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.
- Predictive models 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.

```
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)
```

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

```
10100
```

```
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 those columns 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.
4. There **might be outliers and skewed distributions in columns** like Carbon emission and Energy consumption as certain row's value can vary drastically. e.g. 41686, 40231, 3973.

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()
```

ɔply Chain itainability 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 investigate categorical columns and numerical columns separately as it's impractical to manually inspect every individual value possible for numerical columns.

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	Cc Enę
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	Healtchare	1000000.0	118745.0	19214.0	19333.0	39.16	64.0	Car	77.13	Yes	
2122	2123.0	Company_2123	Healtchare	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 exact unrealistic value for a particular column *(underlined in blue in screenshots above)*, only difference being the value for Finance is negative. This article shows that 20 oil companies collectively produced a total of 1.35 million **$MtCO_2e$** carbon emissions during 1965 - 2017, suggesting that it's improbable for 1 company alone to produce 1 million MT of carbon emission annually, which means these **values are errors** when entered into dataset. Water usage of negative 1 million is also definitely an error (possible explanations: system entry error, need to fill in value for all columns when entering a record but water usage is unknown hence user input -1 for water usage) because a company cannot have negative water usage.

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. Also, the fact that all companies in the same industry have the exact same emission value is unlikely to be true, making it justified to drop them so that the accuracy and reliability of my dataset and analysis will not be compromised; I **don't trust these rows** since 1 column is already wrong, the other values could likely be inaccurate or false too.

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 (imputing 0 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']]
```

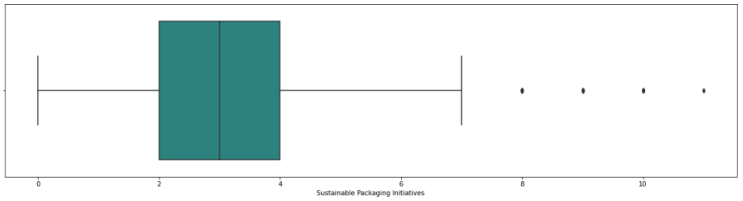
	Record ID	Carbon Emissions (MT)	Water Usage (m^3)	Energy Consumption (MWh)	Waste Generation (MT)	Recycling Rate (%)	Supply Chain Sustainability Score	Renewable Energy Use (%)	Community Engagement Score	Eco-friendly Product Lines (%)	Investment in Sustainability (USD)	Customer Feedback Score on Sustainability	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 as these outliers. 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 users (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:

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:

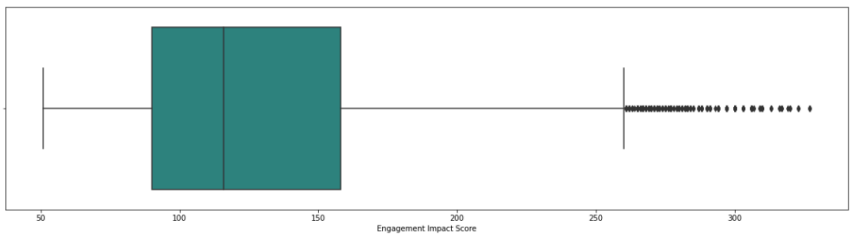
We can address the Social Domain using Customer Feedback Score and Community Engagement Score. However, as there are already many Environmental features and having too many features can hinder model performance, I decided to engineer 1 feature to get an overview of a company's social impact. Hence, I 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.

```
# 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
```

```
# 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.

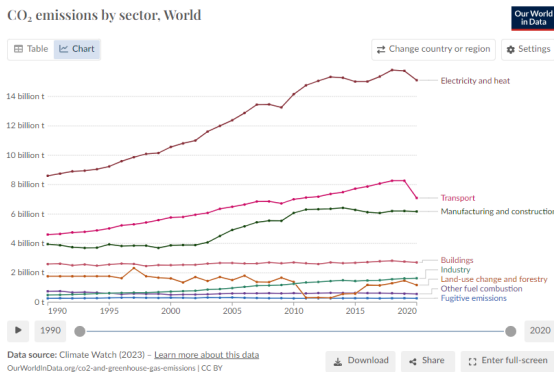
Although all ESG Pillars have been addressed, the features could still benefit from further improvements.

Balancing comparison between various industries:

Based on an [emissions analysis by Our World In Data Various](), **industries have varying emissions as they 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 between diverse sectors inequitable.

Diagram shows how CO2 emissions varied for different industries across the years, revealing specific trends within each industry, supporting the fact that it's inequitable to compare companies across different industries.

I believe this trend is likely present within the dataset too…



Hence, I will **represent the company's emissions as a percentage relative to the industry average** for fair and meaningful comparison for insights.

The code calculates the mean emissions, water usage, energy consumption, and waste generation for each industry and adds it into the dataset. Then, a new column is created by calculating the difference between a company's emission from the average of its industry, normalizing it to a percentage out of 100.

```
# 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)']]

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

# 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
```

For Energy and Carbon Taxes, Renewable Energy Use (%), Recycling Rate (%), Eco-friendly Product Lines (%), and Sustainable packaging initiatives, I expressed them as the difference compared to the industry average values.

Expressing these new features as raw difference instead of a percentage makes it more interpretable since the original features already represent a percentage, and introducing a percentage of them would make it confusing.

```
# 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']]

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

# 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'])

# 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'])
```

For investment in sustainability, I simply compared with the average investment amount for overall comparison into how much more the company spent on sustainability investment than an average company (easier to explain).

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

These engineered features enable a nuanced examination of how usage/emission of carbon, water, energy, waste, and efforts to reduce impacts relative to the industry's average (as a performance benchmark) influences a company's ability to achieve its sustainability goals.

Data is ready to be used for Modelling:

Clear useless columns made while creating new features

```
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'
```

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

...upply Chain ...ustainability 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

```
# 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 correlation between numerical columns and Regression Target, interpreted by very low R-Squared value. Highest R-Square of 0.019 indicates that Waste Generation is 1.9% related to Target, suggesting an insignificant relationship between them. 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. But first, let's understand how the Regression and Classification Target are related.

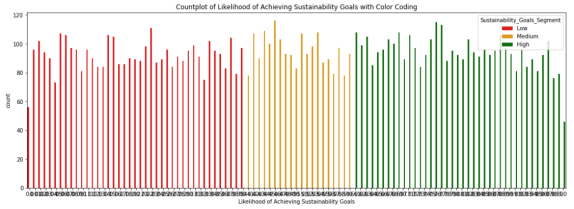
```
df.corr()['Likelihood of Achieving Sustainability Goals'].sort_values(ascending=False)
```

```
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
```

Referring to the graph on the right, Red = 0-0.39 (Low likelihood), Yellow = 0.4-0.6 (Medium likelihood) Green = 0.61-1 (High likelihood). 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 since the range for this class is smaller.

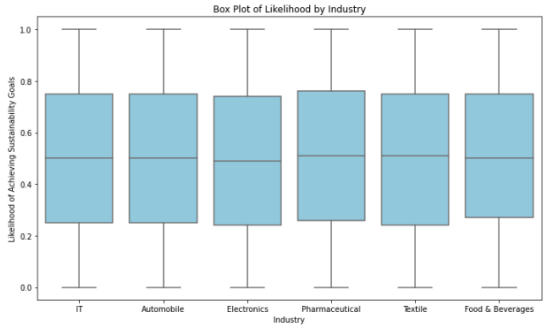
To use the classification target, I can either group Low and Medium as 0 and High as 1 (to identify companies with high success rate and learn factors that lead to their success), or Low as 1 (to identify and alert companies at risk of not achieving sustainability goals).

