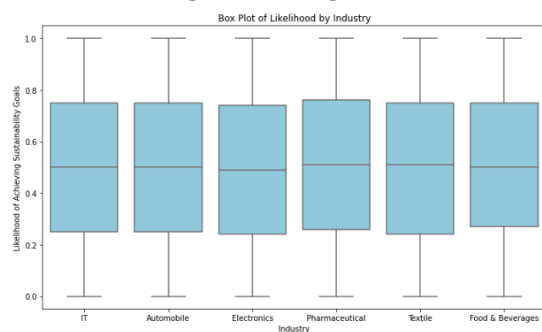
```python
# Find categorical columns
categorical_columns = df.select_dtypes(include='object')
categorical_columns.drop('Company Name', axis=1, inplace=True)

# Plotting box plots
fig, axes = plt.subplots(nrows=len(categorical_columns.columns), ncols=1, figsize=(10, 6 * len(categorical_columns.columns)))

for i, col in enumerate(categorical_columns.columns):
    ax = axes[i]
    sns.boxplot(x=col, y='Likelihood of Achieving Sustainability Goals', data=df, ax=ax, color='skyblue')
    ax.set_title(f'Box Plot of Likelihood by {col}')
    ax.set_xlabel(col)
    ax.set_ylabel('Likelihood of Achieving Sustainability Goals')

plt.tight_layout()
plt.show()
```
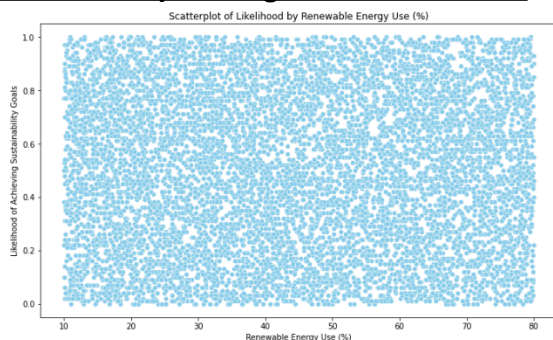
[Code used to generate graphs]

Distribution of Regression Target for every class are closely similar for all columns, implying no pattern. I will try finding patterns using the categorical target instead.

Relationship in Target & num features

```python
# scatterplot to identify linear relationship between numerical features and target
numerical_columns = df.select_dtypes(include=['float64', 'int64'])  # Find numerical columns
fig, axes = plt.subplots(nrows=len(numerical_columns.columns), ncols=1, figsize=(10, 6 * len(numerical_columns.columns)))

for i, col in enumerate(numerical_columns.columns):
    ax = axes[i]
    sns.scatterplot(x=col, y='Likelihood of Achieving Sustainability Goals', data=df, ax=ax, color='skyblue')
    ax.set_title(f'Scatterplot of Likelihood by {col}')
    ax.set_xlabel(col)
    ax.set_ylabel('Likelihood of Achieving Sustainability Goals')

plt.tight_layout()
plt.show()
```
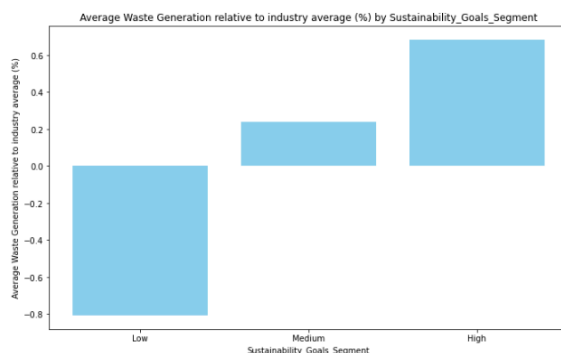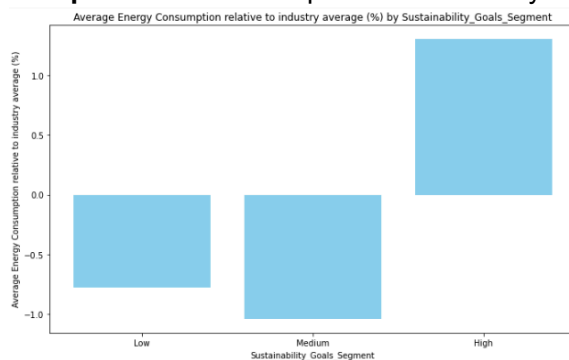
[Code used to generate graphs]

No linear relationship between regression target and any numerical features as all scatter plots have no visible trendline to suggest correlation. The scatterplots mostly resemble the photo on the left, where points are scattered all around with no clear or strong trend.

Pattern in Class Target & num features

All features have no meaningful pattern as all 3 classes have almost the same value…

Average Waste Generation (MT) by Sustainability_Goals_Segment
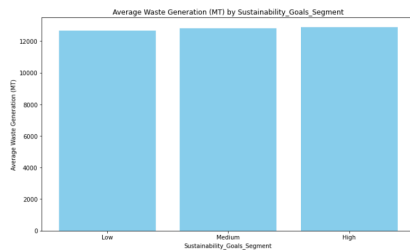
Except for features compared with industry.

Average Energy Consumption relative to industry average (%) by Sustainability_Goals_Segment

```
# Select only numeric columns
numeric_columns = df.select_dtypes(include='number')

# Group by 'Sustainability_Goals_Segment' and calculate the mean for each numeric feature
feature_averages = df.groupby('Sustainability_Goals_Segment').mean()

# Plotting
fig, axes = plt.subplots(nrows=len(numeric_columns.columns), ncols=1, figsize=(10, 6 * len(numeric_columns.columns)))

for i, col in enumerate(numeric_columns.columns):
    ax = axes[i]
    ax.bar(feature_averages.index, feature_averages[col], color='skyblue')
    ax.set_title(f'Average {col} by Sustainability_Goals_Segment')
    ax.set_xlabel('Sustainability_Goals_Segment')
    ax.set_ylabel(f'Average {col}')

plt.tight_layout()
plt.show()
```
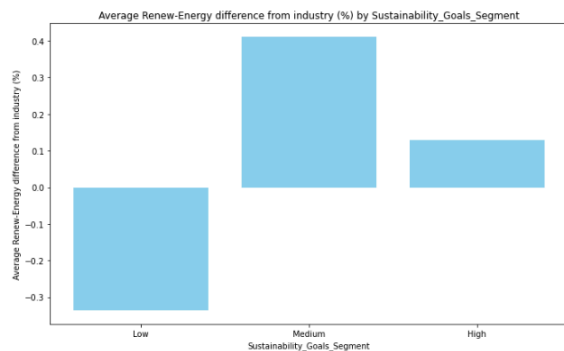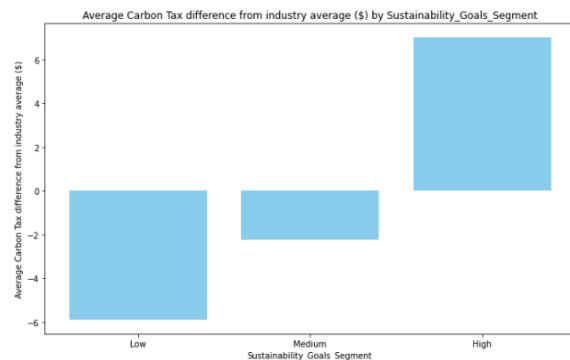
[Code used to generate graphs]

Only graphs with interesting patterns are shown as this code produces a graph for every column.

Average Carbon Tax difference from industry average ($) by Sustainability_Goals_Segment

Average Waste Generation relative to industry average (%) by Sustainability_Goals_Segment

Average Renew-Energy difference from industry (%) by Sustainability_Goals_Segment

All features created in features engineering that are compared with its industry average produced a similar chart where each class has a varied average value from another class, indicating a distinct, underlying pattern and trend within each class.

Graphs **prove that the engineered features** are not just meaningful in addressing the business problem, but also **useful in improving model performance** since they have a varied, distinct trend for each class, revealing potential as impactful predictors.

Conclusion of EDA

- Low Pearson's R-Squared values in correlation test reveals **no significant relationship between any feature and Target**.
- Similarity in distribution of regression target across all categorical columns, implying **no noticeable underlying pattern between categorical features and Target**.
- There is **no linear relationship between numerical features and target**, making **Linear and Logistic Regression unfeasible models** as the data violates the assumption of linearity required for these models to function as intended.
- Newly engineered features comparing values with industry averages exhibit distinct patterns within each class of classification target, making them meaningful predictors to focus on.

According to Oracle, predictive modelling is about training machines to learn patterns in a datasets. Since there is a pattern between engineered features and the classification target, I can **proceed with building Classification models**. Since the patterns are non-linear, I **require models able to capture intricate and complex patterns within data**. As the patterns between regression target and features are weak and insignificant, **regression models will not be attempted** as there is no point overlapping with classification where it already suffices.

Predictive Modelling

Candidate Models

I will proceed with supervised modelling because I have to use a Target column to find out what leads to high or low likelihood. After reading Dataaspirant's article, I have shortlisted the following models:

1. **Decision Tree** (DT) - Easy to interpret and explain as I can visualise the model's tree to understand the decision making process.
2. **Neural Network** (NN) - Advanced model able to capture deep patterns within data, known to deliver high performance (accuracy). However, it is very hard to interpret since it's a black-box method.
3. **K-Nearest Neighbour** (KNN) - Forms k (user-specified) clusters in the data by identifying the closest neighbors based on predictor variables, effectively separating the data into distinct groups or categories.

About Models

What is a Decision Tree (DT):

It recursively partitions the dataset into subsets based on the values of different features. The goal is to create a tree-like model where each internal node represents a decision based on a specific feature, each branch represents the outcome of that decision, and each leaf node represents the final predicted class.

Assumptions:
1. Non-linearity: Decision Trees are well-suited for capturing non-linear relationships within the data.
2. Feature Importance: Assumes certain features are more important than others in making predictions.
3. Hierarchy of Features: Assumes an inherent hierarchy order of predictor features.

How DT can be improved:
1. Prune the tree to prevent overfitting (where tree grows too wide and deep) by limiting tree complexity using the following methods:
   - Setting a maximum depth.
   - Implementing a minimum sample split.
   - Applying a minimum impurity decrease.

What is a Neural Network (NN):

Layers of interconnected neurons, including an input layer, one or more hidden layers, and an output layer. Each connection has an associated weight, and the model learns these weights during training to make predictions.

Assumptions of NN:
1. Complex Patterns: Neural Networks are capable of learning intricate and non-linear patterns in data due to their ability to model complex relationships between input features.
2. Representation Learning: Assumes that the model can automatically learn and extract meaningful representations of the data at different levels of abstraction through the hidden layers.

How NN can be improved:
1. Architecture Tuning: Adjust the number of hidden layers and neurons in each layer. Experiment with different architectures to find the optimal balance between model complexity and performance.
2. Regularization: Apply regularization techniques like dropout that randomly drops a fraction of neurons during training. This prevents the network from relying too heavily on specific neurons (avoid overfitting).
3. Batch Normalization: Introduce batch normalization layers to normalize the inputs of each layer, which can accelerate training and improve the overall stability of the neural network.
4. Learning Rate Schedule: Implement a learning rate schedule to adjust the learning rate during training. This can help the model converge faster and achieve better generalization.
5. Activation Functions: Experiment with different activation functions in hidden layers to introduce non-linearity and enhance the network's capacity to capture complex patterns.

What is K-Nearest Neighbors (KNN):

Classifies a data point by calculating the distances between the target point and every other point in the dataset. The algorithm then identifies the k-nearest neighbors based on these distances and assigns the class label that is most common among these neighbors to the target data point through a majority voting mechanism.

Assumptions:
1. Local Smoothness: Assumes that points in close proximity in the feature space belong to the class.
2. K-Value Significance: Assumes that an appropriate choice of k is crucial for the model's performance.

How KNN can be improved:
- Optimal K that balances bias (high K = more bias) and variance (low K = more variance within cluster).
- Feature Selection: KNN can suffer from the curse of dimensionality where high number of features lead to decreased performance. Hence, features selection or PCA to reduce dimensionality.
- Feature Scaling: ensure that all variables contribute equally to the distance computation and dampen noise from outliers. This helps prevent features with larger scales from dominating the distance metric.
- Distance Metric: Find the distance metrics (e.g., Euclidean, Manhattan) most appropriate for dataset.