```
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

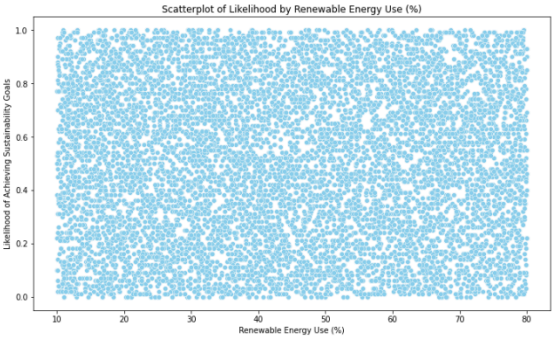



```
# Find categorical columns
categorical_columns = df.select_dtypes(include='object')
categorical_columns.drop('Company Name', axis=1, inplace=True)

# 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 unique value in the column are closely similar for all columns (looks the same as box-and-whisker shown), implying no pattern with Target making categorical columns useless as predictors due to lack of variability.

Relationship in Target & num features



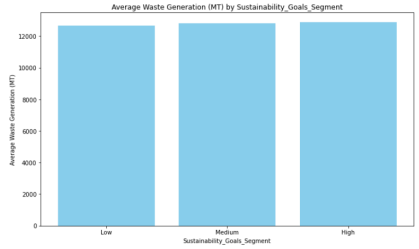

```
# scatterplot to identify linear relationship between numerical features and target
numerical_columns = df.select_dtypes(include=['float64', 'int64'])  # Find numerical columns
fig, axes = plt.subplots(nrows=len(numerical_columns.columns), ncols=1, figsize=(10, 6 * len(numerical_columns.columns)))

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]

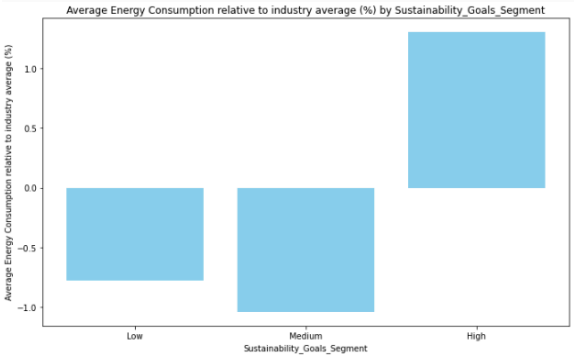
Scatterplot of feature against target show no linear relationship between regression target and any numerical features as all scatter plots have no visible trendline to suggest correlation. The scatterplots mostly resemble the photo on the left, where points are scattered all around with no clear trend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

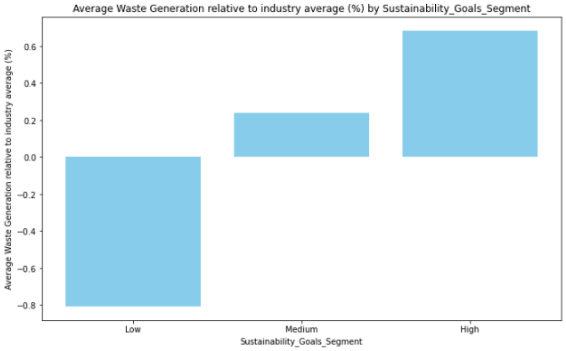
Pattern in Class Target & num features

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



Except for features compared with industry.

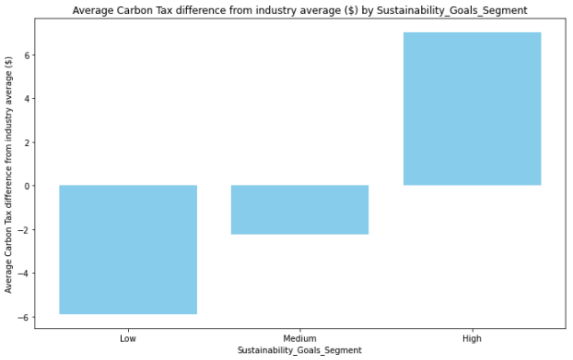


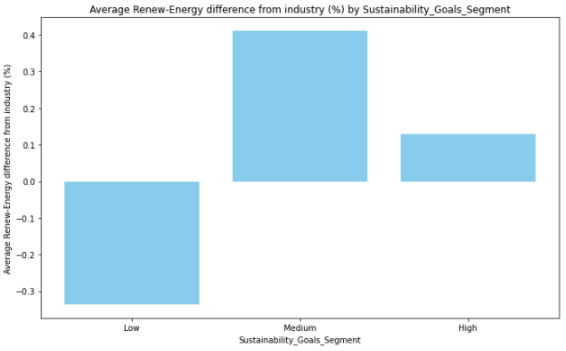


All features created in features engineering that are compared with its industry average produced a similar chart where each class has a varied average value from another class, indicating a distinct, underlying pattern and trend within each class.

```
# 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]





Having a distinguished pattern in each class makes these features useful as predictors to predict either Low or High class since the other classes skew the opposite direction, allowing models to segregate them. This **proves that the engineered features will be useful in improving model performance** (since they have a varied, distinct trend for each class for models to capture) while also being more meaningful by enabling fair comparison despite varied industries.

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**.
- **No linear relationship between numerical features and target**, implying that determining likelihood requires uncovering intricate pattern beyond straightforward linear approaches (e.g. more carbon emission = increased likelihood), making models like **Linear and Logistic Regression unfeasible** 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**.

Predictive Modelling

Chosen Problem Type

I will **only build Classification models** since the patterns between regression target and features are extremely weak and insignificant, foreshadowing lousy and unpassable performance for Regression Models. Since Classification only has 2 possible outcomes (0,1), the accuracy will definitely be better than Regression, thus no point trying regression when classification is already able to identify High or Low likelihood and address problem.

Chosen Model Type

Based on IBM's article on supervised learning, there are 3 types of machine learning: Supervised, Unsupervised, and Semi-supervised. Since the data I have is fully labelled (label refers to having a Target column), I will proceed with supervised modelling because my aim is to let the predictive model learn what factors, under what conditions, lead to High or Low likelihood by letting the model predict the Target/Labelled value.

Candidate Models

After reading Dataaspirant's article, I have shortlisted the following models as they are appropriate for my dataset and my business problem:

1. **Logistic Regression with Polynomial Terms**
   Since its discovered that there is no clear linear relationship between features and target, 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.

2. **Decision Tree** (DT)
   Able to capture intricate patterns. Easy to interpret and explain as I can visualise the model's tree to understand the decision making process.

3. **Neural Network** (NN)
   Highly advanced model able to capture deep nuances within data, known to deliver high performance (accuracy). However, it is very hard to interpret since it's a black-box method. This model will set the ceiling for how well a model can possibly perform on the dataset, but will not be deployed due to lack of transparency (unable to explain to business how the predictions are derived).

4. **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. Allows me to understand what leads to High or Low likelihood by studying how the model clusters the data.

About Models

What is Logistic Regression:

Transforms predictors into a linear equation of independent variables (predictors) and their coefficients using the logistic function (sigmoid function) that produces an output ranging between 0 and 1, representing the probability of the event occurring. The threshold for positive class is often 0.5 and anything below is labelled as 0.

Assumptions:

- **Linearity of the Logit**: Assumes log-odds of the dependent variable are a linear combination of the independent variables.
- **Independence of Errors**: Assumes that errors between observations are independent of each other; occurrence of an event for one observation does not affect the probability of occurrence for another.
- **No Multicollinearity**: Logistic Regression assumes that there is little to no multicollinearity among the independent variables as multicollinearity makes it challenging to determine key drivers.
- **Large Sample Size**: Needs to be trained with sufficient data to ensure stable and reliable parameter estimates (coefficients of each predictor).

How LR can be improved:

- **Handling Multicollinearity**: Perform feature selection to remove overlapping features, or combine them together as one feature.
- **Regularization**: Implement regularization techniques to prevent overfitting and improve model generalization like L1 regularization (Lasso) and L2 regularization (Ridge).
- **Scaling Predictors**: Outliers in the dataset can cause noise which may negatively influence parameter estimates. Scaling dampens the effects of outliers, ultimately improving model performance.
- **Polynomial Features**: help capture complex patterns within data by introducing interactions between features (interaction = feature1*feature2). Higher-degree polynomials can capture complex patterns but may also lead to overfitting, hence I need to find the ideal degree that doesn't overfit.

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. Decision Trees are able to work with features in categorical form, so I do not need to one-hot encode Industry column.