Configurations for File Import in SAS EM Miner:

Name	Label	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
						Yes	.	.
Record_ID	Record ID	ID	Interval	No		No	.	.
Company_Name	Company Name	ID	Nominal	No		No	.	.
Renewable_Energy_Use____	Renewable Energy Use (%)	Input	Interval	No		No	.	.
Sustainability_Reporting_Frequ	Sustainability Reporting Frequen	Input	Nominal	No		No	.	.
Supply_Chain_Sustainability_Sc	Supply Chain Sustainability Scor	Input	Interval	No		No	.	.
Recycling_Rate____	Recycling Rate (%)	Input	Interval	No		No	.	.
Investment_in_Sustainability__	Investment in Sustainability (US	Input	Interval	No		No	.	.
Renew_Energy_difference_fro	Renew-Energy difference from ind	Input	Interval	No		No	.	.
Recycling_Rate_difference_fro	Recycling Rate difference from i	Input	Interval	No		No	.	.
Waste_Generation_relative_to	Waste Generation relative to ind	Input	Interval	No		No	.	.
Water_Usage_relative_to_indu	Water Usage relative to industry	Input	Interval	No		No	.	.
Unrenewable_Energy__MWh_	Unrenewable Energy (MWh)	Input	Interval	No		No	.	.
Water_Usage__m_3_	Water Usage (m^3)	Input	Interval	No		No	.	.
Waste_Generation__MT_	Waste Generation (MT)	Input	Interval	No		No	.	.
Sustainable_Packaging_Initiativ	Sustainable Packaging Initiative	Input	Interval	No		No	.	.
Sustainability_Training	Sustainability Training	Input	Nominal	No		No	.	.
Third_party_Sustainability_Cer	Third-party Sustainability Certi	Input	Nominal	No		No	.	.
Customer_Feedback_Score_on	Customer Feedback Score on Susta	Input	Interval	No		No	.	.
Eco_friendly_Product_Lines___	Eco-friendly Product Lines (%)	Input	Interval	No		No	.	.
Community_Engagement_Score	Community Engagement Score	Input	Interval	No		No	.	.
Employee_Transportation_Mod	Employee Transportation Mode	Input	Nominal	No		No	.	.
Eco_friendly_Product_Lines_di	Eco-friendly Product Lines diffe	Input	Interval	No		No	.	.
Carbon_Emissions_relative_to	Carbon Emissions relative to ind	Input	Interval	No		No	.	.
Carbon_Emissions__MT_	Carbon Emissions (MT)	Input	Interval	No		No	.	.
Carbon_Tax_difference_from_	Carbon Tax difference from indus	Input	Interval	No		No	.	.
Carbon_Tax	Carbon Tax	Input	Interval	No		No	.	.
Engagement_Impact_Score	Engagement Impact Score	Input	Interval	No		No	.	.
Energy_Tax_difference_from_i	Energy Tax difference from indus	Input	Interval	No		No	.	.
VAR35	Investment amount difference fro	Input	Interval	No		No	.	.
Industry		Input	Nominal	No		No	.	.
Energy_Consumption_relative_	Energy Consumption relative to i	Input	Interval	No		No	.	.
Energy_Consumption__MWh_	Energy Consumption (MWh)	Input	Interval	No		No	.	.
Energy_Tax	Energy Tax	Input	Interval	No		No	.	.
Likelihood_of_Achieving_Sustai	Likelihood of Achieving Sustaina	Rejected	Interval	No		Yes	.	.
Sustainability_Goals_Segment		Target	Nominal	No		No	.	.

[Exclude Regression Target. Assign Record ID and Company Name as ID so they will not be used as predictors.]

Data Preparation for Classification Modelling

Encoding Categorical columns
Ordinal encoded:
- **Employee Transportation Mode:**
  Car as 2, Public Transport as 1, and Walking and Bike as 0. (Ranking of environmentally harmful transportation mode most frequently used.)

- **Sustainability Reporting Frequency:**
  Quarterly as 3, Bi-annually as 2, Annually as 1 (Rank how frequent a company reports in a way machine learning models can understand.)

Binary encoded:
- Sustainability Training and Third party Sustainability Certification where 1 represents Yes and 0 represents No.

Encoding is needed because NN and KNN models only accept predictors in numeric form.

Variable	Formatted Value	Replacement Value
Sustainability_Reporting_Frequen	Quarterly	3
Employee_Transportation_Mode	Car	2
Sustainability_Reporting_Frequen	Bi-annually	2
Employee_Transportation_Mode	Public Transport	1
Sustainability_Reporting_Frequen	Annually	1
Sustainability_Training	Yes	1
Third_party_Sustainability_Certi	Yes	1
Employee_Transportation_Mode	Bike	0
Employee_Transportation_Mode	Walking	0
Sustainability_Training	No	0
Third_party_Sustainability_Certi	No	0
Employee_Transportation_Mode	_UNKNOWN_	_DEFAULT_
Industry	_UNKNOWN_	_DEFAULT_
Sustainability_Goals_Segment	_UNKNOWN_	_DEFAULT_
Sustainability_Reporting_Frequen	_UNKNOWN_	_DEFAULT_
Sustainability_Training	_UNKNOWN_	_DEFAULT_
Third_party_Sustainability_Certi	_UNKNOWN_	_DEFAULT_

Train
Interval Variables	
Replacement Editor	...
Default Limits Method	None
Cutoff Values	...
Class Variables	
Replacement Editor	...
Unknown Levels	Ignore

Train-Test Split
I will employ the Holdout Strategy of splitting my dataset into 70-30 partitions where 70% is for training my predictive models and 30% is for testing their performance on unseen data. I chose this ratio to ensure sufficient data in each partition. This Cross-Validation method is chosen as it's computationally cheap, while being sufficient to validate the performance of my model.

I used my admin no. as seed for randomness to ensure the split will always be the same for every run.

Train
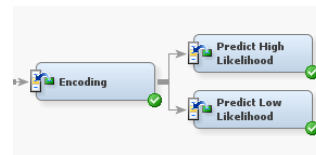| Variables	...
Output Type	Data
Partitioning Method	Default
Random Seed	2202934
Data Set Allocations	
Training	70.0
Validation	30.0
Test	0.0

Encoding Target column

To better understand the key drivers leading to increased probability for a company to achieve their sustainability goals, my predictive models are designed to solve these questions:

1. What conditions lead to a high likelihood of achieving sustainability goals?
2. What conditions lead to low likelihood of achieving sustainability goals?

To answer Qn1, I encoded High likelihood (>60%) as 1 and any probability below 0.6 as 0.

To answer Qn2, I encoded Low likelihood (<40%) as 1 and any probability above 0.4 as 0.

Encoding the Target as such allows the predictive models to learn the nuances of how various features lead to a High or Low likelihood for a company to achieve their sustainability goals, enabling me to generate insights and construct recommendations for companies to increase their likelihood of achieving their sustainability goals on schedule.

Replacement Editor-WORK.OUTCLASS

Variable	Formatted Value	Replacement Value
Sustainability_Goals_Segment	High	1
Sustainability_Goals_Segment	Low	0
Sustainability_Goals_Segment	Medium	0
Industry	_UNKNOWN_	_DEFAULT_
REP_Employee_Transportation_Mode	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Reporting_Fre	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Training	_UNKNOWN_	_DEFAULT_
REP_Third_party_Sustainability_C	_UNKNOWN_	_DEFAULT_
Sustainability_Goals_Segment	_UNKNOWN_	_DEFAULT_

Replacement Editor-WORK.OUTCLASS

Variable	Formatted Value	Replacement Value
Sustainability_Goals_Segment	Low	1
Sustainability_Goals_Segment	High	0
Sustainability_Goals_Segment	Medium	0
Industry	_UNKNOWN_	_DEFAULT_
REP_Employee_Transportation_Mode	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Reporting_Fre	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Training	_UNKNOWN_	_DEFAULT_
REP_Third_party_Sustainability_C	_UNKNOWN_	_DEFAULT_
Sustainability_Goals_Segment	_UNKNOWN_	_DEFAULT_

Features Selection (Understanding)

Although I already found out that only features created during features engineering where I compared to the industry average have a noticeable pattern, I still decided to examine the feature importances using StatExplore.

Based on StatExplore variable worth, I identified that the difference between importances for features are insignificant, as they are all equally weak just as found in correlation testing. Categorical features are worthless and would be insignificant predictors to predict the Target.
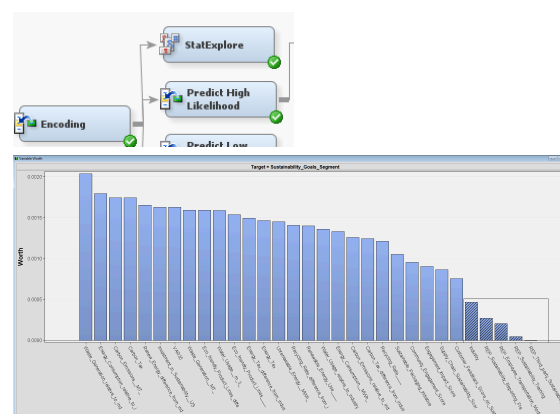

Modelling (TARGET=High Sustainability Goals Segment)

Features Selection

StatExplore to identify useful features for predicting high likelihood class.

DT: **do not need feature selection** as the model's recursion splitting algorithm does the job for me.

NN: advanced model able to learn intricate patterns, so I will **select features relevant to the business domain** of Sustainability.

KNN: **use high-worth features and exclude overlapping features** to avoid multicollinearity (inability to identify key drivers) and curse of dimensionality (lead to a sparse and dispersed data space, making it difficult for the algorithm to identify meaningful patterns; worsen performance).
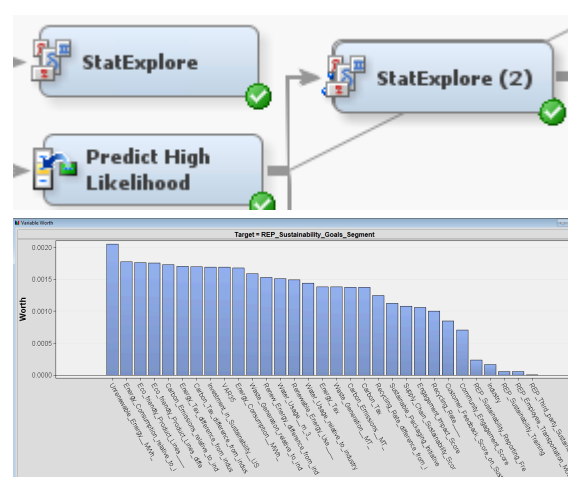
Usually for predictive modelling, we start with a simple model to get a gauge of the baseline performance to measure improvements of advanced models. Since Logistic Regression is not applicable due to non-linear patterns in data, I will attempt Polynomial Logistic Regression, an extension of Logistic Regression applicable for data with non-linear relationships between independent variables and the log-odds of the dependent variable, where polynomial terms are introduced to capture the complex patterns in the data.

Polynomial Logistic Regression
- Excluded low worth predictors (categorical features) and overlapping features like carbon emissions when carbon emission relative to industry already present.
- Introduced polynomial features that may better capture nonlinear relationships within the data.
- Set the **Polynomial degree to 2 so the model doesn't overfit** or become unexplainable due to being overly complex.
- **Did not introduce my own user terms** since I want to **let the machine learn for itself** and identify meaningful interactions between features.

Name	Label	Use	Report	Role	Level
Carbon_Tax	Carbon Tax	Yes	No	Input	Interval
Water_Usage_relative_to_industry	Water Usage relative to industry	Yes	No	Input	Interval
Energy_Tax	Energy Tax	Yes	No	Input	Interval
Carbon_Emissions_relative_to_ind	Carbon Emissions relative to ind	Yes	No	Input	Interval
Unrenewable_Energy__MWh_	Unrenewable Energy (MWh)	Yes	No	Input	Interval
Community_Engagement_Score	Community Engagement Score	Yes	No	Input	Interval
Waste_Generation_relative_to_ind	Waste Generation relative to ind	Yes	No	Input	Interval
Sustainable_Packaging_Initiative	Sustainable Packaging Initiative	Yes	No	Input	Interval
Customer_Feedback_Score_on_Susta	Customer Feedback Score on Susta	Yes	No	Input	Interval
Supply_Chain_Sustainability_Scor	Supply Chain Sustainability Scor	Yes	No	Input	Interval
Eco_friendly_Product_Lines_diffe	Eco-friendly Product Lines diffe	Yes	No	Input	Interval
Energy_Consumption_relative_to_i	Energy Consumption relative to i	Yes	No	Input	Interval
Investment_in_Sustainability__US	Investment in Sustainability (US	Yes	No	Input	Interval
REP_Sustainability_Goals_Segment	Replacement: Sustainability_Goals_Segment	Yes	No	Target	Nominal
Recycling_Rate____	Recycling Rate (%)	Yes	No	Input	Interval
Industry		No	No	Input	Nominal

| Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	Yes
Polynomial Degree	2
User Terms	No
Term Editor	

Decision Tree Classifier
For Baseline Decision Tree (DT), I compared between Chisq and Entropy splitting criteria for the nominal target to find the best splitting criteria that leads to highest accuracy. The model takes full control in training itself (no pruning) to determine best splitting criteria. Then, Model Comparison is used to find the champion model.

Results:
Although ChiSq leads to best testing performance, it only has 1 branch since the other features have insignificant chi-square value to continue splitting.
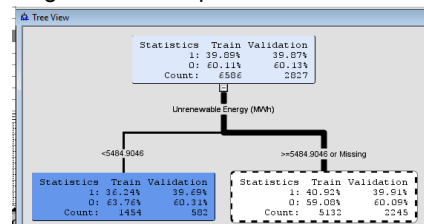
Hence, I will **proceed with Gini splitting criteria** as it's the best model that doesn't stop at its first split.
[*refer to Model Comparison*]
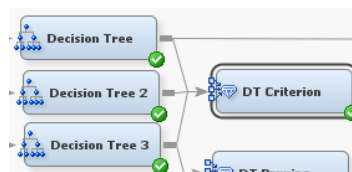
DT:

Splitting Rule	
Interval Target Criterion	ProbF
Nominal Target Criterion	ProbChisq
Ordinal Target Criterion	Entropy
Significance Level	0.2

DT 2:

Splitting Rule	
Interval Target Criterion	ProbF
Nominal Target Criterion	Entropy
Ordinal Target Criterion	Entropy

DT 3:

Splitting Rule	
Interval Target Criterion	ProbF
Nominal Target Criterion	Gini
Ordinal Target Criterion	Entropy

Model Comparison:
```
Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

                                      Train:              Valid:
                           Valid:     Average    Train:   Average
Selected  Model   Model    Misclassification Squared Misclassification Squared
Model     Node    Description   Rate    Error     Rate     Error

   Y      Tree    Decision Tree    0.39866  0.23940  0.39888  0.24005
           Tree5   Decision Tree 3  0.40467  0.23171  0.38263  0.24781
           Tree2   Decision Tree 2  0.40821  0.23377  0.38718  0.24573
```

Pruning DT
Made another DT using baseline DT where Subtree Assessment [photo attached] is used to find the optimal depth to prune the Tree to a depth of 12.

This is done to address overfitting, to make the Decision Tree more generalized to perform equally well with unseen testing data.

Referring to the screenshot below, pruning prevents overfitting since train and test misclassification rate is more consistent. However, this is insignificant, so I conclude that pruning is not important since the DT doesn't face any severe overfitting performance issue without pruning.

```
Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

                                      Train:              Valid:
                           Valid:     Average    Train:   Average
Selected  Model   Model    Misclassification Squared Misclassification Squared
Model     Node    Model Description   Rate    Error     Rate     Error

   Y      Tree3   Pruned Decision Tree  0.39795  0.23664  0.39538  0.24224
           Tree5   Decision Tree 3       0.40467  0.23171  0.38263  0.24781
```

I will retain the unpruned DT 3 as I can still mine insights from it.
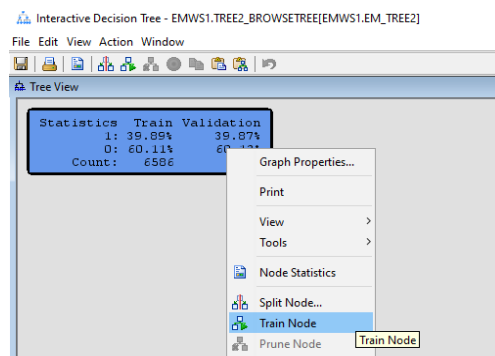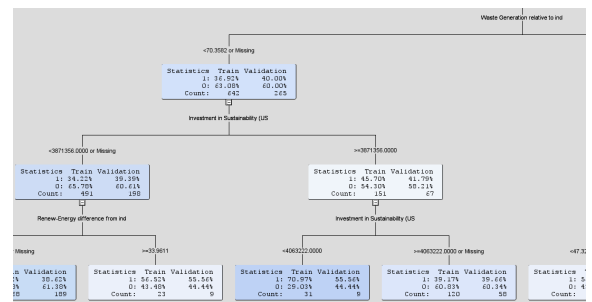
Interpreting DT3

There are some interesting insights like how companies that have low carbon emission but consume more energy have high likelihood (bottom left branch).

Another insight is that high carbon emission companies who spend below $117,237 investing on sustainability have higher likelihood than companies who spend more. However, this insight isn't tested as there are no validation records in this node.
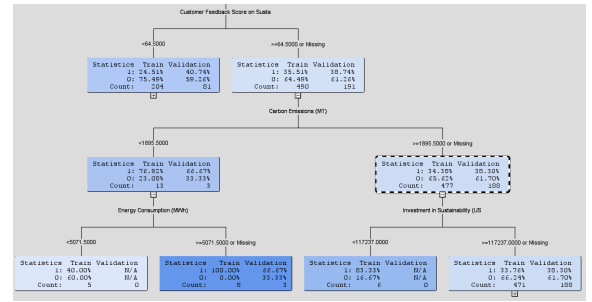
For companies generating less than 70.4% waste than industry average AND invest below $3,871,356 on investment, if their renewable energy (%) is industry average% + 34% or more, they have a 56-57% chance of high likelihood. This is validated by a testing set.

For companies generating less than 70.4% waste than industry average, the optimal investment is $3,871,356 to $4,063,221 based on DT3.

Interpreting Pruned DT

This DT only has 2 leaves that leads to positive class, and gives illogical insights that seem ridiculous if mentioned to businesses. For instance, lower recycling rate, lower customer feedback, and higher carbon emissions lead to high likelihood of achieving sustainability goals.

Th3 other insight does not really dig deep enough, and is only supported by 8 instances:
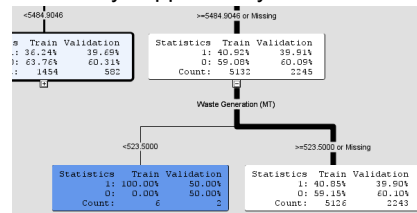
Although this pruned DT might lead to most optimal performance, it's useless since the insights it generates are meaningless and unable to concretely distinguish key factors that lead to high likelihood.
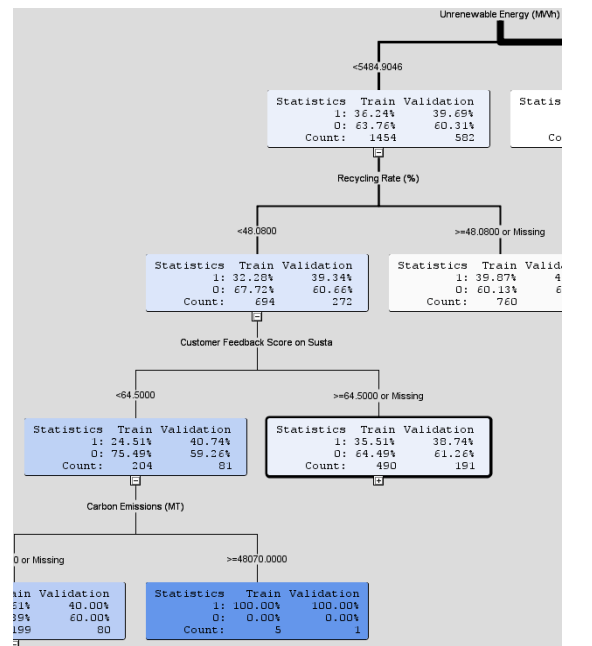
Autoneural

I also attempt a black box model, which is known for exceptional performance at the expense of interpretability, Neural Network (NN), to gauge how well a model can possibly perform on the dataset. I selected predictors which were engineered by comparing with industry average since those features have a smaller range, enhance NN'S ability to discern subtle patterns in the data. **NN will not be chosen as black box models are not transparent and cannot be understood fully.** Therefore, businesses may not trust the model so I won't select this model to present to stakeholders.

So for the configurations, I will let AutoNeural help me search for the best configuration for my dataset that leads to best performance.

Name	Label	Use ▽	Report	Role	Level
Energy_Consur	Energy Consumption relative to i	Yes	No	Input	Interval
REP_Employee	Replacement: Employee Transportation Mode	Yes	No	Input	Nominal
Energy_Tax_di	Energy Tax difference from indus	Yes	No	Input	Interval
Carbon_Tax_d	Carbon Tax difference from indus	Yes	No	Input	Interval
Engagement_I	Engagement Impact Score	Yes	No	Input	Interval
Waste_Genera	Waste Generation relative to ind	Yes	No	Input	Interval
Supply_Chain_	Supply Chain Sustainability Scor	Yes	No	Input	Interval
Water_Usage_	Water Usage relative to industry	Yes	No	Input	Interval
VAR35	Investment amount difference fro	Yes	No	Input	Interval
REP_Sustainab	Replacement: Sustainability Reporting Frequer	Yes	No	Input	Nominal
Renew_Energy	Renew-Energy difference from ind	Yes	No	Input	Interval
REP_Sustainab	Replacement: Sustainability Training	Yes	No	Input	Nominal
REP_Sustainab	Replacement: Sustainability_Goals_Segment	Yes	No	Target	Nominal
Eco_friendly_P	Eco-friendly Product Lines (%)	Yes	No	Input	Interval
Carbon_Emissi	Carbon Emissions relative to ind	Yes	No	Input	Interval
REP_Third_par	Replacement: Third-party Sustainability Certi	Yes	No	Input	Nominal
Recycling_Rate	Recycling Rate difference from i	Yes	No	Input	Interval
Waste_Genera	Waste Generation (MT)	No	No	Input	Interval

| Model Options	
Architecture	Single Layer
Termination	Overfitting
Train Action	Search
Target Layer Error Function	Default
Maximum Iterations	20
Number of Hidden Units	2
Tolerance	Low
Total Time	One Hour

Customized Decision Tree

I will now prune the DT with business understanding and performance optimization in mind. As this model is meant for identifying class 1 to recognize what conditions and factors lead to achieving sustainability goals, I will focus my pruning to generate leaves that can identify class 1.

My priority when building the Decision Tree is ranked accordingly:
1. Business understanding (for choosing features and sequence of conditions).
2. Interpretability (setting the threshold values for each split).
3. Accuracy (purity of each leaf).

Challenge is to balance 1 and 3 by choosing features and sequences that address business understanding while having acceptable performance.

First split is Unrenewable Energy. To make the value more appealing to present to business users, I will round off the split value to 5480.
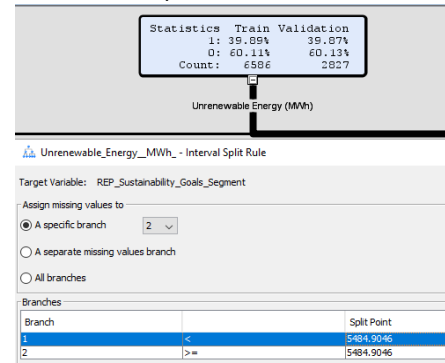
To balance priority 1 and 3, I try to use the best split before working down the interval split rule leaderboard.