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 to determine target.

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.
2. **Ensembling** (Combining multiple Decision Trees to form Random Forest) can enhance performance by utilizing predictive power of multiple models. Potentially improved performance, but computationally expensive to run. Random Forest uses bootstrap sampling to create multiple subsets of the training data, introducing diversity among the individual decision trees thereby improving performance.

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:

1. **Complex Patterns**: 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).
- **Dimensionality Reduction**: KNN can suffer from the curse of dimensionality where high number of features lead to decreased performance. Hence, performing features selection or PCA to reduce dimensionality could improve model performance.
- **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_Initiati	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:

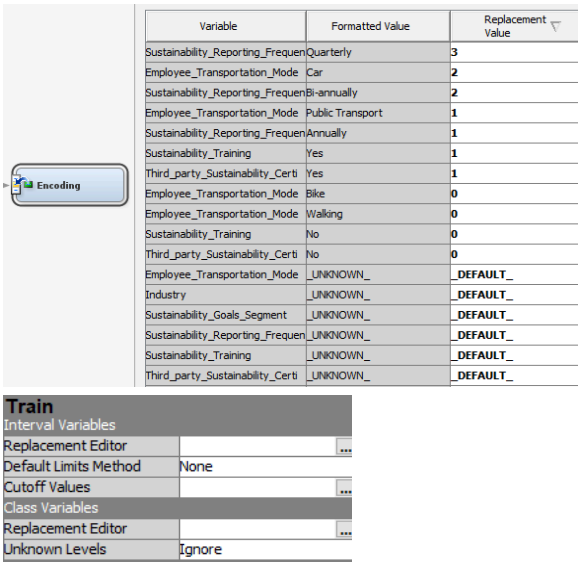
- **Employee Transportation Mode:** Car as 2, Public Transport as 1, and Walking and Bike as 0. (Ranking of environmentally harmful transportation mode most frequently used.)

- **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.

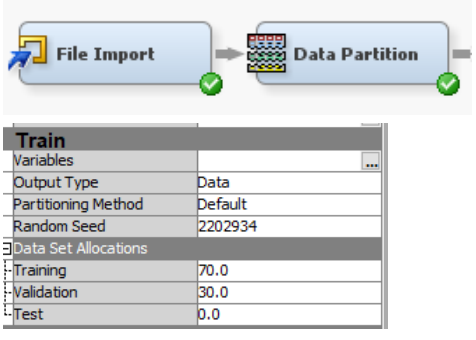
Encoding is needed because NN and KNN models only accept predictors in numeric form.



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.



Encoding Target column

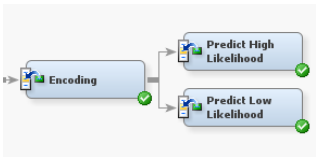
To better understand the key drivers leading to increased probability for a company to achieve their sustainability goals, my predictive models are designed to solve these questions:

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_

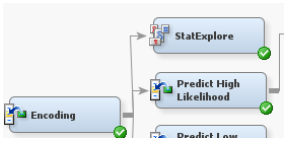
Replacement Editor-WORK.OUTCLASS

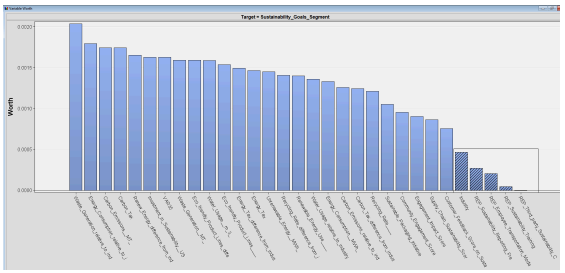
Variable	Formatted Value	Replacement Value
Sustainability_Goals_Segment	Low	1
Sustainability_Goals_Segment	High	0
Sustainability_Goals_Segment	Medium	0
Industry	_UNKNOWN_	_DEFAULT_
REP_Employee_Transportation_Mode	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Reporting_Fre	_UNKNOWN_	_DEFAULT_
REP_Sustainability_Training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

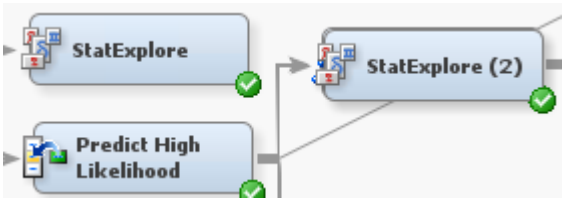
Features Selection

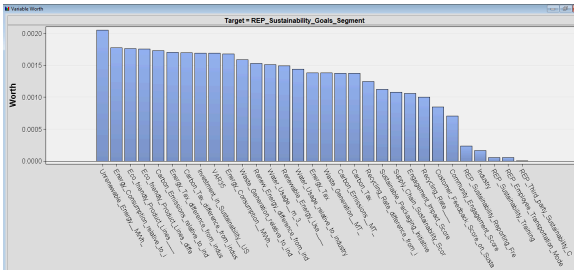
StatExplore to identify useful features for predicting high likelihood class.

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 introduce Polynomial Terms so that LR will work with non-linear assumptions.

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.

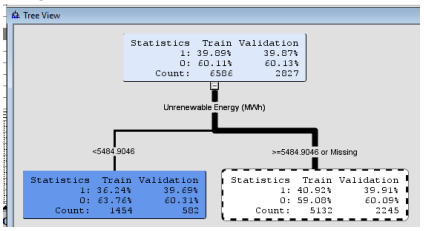
Name	Label	Use	Report	Role	Level
Carbon_Tax	Carbon Tax	Yes	No	Input	Interval
Water_Usage_relative_to_industry	Water Usage relative to industry	Yes	No	Input	Interval
Energy_Tax	Energy Tax	Yes	No	Input	Interval
Carbon_Emissions_relative_to_ind	Carbon Emissions relative to ind	Yes	No	Input	Interval
Unrenewable_Energy__MWh_	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	

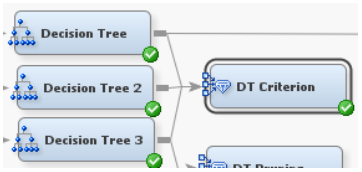
Decision Tree Classifier

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_)

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

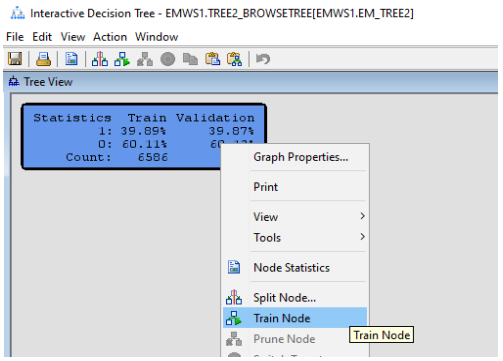
This is done to address overfitting, to make the Decision Tree more generalized to perform equally well with unseen testing data.

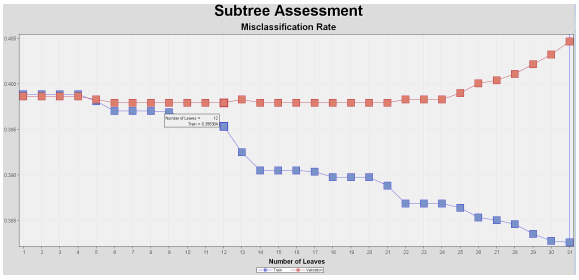
Referring to the screenshot below, pruning prevents overfitting since train and test misclassification rate is more consistent. However, this is insignificant, so I conclude that pruning is not important since the DT doesn't face any severe overfitting performance issue without pruning.

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

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

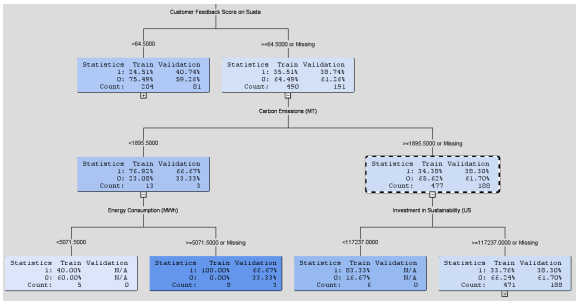




Interpreting DT3

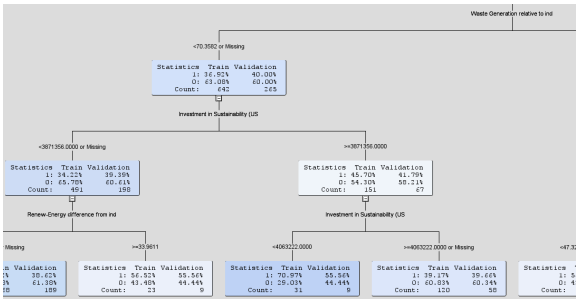
There are some interesting insights like how companies that have low carbon emission but consume more energy have high likelihood (bottom left branch).

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.

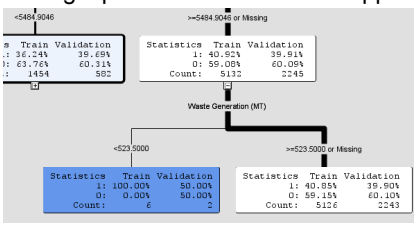
For companies generating less than 70.4% waste than industry average, the optimal investment is $3,871,356 to $4,063,221 based on DT3.



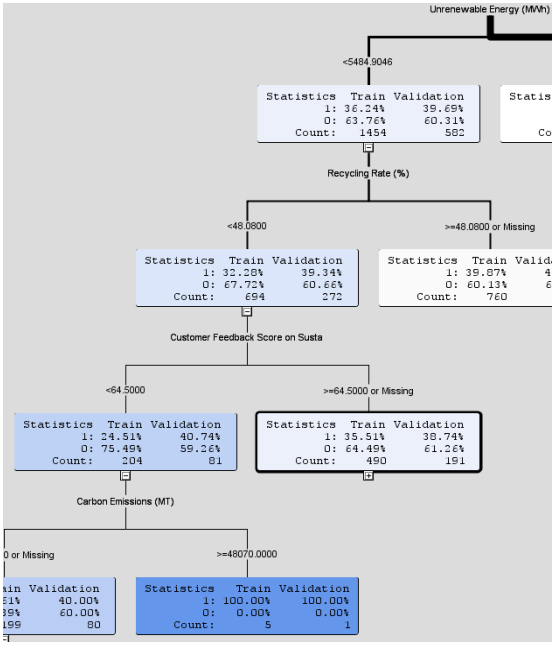
Interpreting Pruned DT

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

The other insight is only supported by 8 instances, making it possible for it to have happen by chance:



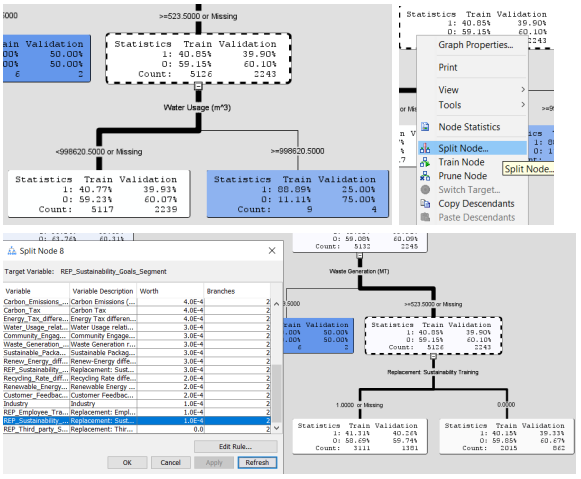
Although this pruned DT might lead to most optimal performance, it's quite useless since the insights it generates are meaningless and unable to concretely distinguish key factors that lead to high likelihood.



Manual drill-down to mine more insights

I noticed there is still a lot of data (5126+2243) within this node as the majority of the data falls in this leaf, revealing potential to dig for more patterns.

I tried all variables to see which feature produces a split that identifies class=1 with high purity. The best result found was Water Usage (has highest split Worth also) but the problem with this insight is that the validation performance does not match up with training which suggests that the high purity in training occurred by chance. Additionally, this insight doesn't logically make sense since it's ridiculous to tell businesses they need to increase unrenewable energy, waste generation, and water usage to be more sustainable. Hence, I will not continue splitting past this node as all potential splits are meaningless.