## How I customized the splits (step-by-step)

LEFT Branch
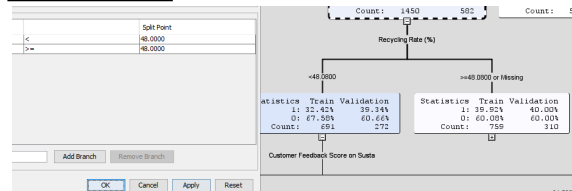
I changed the split value from 48.08 to 48 for business appeal.

After experimenting with different features and splits, I decided to split using carbon emission relative to industry as I spotted some target=1 which could be linked to business understanding. Here, I subsetted carbon emissions into heavy carbon emission companies (90% and above their industry average) and low carbon emitters (90-95% below its industry average). These 2 subsets may reveal a pattern between these conditions and likelihood of achieving goals. Below -95% **(ridiculously low carbon emitters) could be false records as I assume these companies lied about their emissions to boost their reputation**:
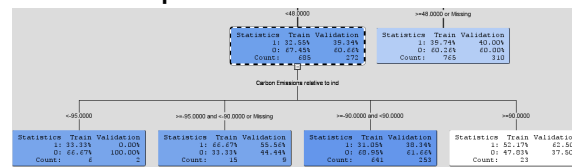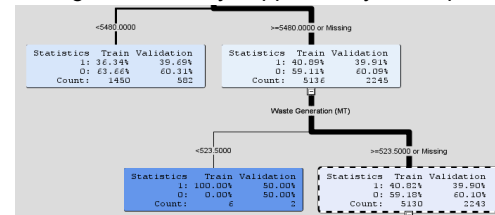

The decision of using Investment in Sustainability as the feature for the next split after >=48% is driven by the business domain (rather than feature worth):
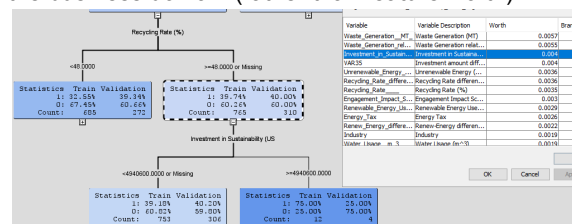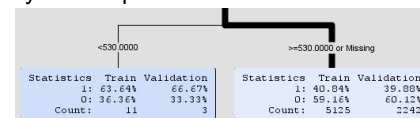

RIGHT Branch

Found a leaf identifying a potential pattern that leads to Target=1, but only supported by 8 companies.


Adjusting the threshold to 530, the accuracy was consistent at 67% for train and test sets, supported by 14 companies:

Found this strong split in water usage. However, this occurrence might just be due to chance since the validation set is suggesting the opposite claim.
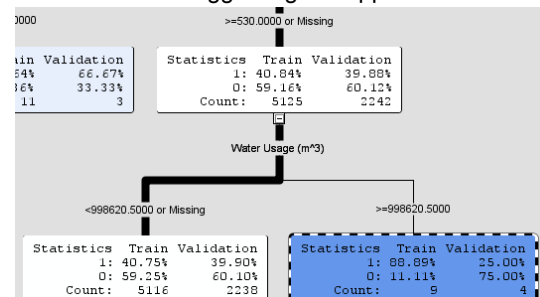
I manually splitted the ranges trying to find interesting subsets. Nothing found this layer, so I will split at 200:
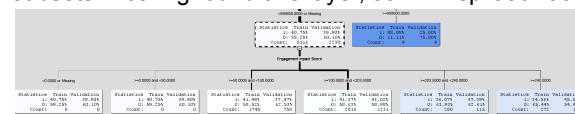

The top splitting feature is the same, so I tried the

My **assumption is that companies that have such a high recycling rate likely care about sustainability for them to be taking these recycling measures**. Hence, I want to assess if they spend considerable money on sustainability.

Another split driven by business domain because I realised that the root split of Unrenewable energy doesn't assess whether the company is an intensive user of energy or if they just have heavy reliance on environmentally-friendly renewable energy.

I pruned the split value to a neat threshold of 33 MWh which also produced a decent 62% purity in target=1.
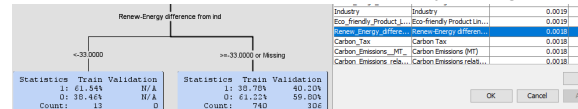
Interpretation: Businesses **can be 62% confident that they have a high likelihood** of achieving sustainability goals if they follow the series of decisions that leads to this node **BUT** this claim is not validated since no company was tested.

second one. After experimenting with different features and split ranges, I was unable to find any high likelihood=1 leaf.
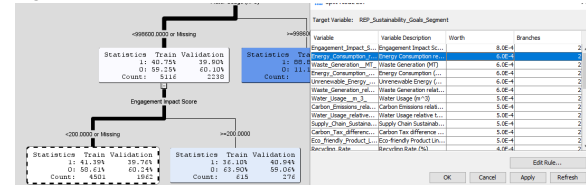

Since there is no high likelihood to be found, I will prune this node to remove the Engagement Impact Score branch as it is meaningless:
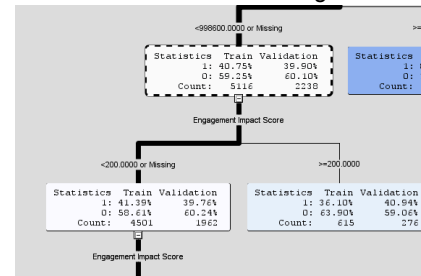
Ensemble

Ensemble Custom DT and Pruned DT to utilize both decent trees instead of only choosing 1. I also ensemble Custom DT, Pruned DT, and DT3 to utilize the insightful positive class nodes found in DT3.

I use maximum probability to let Ensemble choose the best decision that it's most confident with (purest leaf) out of both trees.
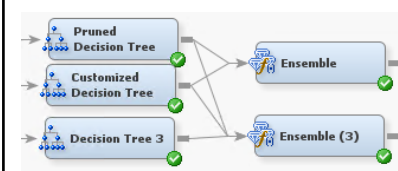
Models Evaluation

Model Node	Model Description	Data Role	False Negative	True Negative	False Positive	True Positive
Tree5	Decision Tree 3	TRAIN	2454	3893	66	173
Tree5	Decision Tree 3	VALIDATE	1090	1646	54	37
Tree4	Customized Decision Tree	TRAIN	2573	3930	29	54
Tree4	Customized Decision Tree	VALIDATE	1113	1686	14	14
Tree3	Pruned Decision Tree	TRAIN	2603	3958	1	24
Tree3	Pruned Decision Tree	VALIDATE	1123	1698	2	4
Reg3	Polynomial Logistic Regression	TRAIN	2461	3810	149	166
Reg3	Polynomial Logistic Regression	VALIDATE	1081	1621	79	46
AutoNeural	AutoNeural	TRAIN	2625	3958	1	2
AutoNeural	AutoNeural	VALIDATE	1125	1700	.	2
Ensmbl	Ensemble	TRAIN	2566	3929	30	61
Ensmbl	Ensemble	VALIDATE	1113	1685	15	14
Ensmbl2	Ensemble (3)	TRAIN	2428	3866	93	199
Ensmbl2	Ensemble (3)	VALIDATE	1081	1636	64	46

As the model's purpose is to identify high likelihood companies, I will use 2 main metrics:
Precision of class 1: Ratio of correct predicted positive class instances; quantify accuracy of class 1 predictions.

Sensitivity: To assess if a model is able to identify all instances of positive class that exist. Higher sensitivity is desired as it indicates lower type 2 error (where model failed to capture high likelihood observations)

Precision Formula: TP/(TP+FP), Sensitivity Formula: TP/(TP+FN)
Rejected: AutoNeural as it's barely able to identify TP, making it useless. Polynomial as it is just a worse version of Ensemble (3) with more FP and lesser TP; lower precision than Ensemble, hence just use Ensemble.

Considered Models

DT3: Train Precision = 0.72, Train Sensitivity = 0.07, Test Precision = 0.4, Test Sensitivity = 0.03
Train precision is decent but the test is lousy. Model only able to identify 3-7% of high likelihood companies.
C-DT: Train Precision = 0.65, Train Sensitivity = 0.02, Test Precision = 0.5, Test Sensitivity = 0.01
Train precision is acceptable but the test is not. Model only able to identify 1-2% of high likelihood companies.
Pruned DT: Train Precision = 0.96, Train Sensitivity = 0.01, Test Precision = 0.67, Test Sensitivity = 0.004
Train and test precision are decent, but the model is only able to identify below 1% of high likelihood companies.
Ensemble: Train Precision = 0.67, Train Sensitivity = 0.02, Test Precision = 0.48, Test Sensitivity = 0.005
Train precision is decent but the test is lousy. Model only able to identify below 2% of high likelihood companies.
Ensemble (3): Train Precision = 0.68, Train Sensitivity = 0.08, Test Precision = 0.42, Test Sensitivity = 0.04
Train precision is decent but the test is lousy. Model able to identify 8% of high likelihood companies in training with around 70% accuracy, but validation performance does not match training, suggesting the distinguished pattern is not well-generalized with unseen data and is subjected to vary.
Pruned DT is the only model that performs decently for training and testing, but **a model that misses 99% of positive class** will not be helpful in identifying companies that have high likelihood.
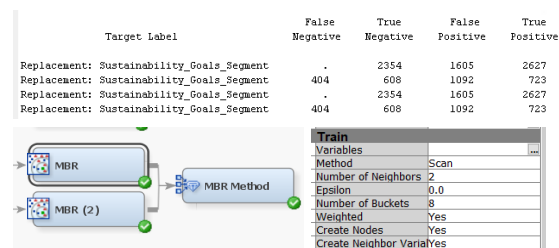**Note:** All Decision Trees suffer from significant overfitting above 10%.
Since all Decision Trees are unacceptable due to atrocious sensitivity rendering them useless…

I will proceed to the Clustering Model, KNN:

|					

KNN (using MBR Node)
I first tried both the RD-Tree and Scan method using k=2 to see which leads to better performance. Both led to identical performance, so I proceed with Scan since it uses distance to a probe observation which aligns with KNN's method. Since I'm using Scan, the other configurations of MBR are not applicable.

Target Label	False Negative	True Negative	False Positive	True Positive
Replacement: Sustainability_Goals_Segment	.	2354	1605	2627
Replacement: Sustainability_Goals_Segment	404	608	1092	723
Replacement: Sustainability_Goals_Segment	.	2354	1605	2627
Replacement: Sustainability_Goals_Segment	404	608	1092	723

Train
Variables	...
Method	Scan
Number of Neighbors	2
Epsilon	0.0
Number of Buckets	8
Weighted	Yes
Create Nodes	Yes
Create Neighbor Varial	Yes

Now, I will compare the optimal k clusters. The criteria I will use to assess better performance using Sensitivity and Accuracy of models.

Notably, K=2 has the highest Lift, no false negative (indicating 100% sensitivity) and highest amount of true positive for training. Train precision = 62%, test precision = 40% and sensitivity = 64%, which indicate overfitting of 22% (deviation in test precision).
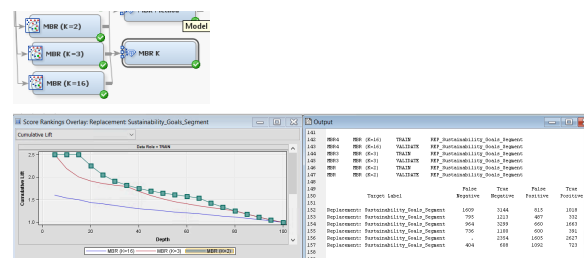
Features Selection
Overfitting could be caused by too many features used as inputs (discussed in pg 11). I will build another MBR using K=2 without overlapping features to check if it fixes the overfitting issue.