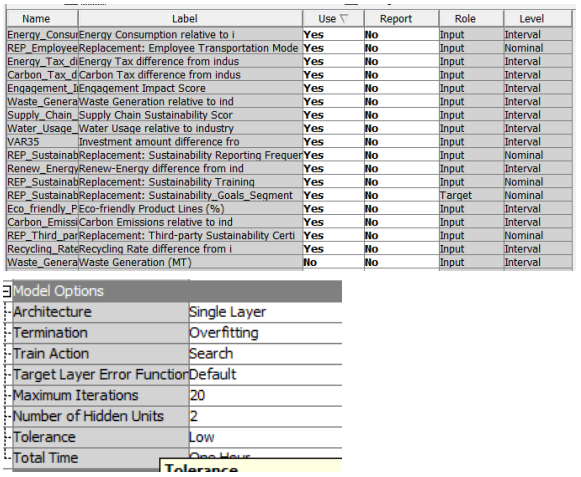


Autoneural

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_Ti	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	

Customized Decision Tree

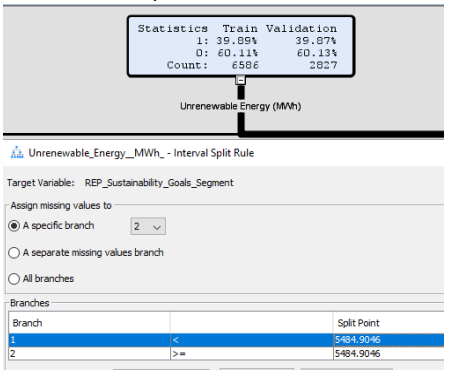
I will now prune the DT with business understanding and performance optimization in mind. As this model is meant for identifying class 1 to recognize what conditions and factors lead to achieving sustainability goals, I will focus my pruning to generate leaves that can identify class 1.

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.

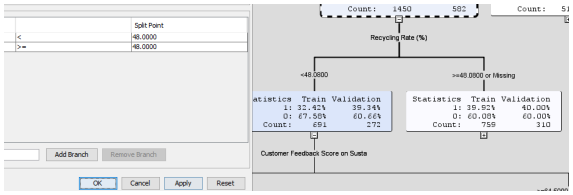
First split is Unrenewable Energy. To make the value more appealing to present to business users, I will round off the split value to 5480.



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

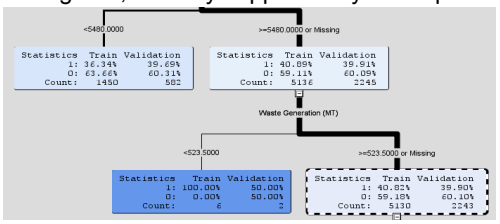


I changed the split value from 48.08 to 48 for business appeal.

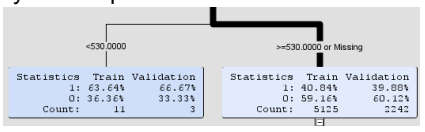
After experimenting with different features and splits, I decided to split using carbon emission relative to industry as I spotted some target=1 which could be linked to business understanding. Here, I subsetted carbon emissions into heavy carbon emission companies (90% and above their industry average) 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**

RIGHT Branch

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

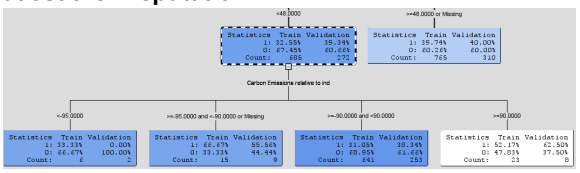


Adjusting the threshold to 530, the accuracy was consistent at 67% for train and test sets, supported by 14 companies:

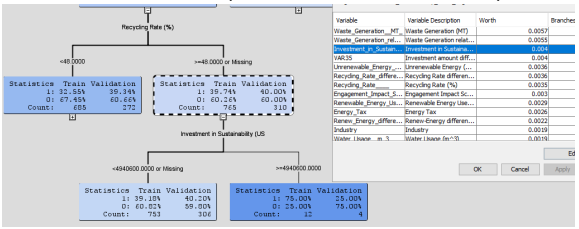


Found this strong split in water usage. However, this occurrence might just be due to chance since the validation set is suggesting the opposite claim.

boost their reputation:



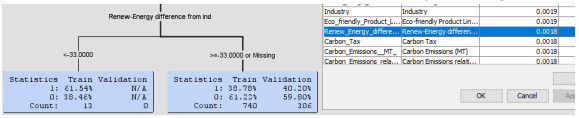
The decision of using Investment in Sustainability as the feature for the next split after >=48% is driven by the business domain (rather than feature worth):



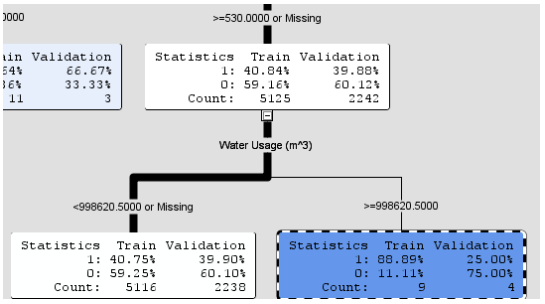
My **assumption is that companies that have such a high recycling rate likely care about sustainability for them to be taking these recycling measures**. Hence, I want to assess if they spend considerable money on sustainability.

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.

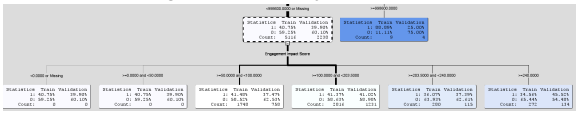
I pruned the split value to a neat threshold of 33 MWh which also produced a decent 62% purity in target=1:



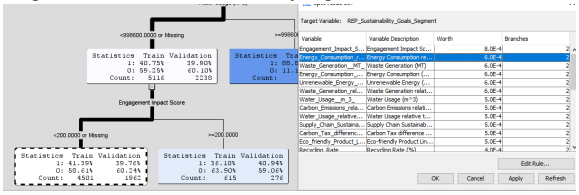
Interpretation: Businesses **can be 62% confident that they have a high likelihood** of achieving sustainability goals if they follow the series of decisions that leads to this node **BUT** this claim is not validated since no company was tested.



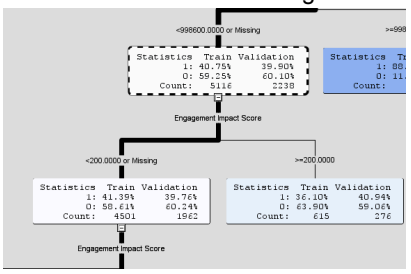
I manually splitted the ranges trying to find interesting subsets. Nothing found this layer, so I will split at 200:



The top splitting feature is the same, so I tried the second one. After experimenting with different features and split ranges, I was unable to find any high likelihood=1 leaf.



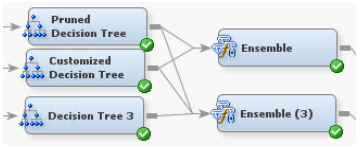
Since there is no high likelihood to be found, I will prune this node to remove the Engagement Impact Score branch as it is meaningless:



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.

Used maximum probability as criteria to let Ensemble choose best decision it is most confident with (highest purity of either class in the leaf) out of all trees.



Model Evaluation

Based on article 1 and article 2 from neptune.ai here are metrics applicable for assessing Classification models:

1. **Precision** - Measures accuracy of positive predictions. Formula = **TP/(TP+FP)**.
2. **Sensitivity** - Measures model's ability to capture positive instances. Formula = **TP/(TP+FN)**.
3. **F1 Score** - Harmonic mean of precision and recall to provide balanced measure that accounts for positive and false negatives. Formula = **2 * (Precision * Recall) / (Precision + Recall).**
4. **Specificity** - Measures accuracy of negative predictions. Formula = **TN/(TN+FP)**.
5. **Accuracy** - Measures overall accuracy of predictions. Formula = **(TP+TN) / (TP+TN+FP+FN)**.
6. **Misclassification rate -** Proportion of total misclassified/wrong instances. Calculated using **(FP + FN) / (TP+TN+FP+FN)**. Lower value indicates better accuracy in the model. Negate/Opposite of Accuracy.
7. **AUC-ROC** - evaluates the trade-off between sensitivity and specificity across various thresholds. Higher steepness indicates better ability to distinguish between positive and negative instances.
8. Matthews Correlation Coefficient (**MCC**) - Gauges model performance from -1 to 1, 1 signifying perfect predictions, 0 indicating random guessing, and -1 denoting complete disagreement with outcomes. MCC offers a balanced assessment by considering both True and False positives/negatives, making it suitable to evaluate models with class imbalance compared to accuracy or misclassification rate for imbalance models. Formula: **(TP * TN - FP * FN) / sqrt((TP + FP) * (TP + FN) * (TN + FP) * (TN + FN))**.

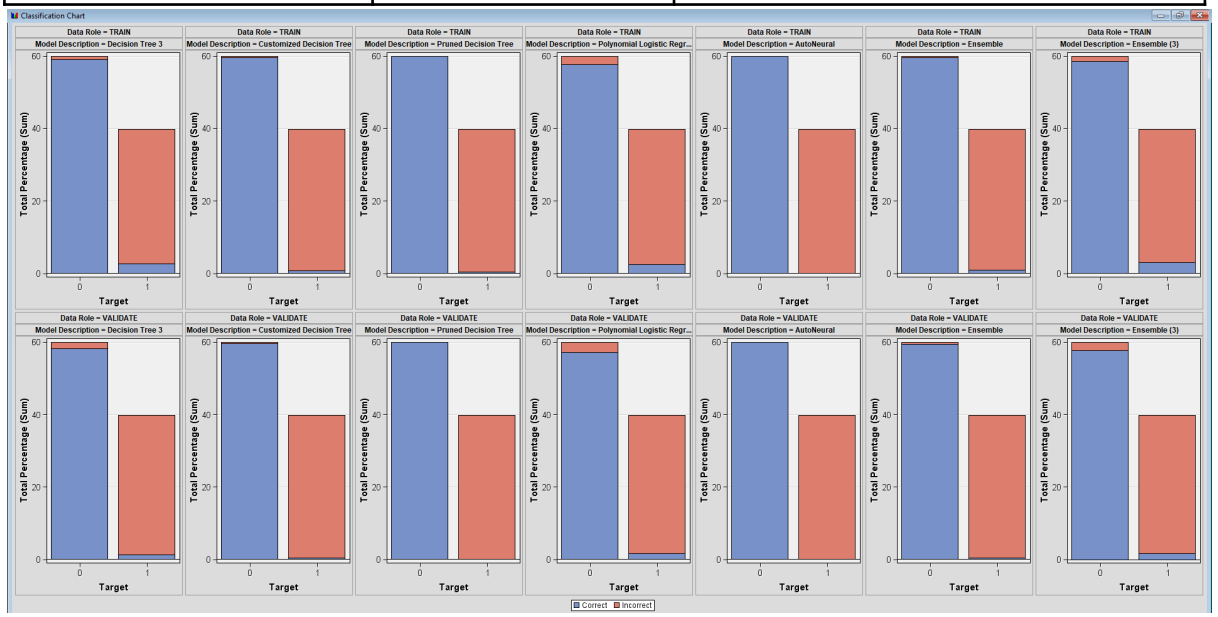