Removed:
- community engagement score and customer feedback score as they were used to calculate engagement impact score.
- carbon tax diff and investment amount diff as it's easier to interpret the raw value.
- raw carbon emissions, water usage, etc. as using the features compared with industry average is more impactful based on EDA.
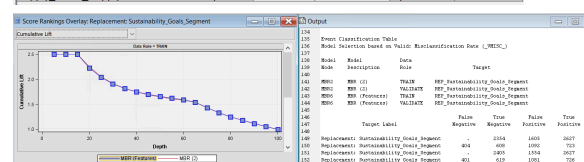
Performance Evaluation:
Train - Sensitivity=100%, Precision=63%.
Test - Sensitivity=64%, Precision=40%.
These changes led to slight improvement for train and test as there are fewer FP and more TP, but **still overfitted as precision deviates by 23%**.
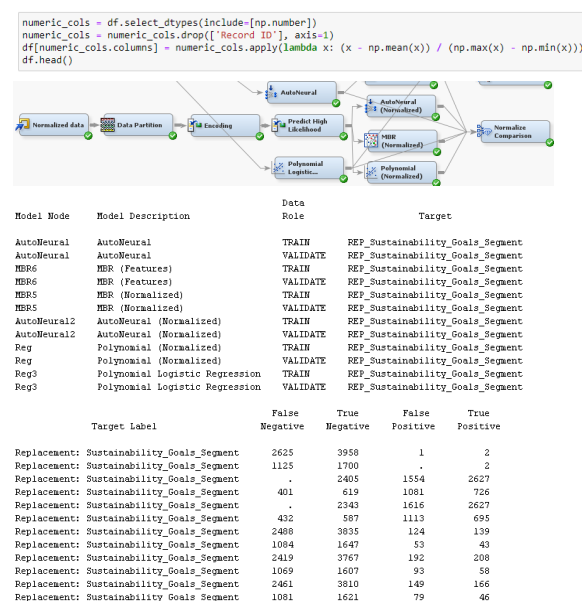

Normalize predictors for better performance
Based on this article by Machine Learning Models.org, distance-based clustering models like KNN and Neural Networks benefit from scaling features - also discussed in pg 11 (About Models).
Steps:
Z-score normalization as my scaling method instead of Log transform as features compared with ind avg have negative bounds. I loaded the cleaned csv into a Jupyter notebook and normalized all the numerical columns in the dataset except for Record ID. Then, I loaded the transformed dataset back into SAS EM Miner using a file reader. I copy and pasted the preparation nodes like data partitioning and encoding and connect the file reader into this pipeline.
Performance Evaluation:
NN improved drastically as it is now able to detect positive classes (but still worse than Ensemble (3)). However, performance for MBR worsened. Polynomial Regression also did not improve despite increased TP since FP also increased.

```
numeric_cols = df.select_dtypes(include=[np.number])
numeric_cols = numeric_cols.drop(['Record ID'], axis=1)
df[numeric_cols.columns] = numeric_cols.apply(lambda x: (x - np.mean(x)) / (np.max(x) - np.min(x)))
df.head()
```


Model Node	Model Description	Data Role	Target
AutoNeural	AutoNeural	TRAIN	REP_Sustainability_Goals_Segment
AutoNeural	AutoNeural	VALIDATE	REP_Sustainability_Goals_Segment
MBR6	MBR (Features)	TRAIN	REP_Sustainability_Goals_Segment
MBR6	MBR (Features)	VALIDATE	REP_Sustainability_Goals_Segment
MBR5	MBR (Normalized)	TRAIN	REP_Sustainability_Goals_Segment
MBR5	MBR (Normalized)	VALIDATE	REP_Sustainability_Goals_Segment
AutoNeural2	AutoNeural (Normalized)	TRAIN	REP_Sustainability_Goals_Segment
AutoNeural2	AutoNeural (Normalized)	VALIDATE	REP_Sustainability_Goals_Segment
Reg	Polynomial (Normalized)	TRAIN	REP_Sustainability_Goals_Segment
Reg	Polynomial (Normalized)	VALIDATE	REP_Sustainability_Goals_Segment
Reg3	Polynomial Logistic Regression	TRAIN	REP_Sustainability_Goals_Segment
Reg3	Polynomial Logistic Regression	VALIDATE	REP_Sustainability_Goals_Segment

Target Label	False Negative	True Negative	False Positive	True Positive
Replacement: Sustainability_Goals_Segment	2625	3958	1	2
Replacement: Sustainability_Goals_Segment	1125	1700	.	2
Replacement: Sustainability_Goals_Segment	.	2405	1554	2627
Replacement: Sustainability_Goals_Segment	401	619	1081	726
Replacement: Sustainability_Goals_Segment	.	2343	1616	2627
Replacement: Sustainability_Goals_Segment	432	587	1113	695
Replacement: Sustainability_Goals_Segment	2488	3835	124	139
Replacement: Sustainability_Goals_Segment	1084	1647	53	43
Replacement: Sustainability_Goals_Segment	2419	3767	192	208
Replacement: Sustainability_Goals_Segment	1069	1607	93	58
Replacement: Sustainability_Goals_Segment	2461	3810	149	166
Replacement: Sustainability_Goals_Segment	1081	1621	79	46

Interpreting Polynomial Regression

Now that features are normalized, I can interpret the influence of each significant feature and interaction between 2 features fairly because scaling enables me to compare how impactful each interaction influences the likelihood with the same weightage of coefficient.

Inspecting significant interactions by hovering on the bars most of these interactions consist of Carbon Tax or Carbon Emissions relative to industry average.
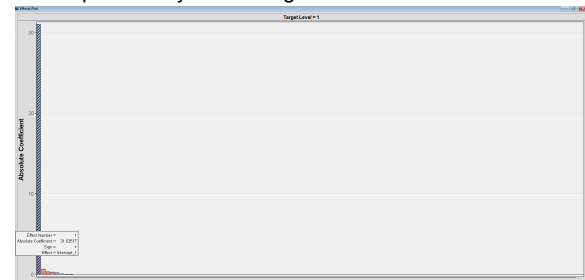
Referring to 'Table: Effects Plot', only 3 interactions are significant enough with a p-value below 0.05:

1. Energy consumption relative to ind avg * Recycling Rate (%)
2. Carbon Tax * Eco-friendly Product Line difference from industry average
3. Carbon Emissions relative to ind avg * Eco-friendly Product Line diff from ind avg

My takeaway from the Polynomial Regression Model is that Carbon Tax and Carbon emissions relative to industry average for a company are important factors to help determine if a company has high likelihood in achieving sustainability goals. Interaction 2 and 3 could have underlying patterns to determine high likelihood. Interaction 1's coefficient of -1 is too small to influence the output of the model significantly.

Effects plot of Polynomial Regression:

Effects plot of Polynomial Regression with scaled features:
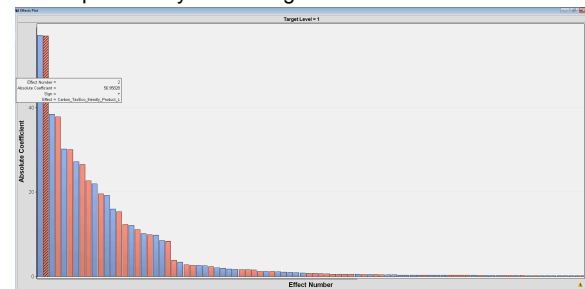
Table: Effects Plot sorted by P-Value ascending:

Effect	Effect Label	Coefficient	T-value	P Value ▲
Energy_Co...	Energy Consumption relative to i * Recycling Rate (%): REP_...	-1.07018	-2.05411	0.039965
Carbon_Ta...	Carbon Tax * Eco-friendly Product Lines diffe: REP_Sustaina...	56.95828	2.01082	0.044344
Carbon_E...	Carbon Emissions relative to ind * Eco-friendly Product Line...	-57.1008	-1.99453	0.046094
Investment...	Investment in Sustainability (US * Waste Generation relative t...	0.615201	1.955699	0.050501
Waste_Gen...	Waste Generation relative to ind * Waste Generation relative t...	0.896249	1.922728	0.054512

Flaw of Z-score scaling with Polynomial Logistic Regression: 'Relative to ind avg' and 'Diff from ind avg' have negative values, making it impossible to identify which feature's direction (+ or -) in the interaction led to the higher probability. Since negative values cancel out, model assumes (-)carbon *(-)eco-friendly = (+)carbon*(+)eco-friendly. This issue is fixed by using Min-max normalization instead. Performance of min-max and z-score are identical, but min-max scaling facilitates intuitive interpretation.


```
numeric_cols = df.select_dtypes(include=['float64', 'int64']) # Extract numeric columns
numeric_cols = numeric_cols.drop(['Record ID'], axis=1)  # drop record id
scaler = MinMaxScaler()                         # instantiate MinMaxScaler
df[numeric_cols.columns] = scaler.fit_transform(numeric_cols) # apply MinMaxScaler
df.describe()
```

Effect Label	Coefficient	P Value ▲
Carbon Emissions relative to ind: REP_Sustainability_Goals_Segment=1	66.68268	0
Energy Consumption relative to i: REP_Sustainability_Goals_Segment=1	2.567832	0.014738
Carbon Tax * Eco-friendly Product Lines diffe: REP_Sustainability_Goals_Se...	56.95828	0.035754
Carbon Emissions relative to ind * Eco-friendly Product Lines diffe: REP_Su...	-57.1008	0.037277
Energy Consumption relative to i * Recycling Rate (%): REP_Sustainability_...	-1.07018	0.039961
Investment in Sustainability (US * Waste Generation relative to: REP_Su...	0.615201	0.050495

Analysis of Models


```
231
232  Event Classification Table
233  Model Selection based on Valid: Misclassification Rate (_VMISC_)
234
235                                        Data
236  Model Node   Model Description        Role          Target
237
238  Tree5        Decision Tree 3          TRAIN         REP_Sustainability_Goals_Segment
239  Tree5        Decision Tree 3          VALIDATE      REP_Sustainability_Goals_Segment
240  Tree3        Pruned Decision Tree     TRAIN         REP_Sustainability_Goals_Segment
241  Tree3        Pruned Decision Tree     VALIDATE      REP_Sustainability_Goals_Segment
242  Tree4        Customized Decision Tree TRAIN         REP_Sustainability_Goals_Segment
243  Tree4        Customized Decision Tree VALIDATE      REP_Sustainability_Goals_Segment
244  Ensmbl       Ensemble                 TRAIN         REP_Sustainability_Goals_Segment
245  Ensmbl       Ensemble                 VALIDATE      REP_Sustainability_Goals_Segment
246  Ensmbl2      Ensemble (3)             TRAIN         REP_Sustainability_Goals_Segment
247  Ensmbl2      Ensemble (3)             VALIDATE      REP_Sustainability_Goals_Segment
248  MBR2         MBR (2)                  TRAIN         REP_Sustainability_Goals_Segment
249  MBR2         MBR (2)                  VALIDATE      REP_Sustainability_Goals_Segment
250  MBR6         MBR (Features)           TRAIN         REP_Sustainability_Goals_Segment
251  MBR6         MBR (Features)           VALIDATE      REP_Sustainability_Goals_Segment
252  AutoNeural2  AutoNeural (Normalized)  TRAIN         REP_Sustainability_Goals_Segment
253  AutoNeural2  AutoNeural (Normalized)  VALIDATE      REP_Sustainability_Goals_Segment
254  Reg          Polynomial (Normalized)  TRAIN         REP_Sustainability_Goals_Segment
255  Reg          Polynomial (Normalized)  VALIDATE      REP_Sustainability_Goals_Segment
256
257                                       False    True     False    True
258            Target Label               Negative Negative Positive Positive
259
260  Replacement: Sustainability_Goals_Segment  2454  3893   66    173
261  Replacement: Sustainability_Goals_Segment  1090  1646   54    37
262  Replacement: Sustainability_Goals_Segment  2603  3958   1     24
263  Replacement: Sustainability_Goals_Segment  1123  1698   2     4
264  Replacement: Sustainability_Goals_Segment  2573  3930   29    54
265  Replacement: Sustainability_Goals_Segment  1113  1686   14    14
266  Replacement: Sustainability_Goals_Segment  2566  3929   30    61
267  Replacement: Sustainability_Goals_Segment  1113  1685   15    14
268  Replacement: Sustainability_Goals_Segment  2428  3866   93    199
269  Replacement: Sustainability_Goals_Segment  1081  1636   64    46
270  Replacement: Sustainability_Goals_Segment   .    2354   1605  2627
271  Replacement: Sustainability_Goals_Segment  404   608    1092  723
272  Replacement: Sustainability_Goals_Segment   .    2405   1554  2627
273  Replacement: Sustainability_Goals_Segment  401   619    1081  726
274  Replacement: Sustainability_Goals_Segment  2488  3835   124   139
275  Replacement: Sustainability_Goals_Segment  1084  1647   53    43
276  Replacement: Sustainability_Goals_Segment  2419  3767   192   208
277  Replacement: Sustainability_Goals_Segment  1069  1607   93    58
278
```

All models fail in precision (0.5 or below) for testing except for Pruned DT, meaning the accuracy in class 1 predictions is not validated as the testing performance does not match training. Custom DT's precision for testing is 0.5. All models can only detect less than 10% (Sensitivity below 0.1) of positive instances except for MBR, making these models useless since over 90% of high likelihood companies are not even detected.

Chosen Model for identifying high likelihood: MBR

For Pruned DT, business can be 67% confident about a positive prediction being true, supported by a validation set. However, 99% of high likelihood conditions are not detected by this model. For MBR, business can be 63% confident about a positive prediction being true only and at least 64% of high likelihood conditions are detectable by model. I will select MBR since a DT that only less than 1% of positive class is not ideal as most companies will be labelled as 0 even if they might have high likelihood, which could cause unnecessary distress to stakeholders. Proceeding with MBR means I must **disregard testing results** and **assume that the training data accurately depicts actual real-world scenarios**, ignoring potential limitations in generalizability with new unseen trends.

Modelling (TARGET=Low Sustainability Goals Segment)

Predicting companies at risk of NOT achieving sustainability goals As this model is just for supplementing the model that predicts high likelihood, I assume the pattern and intricacies are similar. Hence, I reuse the Top 2 models for this classification problem. I will prune the Decision Tree to understand the intricacies in detail.	
DT Did some manual pruning by deleting branches that do not identify low likelihood (class=1). Also experimented with different split values to find the optimal value that looks appealing to interpret and produces high purity of class=1. After pruning all the nodes, some key highlights was that the validation purity of class=1 is not consistent with training, which could make the reliability of claims made from these decisions questioned.	
Interpreting DT Insights from this Decision Tree were either illogical and/or have conflicting performance between train and test (e.g. train=0.6, test=0.3) and/or not supported by enough records. This is the only logical insight that can be presented: 	
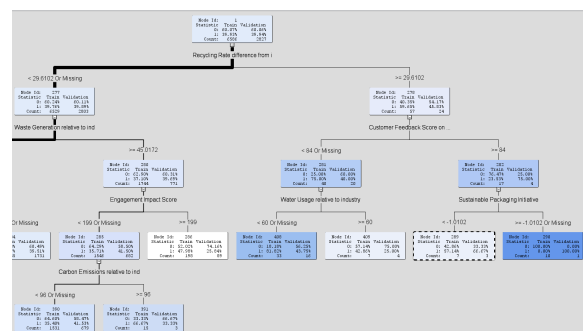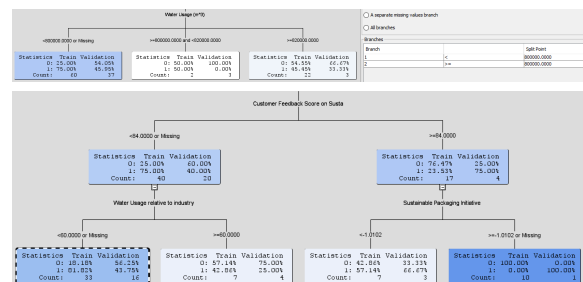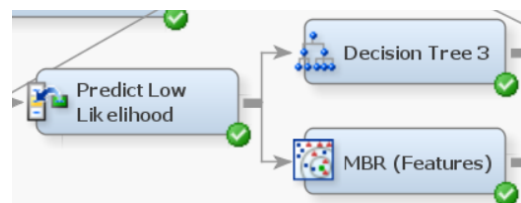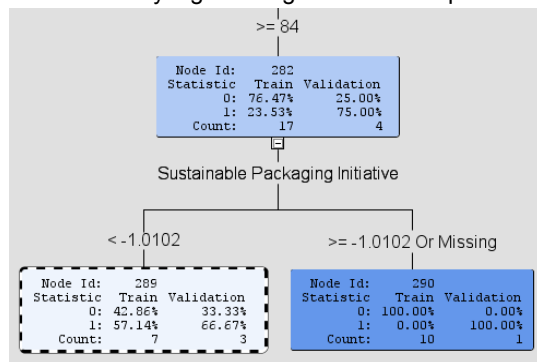
Model Selected: MBR (Better usefulness since more TP were identified)

Based on training data, 100% of all 1048 positive predictions are correct, achieving perfect precision in predicting companies at risk. Having a sensitivity of 0.40, this means that around 40% of all positive instances are identified by the model. Similar to the other selected MBR, testing precision of 0.43 is disregarded as I **assume training data accurately represents the real-world scenario**, hence **placing more confidence in these results than in potentially varied outcomes obtained during testing**.	

DMBA Assignment (JAVEN LAI LE YU)   Page 20

Interpretation and Recommendation

How to interpret MBR

General	
Number of midpoints	8
Profile All	Yes
Cutoff percentage	95
Input Variables	
Number of Inputs	10
Minimum Worth	0.01
Maximum Depth	1
Print Worth Statistics	Yes
Target Variables	
Analysis Role	Segment
Report Variables	
Use Report Variables	Yes
Number of report varia	10
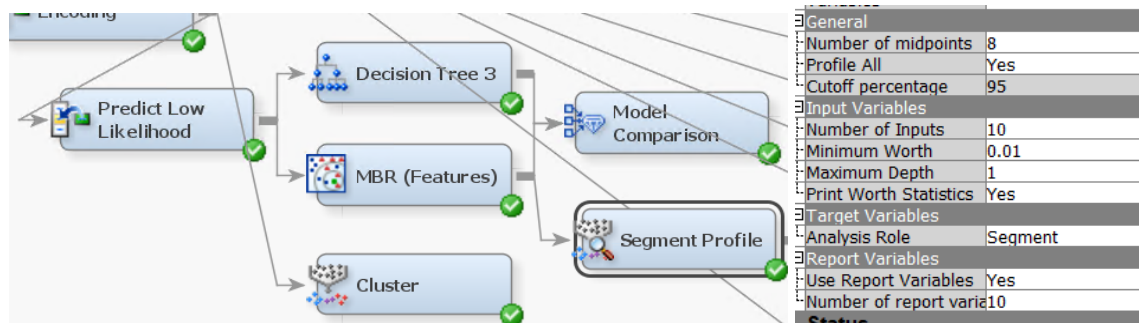
I will interpret MBR using Segment Profile to get the feature importances to determine which factors have most significance in causing high or low likelihood. For the configurations, I changed the analysis role to Segment so that I can get the report to understand how the data is splitted by MBR.

Interpreting factors that leads to high likelihood of achieving sustainability goals

Based on MBR, the key factors are:
1. Energy usage for company
   - How much unrenewable energy is consumed.
   - Consumption relative to industry average.

2. Percentage of Eco-friendly products in a company.

3. Carbon footprint
   - Total tax on a company's carbon emissions.
   - Emission relative to industry average.

4. How much a company invests in sustainability.

Segment Variable	Segment Value	Frequency Count	Percent of Total Frequency
REP_Sustainability_Goals_Segment	0	3959	60.1124
REP_Sustainability_Goals_Segment	1	2627	39.8876

Variable: REP_Sustainability_Goals_Segment Segment: 0 Count: 3959
Decision Tree Importance Profiles

Variable	Worth	Rank
Unrenewable_Energy__MWh_	.002053926	1
Energy_Consumption_relative_to_i	.001776568	2
Eco_friendly_Product_Lines____	.001763858	3
Eco_friendly_Product_Lines_diffe	.001752369	4
Carbon_Emissions_relative_to_ind	.001727656	5
Energy_Tax_difference_from_indus	.001702645	6
Carbon_Tax_difference_from_indus	.001698525	7
Investment_in_Sustainability__US	.001689581	8
VAR35	.001689581	9
Energy_Consumption__MWh_	.001676166	10

Name	Use	Report
Energy_Consumption_relative_to_i	Default	No
Investment_in_Sustainability__US	Default	No
Carbon_Tax	Default	No
Carbon_Emissions_relative_to_ind	Default	No
Unrenewable_Energy__MWh_	Default	No
Eco_friendly_Product_Lines____	Default	No
Eco_friendly_Product_Lines_diffe	No	No

Hoping to resolve or at least reduce overfitting in MBR, I tried reducing dimensionality by cutting down the features to just the few relevant and significant ones.

However, the model still ends up overfitting as the FP is more than TP for testing.

Since there is no hyperparameter tuning to configure for the Scan method as Epsilon and Number of Buckets is only applicable for RD-Tree, and I already discovered the optimal K=2, there is nothing left to do to address the overfitting. I can only draw the conclusion that the issue lies in the dataset since all Decision Trees also overfitted too despite pruning.
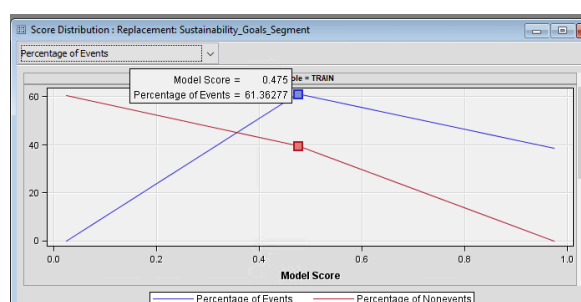
```
118
119     Data Role=TRAIN Target=REP_Sustainability_Goals_Segment Target Label=Replacement: Sustaina
120
121     False       True       False       True
122    Negative   Negative   Positive   Positive
123
124      .          2392       1567       2627
125
126
127     Data Role=VALIDATE Target=REP_Sustainability_Goals_Segment
128
129     False       True       False       True
130    Negative   Negative   Positive   Positive
131
132      404        608        1092        723
```

Train	
Variables	
Method	Scan
Number of Neighbors	2
Epsilon	0.0
Number of Buckets	8
Weighted	Yes
Create Nodes	Yes
Create Neighbor Variables	Yes

Interpreting factors that leads to high risk of NOT achieving sustainability goals

Based on MBR, the key factors are:
1. Waste generated by the company.
2. Carbon tax of the company.
3. Renewable energy used by the company.
4. Eco-friendly products in a company.
5. Recycling rate of the company.

Segment Variable	Segment Value	Frequency Count	Percent of Total Frequency
REP_Sustainability_Goals_Segment	0	3956	60.0668
REP_Sustainability_Goals_Segment	1	2630	39.9332

Variable: REP_Sustainability_Goals_Segment Segment: 0 Count: 3956
Decision Tree Importance Profiles

Variable	Worth	Rank
Waste_Generation_relative_to_ind _N2	.002219084	1
	.002022449	2
Carbon_Tax_difference_from_indus	.001892046	3
Water_Usage_relative_to_industry	.001806451	4
Renew_Energy_difference_from_ind	.001729958	5
Eco_friendly_Product_Lines_diffe	.001716946	6
Renewable_Energy_Use____	.001700829	7
Recycling_Rate___	.001670345	8
Waste_Generation__MT_	.001619185	9
Carbon_Emissions_relative_to_ind	.001600788	10

Since I'm unable to interpret much apart from stating the important features, I will use the Cluster node to learn patterns of Low, Medium, and High likelihood clusters since MBR using Scan works just like clustering and MBR was able to capture patterns to distinguish High or Low likelihood from the rest, with decent precision (for training) and sensitivity. Variables: same as MBR (Features). This allows me to understand exactly what conditions cause a high or low likelihood.
I will cluster using K=3, to split the data in Low, Medium, and High likelihood partitions.