Since my **data has class imbalance** as there are more negative classes (negative class consists of Low and Medium likelihood, while positive only has High), Accuracy and Misclassification Rate isn't ideal because minority class (positive) may be misrepresented. Thus, I shall focus on Precision and Sensitivity as the model's purpose is to identify high likelihood companies, which are positive classes. Therefore, I do not need to evaluate specificity to interpret performance in predicting negative class. I will use Precision and Sensitivity instead of F1 or MCC as I want to quantify the accuracy of class 1 predictions and ability to capture class 1 separately. It's also **easier to explain and interpret** Precision and Sensitivity **for business users** compared to MCC value or AUC-ROC chart.

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

Metrics used to assess model performance

Precision of class 1: Ratio of correctly predicted positive class instances to quantify accuracy of positive class 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)



Rejected: AutoNeural as it's barely able to identify TP, making it useless. Polynomial as it is just a slightly 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: C-DT + Pruned DT
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): All 3 DT
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.

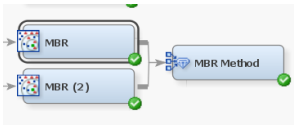
Pruned DT is the only model that performs decently for training and testing, but **a model that misses 99% of positive class** will not be helpful in identifying companies that have high likelihood. All Decision Trees suffer from significant overfitting above 10%.

Since all Decision Trees are unacceptable due to atrocious sensitivity rendering them useless…
I will proceed with the Clustering Model, KNN.

KNN (using MBR Node)

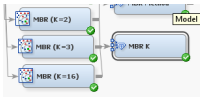
Tried both RD-Tree and Scan methods using k=2 to see which performs better. Both led to identical results. SAS documentation claims RD-Tree leads to better performance, but I proceeded with Scan since it uses distance to a probe observation which aligns with KNN while RD-Tree uses hierarchical clustering (less interpretable than DT). Since I'm using Scan, the other configurations of MBR are not applicable.

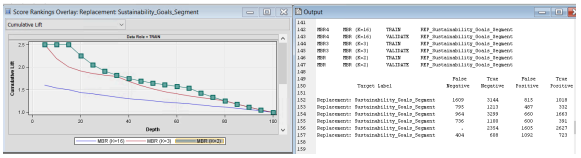
Target Label	False Negative	True Negative	False Positive	True Positive
Replacement: Sustainability_Goals_Segment	.	2354	1605	2627
Replacement: Sustainability_Goals_Segment	404	608	1092	723
Replacement: Sustainability_Goals_Segment	.	2354	1605	2627
Replacement: Sustainability_Goals_Segment	404	608	1092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



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

Now, I will compare using different k to find the optimal number of neighbours. 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).





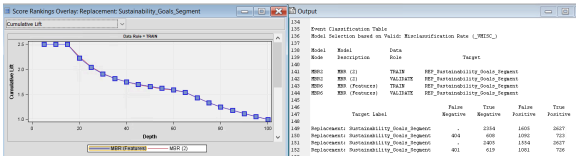
Features Selection

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%**.

Name	Label	Use	Report	Role	Level
REP_Sustainab	Replacement: Sustainability_Goals	Yes	No	Target	Nominal
Waste_Genera	Waste Generation (MT)	No	No	Input	Interval
Carbon_Tax_d	Carbon Tax difference from indus	No	No	Input	Interval
Record_ID	Record ID	No	No	ID	Interval
Carbon_Emissi	Carbon Emissions (MT)	No	No	Input	Interval
VAR35	Investment amount difference fro	No	No	Input	Interval
Water_Usage_	Water Usage (m^3)	No	No	Input	Interval
Renew_Energy	Renew-Energy difference from ind	No	No	Input	Interval
Recycling_Rate	Recycling Rate difference from i	No	No	Input	Interval
Community_En	Community Engagement Score	No	No	Input	Interval
Customer_Fee	Customer Feedback Score on Sus	No	No	Input	Interval
Eco_friendly_P	Eco-friendly Product Lines diffe	No	No	Input	Interval
Energy_Consur	Energy Consumption (MWh)	No	No	Input	Interval
dataobs	Observation Number	No	No	ID	Interval
Energy_Consur	Energy Consumption relative to i	Default	No	Input	Interval
Energy_Tax	Energy Tax	Default	No	Input	Interval
Investment_in_	Investment in Sustainability (US	Default	No	Input	Interval
Energy_Tax_di	Energy Tax difference from indus	Default	No	Input	Interval
Carbon_Tax	Carbon Tax	Default	No	Input	Interval
Engagement_I	Engagement Impact Score	Default	No	Input	Interval
Waste_Genera	Waste Generation relative to ind	Default	No	Input	Interval
Recycling_Rate	Recycling Rate (%)	Default	No	Input	Interval
Renewable_En	Renewable Energy Use (%)	Default	No	Input	Interval
Unrenewable_I	Unrenewable Energy (MWh)	Default	No	Input	Interval
Carbon_Emissi	Carbon Emissions relative to ind	Default	No	Input	Interval
Sustainable_Pa	Sustainable Packaging Initiative	Default	No	Input	Interval
Eco_friendly_P	Eco-friendly Product Lines (%)	Default	No	Input	Interval
Water_Usage_	Water Usage relative to industry	Default	No	Input	Interval
Supply_Chain_	Supply Chain Sustainability Scor	Default	No	Input	Interval

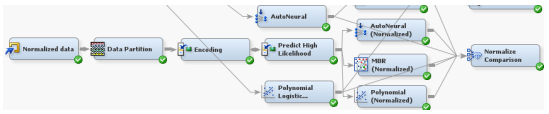


Normalize predictors for better performance

Based on this article by Machine Learning Models.org, distance-based clustering models and Neural Networks benefit from scaling features.

Steps to scale predictors:
Z-score normalization was chosen instead of Log transform as features compared with ind avg have negative bounds and Log function doesn't work on negative values. 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 same preparation nodes like data partitioning and encoding and connect the file reader into this pipeline.