Train	
Variables	
Cluster Variable Role	Target
Internal Standardizatio	Standardization
Number of Clusters	
Specification Method	User Specify
Maximum Number of C	3

Interpreting (using Cluster Node) what determines likelihood of achieving goals
Unrenewable Energy Impact:
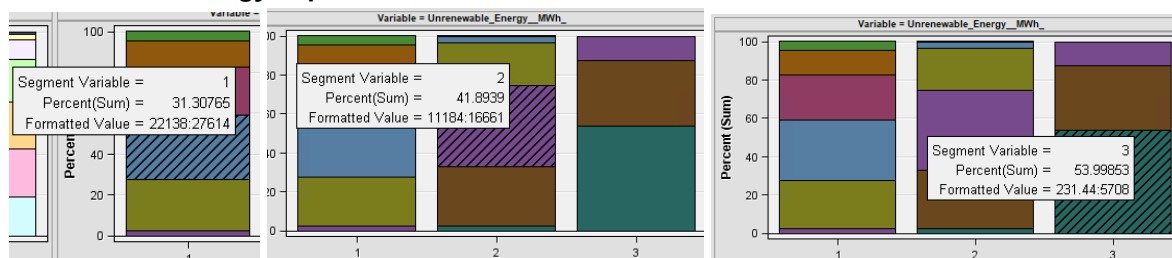
- **Low likelihood companies use at least 11,184 MWh of Unrenewable energy**.
- A company is at a **very high risk of falling into low likelihood if they exceed 22,138 MWh** because no noticeable 22138 (teal) instances in cluster 3, and **no noticeable instances in cluster 2 and 3 that exceed 22,138 MWh** (red onwards).
- Companies **can only achieve high likelihood if their unrenewable energy consumption is below 11,184 MWh**.
- Can be **certain of a high likelihood if their unrenewable usage is around 230 MWh** since Medium only has 2% of such companies, and none of such companies fall in Low.

Renewable Energy (%) Impact:

Only 0.8 Low likelihood (Segment 1) have renewable energy usage of around 62.5%.

To address the need of heavy electricity usage required for operations, companies should try to have at least 62.5% of their energy consumption using renewable energy to avoid falling into the risk cluster that is not likely to achieve sustainability goals in the next 5 years.
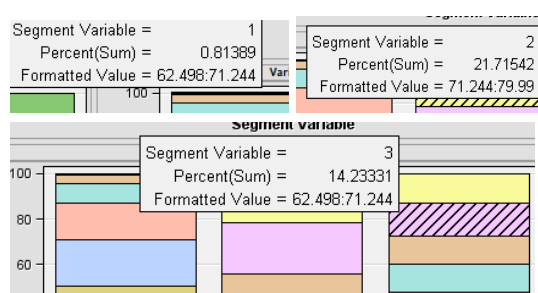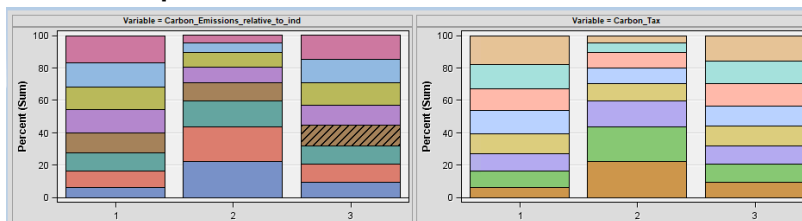
Eco-friendly product lines (%) and Investment in Sustainability:
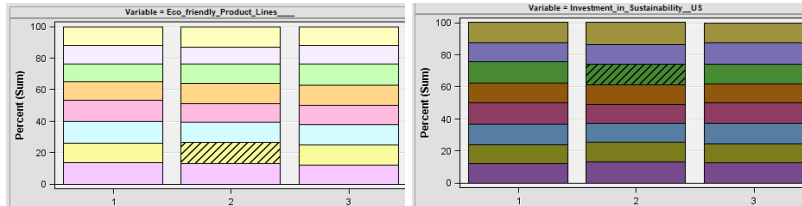
**Waste Management and recycling efforts:**
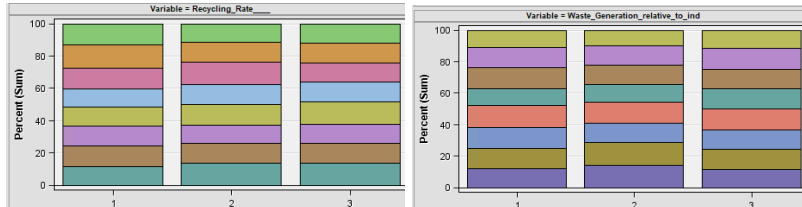
Carbon Footprint:

No pattern between Low and High clusters as proportion of each colour are closely similar, suggesting these factors are not impactful enough to distinguish low and high likelihood companies.

Conclusion:

Energy Consumption seems to be the main factor that determines the likelihood of achieving low or high likelihood based on the Clustering model.

Additional:

Likelihood of achieving sustainability goals is NOT influenced by Industry, meaning that dropping those 2 industries in Data Cleaning earlier will not have impacted the analysis (assuming if those 2 industries are not errors that do not belong in the dataset). This is interpreted by the fact that there isn't any prevailing industry in either cluster, or any industry completely absent in any cluster.
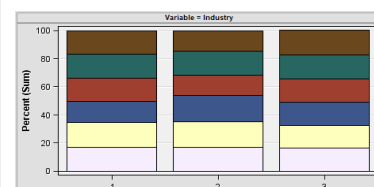
Recommendations to achieving sustainability goals in the next 5 years

I will use strong paths in Decision Trees to justify the recommended plan of actions since I can use the train and test purity to show confidence in claims.

If your company consumes over 5484MWh of unrenewable energy, besides trying to increase reliance on renewable energy if possible, you could try reducing the amount of waste generated by reducing and reusing as much as possible. Based on the article: 13 ways a company can reduce waste, some ways include going paperless, using energy and water efficient appliances, opting for appliances made with environmentally friendly materials, and encouraging recycling by adding recycling and composition bins in the workplace.

Based on training results, this worked for all 6 companies with this scenario.
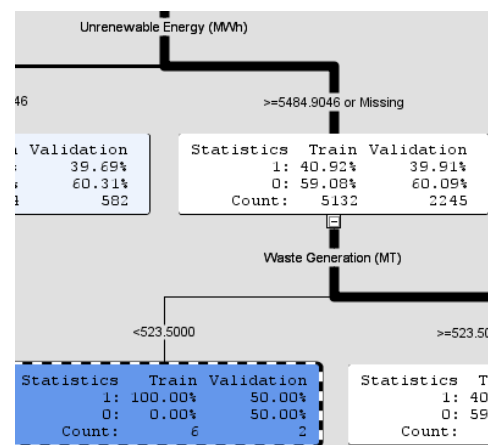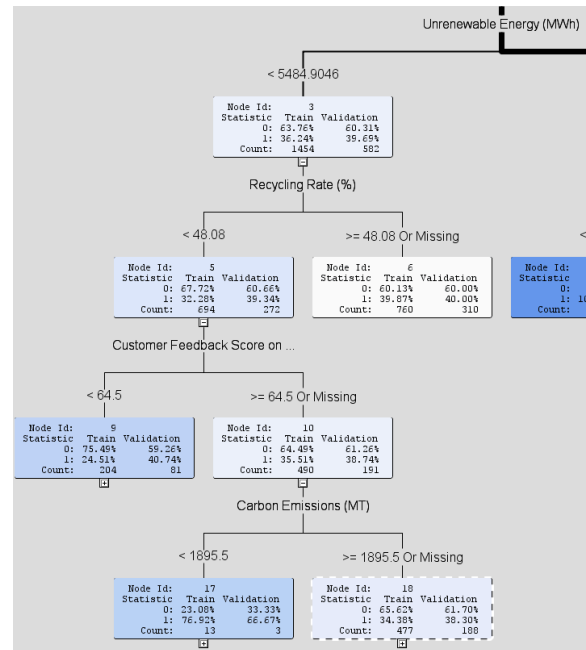

[Pruned DT, Precision Train: 100%, Test: 67%]

If your company consumes below 5484MWh of unrenewable energy, has recycling rate below 48% with customer feedback score of at least 64.5, but has carbon emissions below 1895.5 MT, your company has around 70% chance of having high likelihood in achieving sustainability goals.

This insight is supported by 16 companies, 3 of which are in the validation set. Although few companies have these exact conditions for the decision tree to assess, the confidence in this claim is quite high as the purity of class 1 in the node is high and the claim also sounds logical.

Hence, companies can try replicating these conditions to have high confidence that they can achieve their sustainability goals in the next 5 years by reducing reliance on unrenewable energy and emitting carbon, and take proactive steps to raise customer feedback scores.

[Pruned DT, Precision Train: 100%, Test: 67%]

If a company consumes below 5480 MWh of unrenewable energy AND has carbon emissions of 90% below industry average, the company has around 60% chance of having high likelihood. This is supported by the validation set too. However, beyond 95% below industry average, the pattern shifts, and it's assumed that these companies falsely report their carbon emission as part of greenwashing to boost their company reputation.

I **assume that high carbon emitters are large companies** since they generate a lot of carbon from their heavy operations, but the fact that they still fall in the minority group of low unrenewable energy users suggest that <u>as long as a company takes initiative to reduce non renewable energy usage, they should be on track to achieve their sustainability goals</u>.
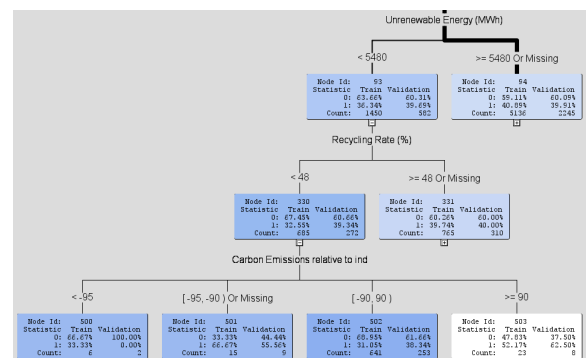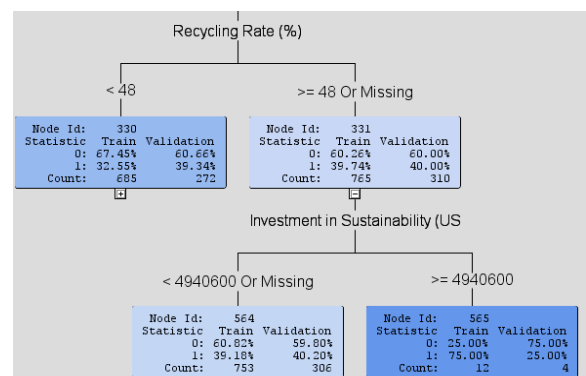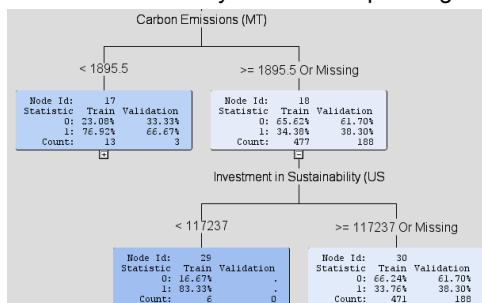

[Custom DT, Precision Train: 65%, Test: 50%]
Another interesting insight is that high carbon emitters (90% above industry average) are more likely to achieve high likelihood than mid-range (-90 to 89) carbon emitters.

Investing heavily in sustainability is not a guarantee to achieve sustainability goals because the validation set shows that companies who invested a lot still did not fall in high likelihood.

This branch from DT3 also supports this insight as companies who spent below threshold were likelier to achieve sustainability than those spending more:
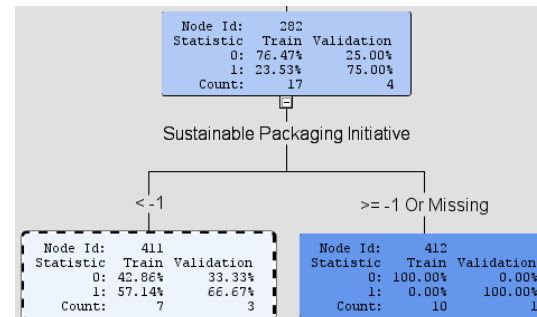

[Custom DT, Precision Train: 65%, Test: 50%]
Hence, **companies should focus on taking measurable actions** like switching to renewable alternatives to reach their sustainability goals, instead of solely pouring money to invest in sustainability.

There is a very **weak claim that companies that have 2 less sustainability packaging initiatives than its industry average** have a slightly higher risk of falling into low likelihood of achieving sustainability goals. Hence, companies may consider having at least 1-2 more sustainability packaging initiatives than its industry standard if they have enough resources to do so. However, this should not be prioritised since it's a very weak claim with little proof.	 [DT predicting low likelihood as class=1]

Summary of recommendations to achieving sustainability goals:
1. Reduce non renewable energy usage by increasing reliance on renewable energy.
2. Reduce and Reuse energy, water, and waste. Opt for environmentally sustainable appliances and products. Include recycling and composition bins to encourage recycling in the workplace.
3. Don't just invest in sustainability using money. ACT on the goals by doing step 1 and 2 above.
4. Implement sustainable packaging for products whenever possible to increase sustainability efforts and raise customers' impression on the company - thereby leading to increased customer feedback score on sustainability when customers see the company taking actions.

Conclusion of Modelling:
No predictive model is able to deliver acceptable performance for both training and testing, hence no model is ready for deployment. My best model is MBR with Feature Selection, and what sets it apart is its Sensitivity in predicting positive classes because while all models struggle to even identify 10% of positive class, MBR is able to achieve at least 60% for both training and testing. The flaw of MBR is that it's overfitted by 23%, hence lacking generalizability with unseen data because it only achieved 40% accuracy in predicting positive class for testing. In terms of accuracy in predicting positive class, Pruned DT is the only model with passable precision in testing of 67%, but with sensitivity below 1% which makes it useless since it misses over 99% of high likelihood instances.

Limitations of analysis
Flaws:
1. **Data Quality:** Accuracy of analysis is highly dependent on the data provided. Any inaccuracies, biases, or missing information could lead to flawed conclusions. As the dataset had little info given in metadata, I am unable to assess the reliability of the information beyond the dataset itself (e.g. collect methods, are values edited, accuracy of details provided by company).
2. **Small datasize**: There are too few records that fall in certain nodes for Decision Tree to validate the insight. Hence, I proceeded with the assumption that those few companies reflect the actual trend if data size were to scale up. This issue of having a small data size doesn't really hinder analysis, but having more data in these nodes would boost the credibility of the claims further.
3. **External factors** like what steps the company took, what measures they enforced in their company, what the investment in sustainability was spent on, or seasonal trends.
4. **Insights are not validated or backed up**. Hence, patterns may vary in real-world scenarios due to poor accuracy performance when the model was tested with a validation set.
5. **Time Sensitivity**: Sustainability practices and goals evolve over time which causes patterns in Low and High likelihood companies to change, but the models built did not account for such changes. Hence, trends observed during this analysis period may not be indicative of future scenarios.
6. **Varying goals in Companies:** Different companies may have distinct sustainability challenges and opportunities. The analysis might not adequately consider industry-specific nuances, potentially leading to generalizations that do not hold across all sectors.
7. **Human Factor:** This analysis does not account for the human element, such as changes in leadership, company culture, or shifts in employee behavior, which can affect a company's ability to achieve their sustainability goals.

Analysis could be improved if:
1. **All necessary information to fully understand the Target is provided** (such as how it was measured or calculated), because I would be able to integrate much more external info and data into analysis. If business could provide the methodology for deriving the value, it could be beneficial in improving this analysis.
2. The **"Sustainability Goals" that the companies set are provided**. I built the models with the assumption that these goals set are related to the ESG pillars of sustainability. If companies provided what their sustainability goals are, the analysis would be more accurate as I have more data to work with and tailor analysis specifically to certain companies to resolve their unique issues.

Considerations:
The poor performance where precision below 0.7 (acceptable range: 0.7-0.9) and inconsistent performance in train-test due to poor generalization may suggest that:

    Scope of Analysis: The data is lacking sufficient pattern (true key drivers not in dataset) for machine learning.

    OR

    Problem solvable without Machine Learning: This problem of determining if a company is likely to achieve their sustainability goals might not even require ML. Since the company should know their own sustainability goals, uncovering the causal relationships hindering progress of achieving sustainability goals and resolving them directly guarantees success, making it redundant to perform machine learning if the problem can be resolved logically by addressing the root problem.
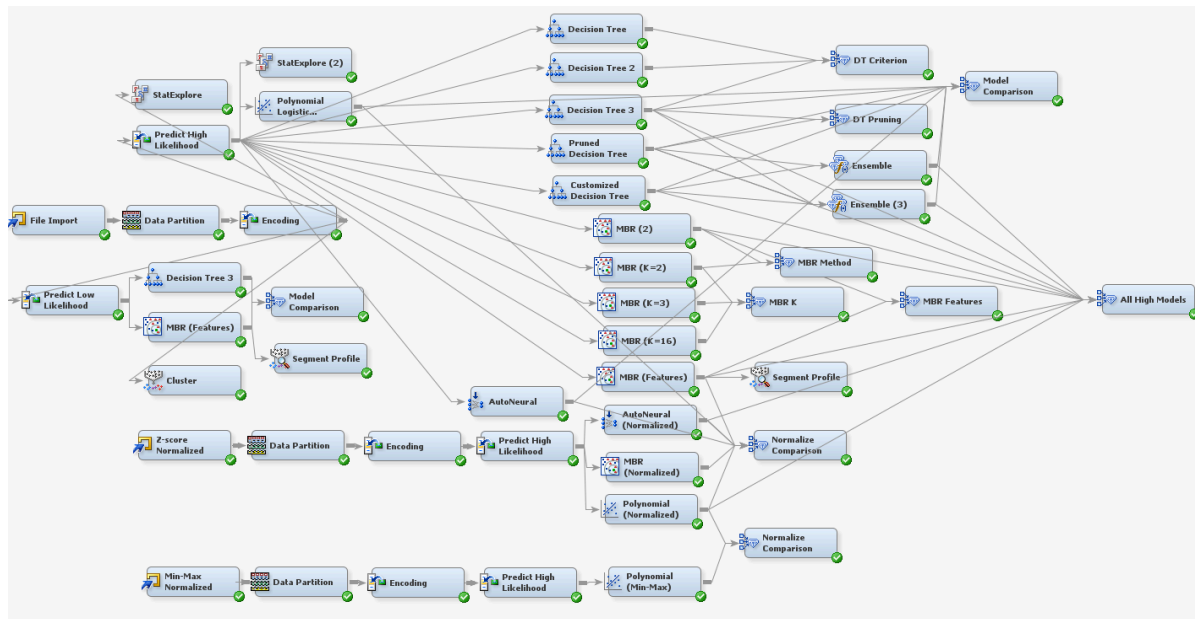
    OR

    Problem beyond scope of Machine Learning: The nature of achieving low or high likelihood is uncertain and realistically impossible to be predicted with decent accuracy and certainty (similar to stocks).

Conclusion
- It depends on what are the sustainability goals set by the company.
- Even though carbon, water, and waste is not significant based on analysis, but companies should still practice conventionally socially responsible practices.
- This machine learning analysis should only act as a reference to support a company's existing planned course of action, and should not be the main one.

***** END OF ASSIGNMENT *****

Overall SAS EM Miner workflow

References

https://chat.openai.com/share/b01c4d76-57f2-4f02-b3f4-c37cd160e389 [Constructing Background and Purpose of Project writeup]

https://chat.openai.com/share/9b4e8659-977f-4626-9816-8bb725381230 [What features I should use for evaluating Sustainability of a company]

https://www.rgconsult.com/find-a-consultant/ [7 Key industry sectors]

https://www.statista.com/statistics/1427506/workers-transport-modal-share-for-commuting-united-states/ [Modes of Transport]

Justification to drop missing target:
https://stackoverflow.com/questions/73644442/missing-value-in-target-variable

Why Pharmaceutical and Healthcare are different industries:

https://www.efpia.eu/media/25672/understanding-the-working-relationship-between-the-pharmaceutical-industry-and-healthcare-professionals.pdf

https://biglanguage.com/blog/what-are-the-differences-between-health-care-medical-life-science-and-pharmaceutical-translations/

Carbon Emission of oil companies: https://www.visualcapitalist.com/companies-carbon-emissions/ [To justify outlier value in carbon emission column is an error]

Deciding whether to drop rows with problematic column:
https://towardsdatascience.com/7-ways-to-handle-missing-values-in-machine-learning-1a6326adf79e

Justification for dropping rows: https://gutcheckit.com/blog/veracity-big-data-v/

Corporate Sustainability explained by Forbes:
https://www.forbes.com/sites/forbesbusinesscouncil/2023/10/19/how-companies-can-achieve-their-sustainability-targets/?sh=66d04f2443b7

ESG explained:
https://www.ehs.com/esg-central/?keyword=esg%20risk%20assessment%20template&msclkid=9d32f9e1d9ec1d480ba39c7276a13750&utm_source=bing&utm_medium=cpc&utm_campaign=Materiality%

20ESG&utm_term=esg%20risk%20assessment%20template&utm_content=Materiality%20-%20Broad [VelocityEHS]

Actions taken by world leaders to enforce sustainability practices on companies: https://www.weforum.org/agenda/2022/05/global-sustainability-corporations-lead-net-zero/

Carbon Tax per metric tonne based on NEA: https://www.nea.gov.sg/our-services/climate-change-energy-efficiency/climate-change/carbon-tax [External data integrated to calculate tax]

Energy Tax on different countries by Eurostat https://ec.europa.eu/eurostat/databrowser/view/env_ac_taxener/default/table?lang=en [External dataset integrated to calculate energy tax]

Assumptions of Linear Regression: https://www.statology.org/linear-regression-assumptions/ [To justify whether doing Linear Regression is appropriate]

What is predictive modelling: https://www.netsuite.com.sg/portal/sg/resource/articles/financial-management/predictive-modeling.shtml

All about supervised machine learning: https://www.ibm.com/topics/supervised-learning#:~:text=Supervised%20learning%2C%20also%20known%20as%20supervised%20machine%20learning%2C,that%20to%20classify%20data%20or%20predict%20outcomes%20accurately.

Research on non-linear classifier models: https://dataaspirant.com/non-linear-classifiers/

https://chat.openai.com/share/82c29d8d-244d-4a88-a5b9-9403b6d3d0c7 [Writeup of Decision Tree, Neural Network, and KNN]

https://iq.opengenus.org/precision-recall-sensitivity-specificity/ [Understand various evaluation metrics for classification]

How multicollinearity influences performance and interpretability in Machine Learning: https://www.linkedin.com/pulse/what-multicollinearity-how-affects-model-performance-machine-cherukuu/

Curse of Dimensionality in Clustering: https://courses.cs.cornell.edu/cs4780/2022fa/slides/curse_of_dim_clustering_annotated.pdf

Blog about Polynomial Regression: https://datascience.stackexchange.com/questions/21896/can-the-linearly-non-separable-data-be-learned-using-polynomial-features-with-lo

Research on Logistic Regression in SAS EM: https://documentation.sas.com/doc/en/espcdc/6.1/espan/p1iyy8xvfytolsn16djcp243wkx3.htm [learn how to configure and use node]

Research on Decision Tree: https://support.sas.com/content/dam/SAS/support/en/books/decision-trees-for-analytics-using-sas-enterprise-miner/63319_excerpt.pdf

Research on using MBR: https://documentation.sas.com/doc/en/emref/15.3/n1fevkin0iu4cxn1khv8ow9r8pmj.htm

Which models benefit from scaling and why: https://machinelearningmodels.org/machine-learning-models-that-require-feature-scaling/