```
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

Performance Evaluation:
NN improved drastically as it is now able to detect positive classes (but still worse than Ensemble (3) for training). However, performance for MBR worsened. Polynomial Regression also did not improve despite increased TP since FP also increased. Hence, **no normalized model is beneficial** for analysis.

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.

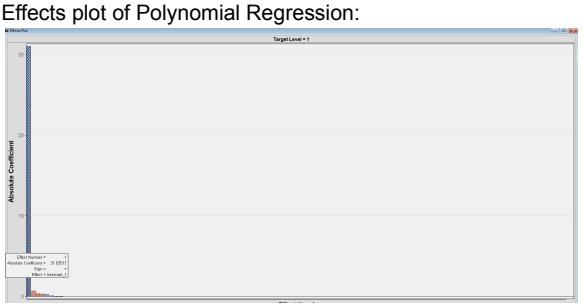
Inspecting significant interactions by hovering on the bars most of these interactions consist of Carbon Tax or Carbon Emissions relative to industry average.

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:

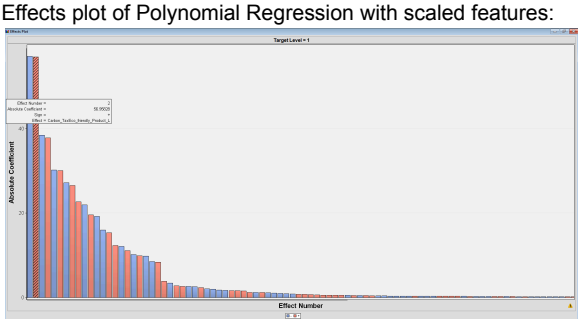
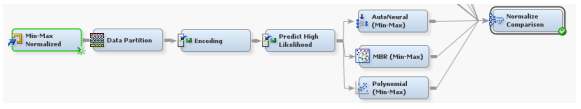


Table: Effects Plot sorted by P-Value ascending:

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

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 for Polynomial as the range is normalized from 0 to 1.




```
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 ind: REP_Su...	0.615201	0.050495

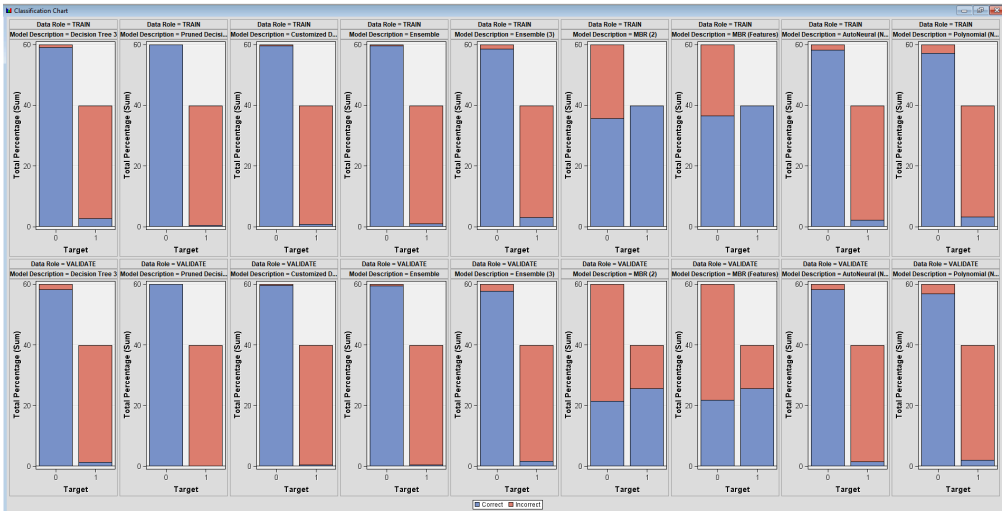
Analysis of Models

Model Node	Model Description	Data Role	Target Label	False Negative	True Negative	False Positive	True Positive
Tree5	Decision Tree 3	TRAIN	Replacement: Sustainability_Goals_Segment	2454	3893	66	173
Tree5	Decision Tree 3	VALIDATE	Replacement: Sustainability_Goals_Segment	1090	1646	54	37
Tree3	Pruned Decision Tree	TRAIN	Replacement: Sustainability_Goals_Segment	2603	3958	1	24
Tree3	Pruned Decision Tree	VALIDATE	Replacement: Sustainability_Goals_Segment	1123	1698	2	4
Tree4	Customized Decision Tree	TRAIN	Replacement: Sustainability_Goals_Segment	2573	3930	29	54
Tree4	Customized Decision Tree	VALIDATE	Replacement: Sustainability_Goals_Segment	1113	1686	14	14
Ensmbl	Ensemble	TRAIN	Replacement: Sustainability_Goals_Segment	2566	3929	30	61
Ensmbl	Ensemble	VALIDATE	Replacement: Sustainability_Goals_Segment	1113	1685	15	14
Ensmbl2	Ensemble (3)	TRAIN	Replacement: Sustainability_Goals_Segment	2428	3866	93	199
Ensmbl2	Ensemble (3)	VALIDATE	Replacement: Sustainability_Goals_Segment	1081	1636	64	46
MBR2	MBR (2)	TRAIN	Replacement: Sustainability_Goals_Segment	.	2354	1605	2627
MBR2	MBR (2)	VALIDATE	Replacement: Sustainability_Goals_Segment	404	608	1092	723
MBR6	MBR (Features)	TRAIN	Replacement: Sustainability_Goals_Segment	.	2405	1554	2627
MBR6	MBR (Features)	VALIDATE	Replacement: Sustainability_Goals_Segment	401	619	1081	726
AutoNeural2	AutoNeural (Normalized)	TRAIN	Replacement: Sustainability_Goals_Segment	2488	3835	124	139
AutoNeural2	AutoNeural (Normalized)	VALIDATE	Replacement: Sustainability_Goals_Segment	1084	1647	53	43
Reg	Polynomial (Normalized)	TRAIN	Replacement: Sustainability_Goals_Segment	2419	3767	192	208
Reg	Polynomial (Normalized)	VALIDATE	Replacement: Sustainability_Goals_Segment	1069	1607	93	58

All models fail in precision (0.5 or below) for testing except for Pruned DT because there are more FP than TP. This means the accuracy in class 1 predictions is not validated as the testing performance does not match training, which could suggest that the models are unreliable. 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 (Features)

Train Precision = 0.63, Train Sensitivity = 1, Test Precision = 0.40, Test Sensitivity = 0.64



The classification chart above shows that MBR is the only model where more than 50% of class 1 were correctly predicted. However, the performance between training and validation sets are inconsistent, indicating overfitting. This only became a concerning issue during validation because over 50% of class 0 predictions are incorrect, revealing poor precision as most positive predictions are wrong. This suggests poor reliability in MBR.

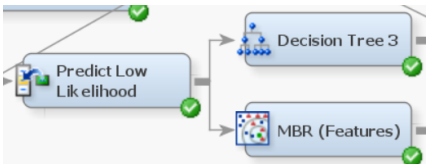
For Pruned DT, businesses can be 67% confident (0.67 test precision) about a positive prediction being true, supported by a validation set. However, 99% of high likelihood conditions are not detected by this model.
For MBR, businesses can be 63% confident about a positive prediction being true and at least 64% of high likelihood conditions are detectable. But, these patterns may not always be true and applicable for companies because the validation performance is poor. Both models have tradeoffs, but since I aim to assess the majority…

MBR is the preferred model as a DT that identifies less than 1% of positive class is useless since the majority of companies will be labeled as 0 anyways, and when 99% of high likelihoods are not found, these companies are better off guessing whether they fall in high likelihood than to use the model. However, MBR provides broader coverage by identifying more positive classes. Despite misclassifying some negative instances as positive, the pattern that leads to high likelihood should generally be reflective for most companies, having some exceptions occasionally. Moreover, the commitment to sustainability goals is considered valuable, as even attempts to reduce carbon emissions, albeit not achieving the goal, demonstrate a proactive approach. Proceeding with MBR involves **assuming the scenario being predicted resonates with training data** (e.g. same timeframe, country, goals, etc.), as MBR is unable to generalize with new and unseen trends; **validation performance is terrible**.

Modelling (TARGET=Low Sustainability Goals Segment)

Predict at risk of NOT achieving 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.

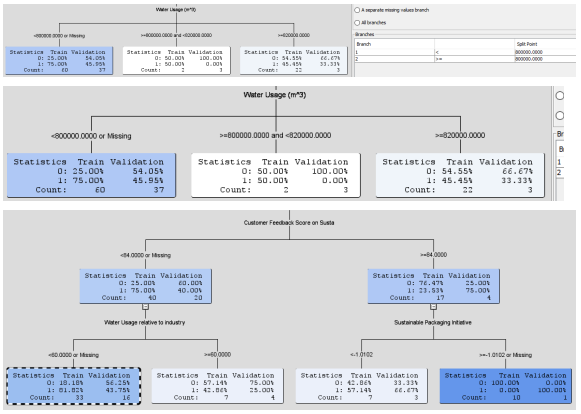


DT

Manually pruned by deleting branches that do not identify low likelihood (class=1).

Experimented with different split values to find optimal values that look appealing while producing high purity of class=1.

After pruning, the key highlight was that validation purity of class=1 is not consistent with training, making reliability and validity of claims from these decisions questionable. I was unable to dig much insights from the data and this is because there weren't any interesting patterns within the data to uncover in the first place.

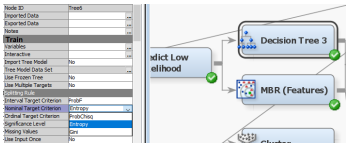


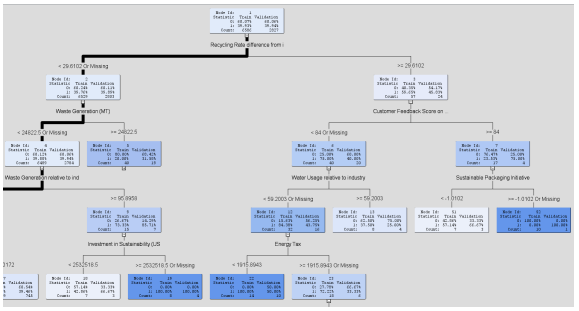
[Finding ideal splits from trial and error]

Interpreting DT (entropy)

I changed splitting criterion to entropy and regrew my tree trying to find insightful splits. ProbChiSq splitting wasn't considered since it was found earlier that all predictors had weak chi-sq which results in only a very small tree being built. **Insights from this Decision Tree seem rather bizarre**.

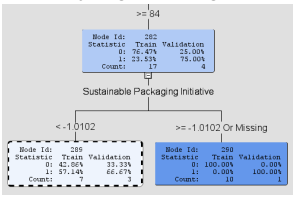
For instance, if a company's feedback score is below 84, even if the company has lower water usage and lesser energy tax, they still have a high risk of not achieving sustainability goals. This could imply that having a customer feedback score of 84 or above is crucial in ensuring a company is NOT AT RISK of being unable to achieve goals, and could be a key factor to identifying low likelihood.

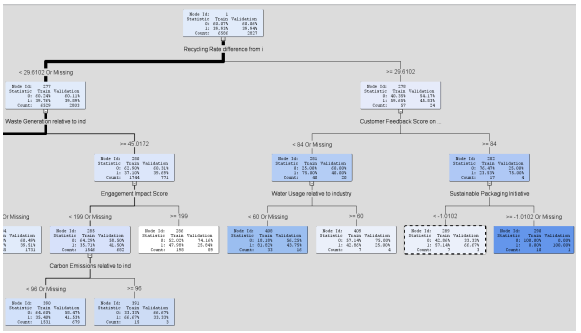




Interpreting DT (gini)

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.
The only logical insight (same insight in entropy):





[Photo of Decision Tree]

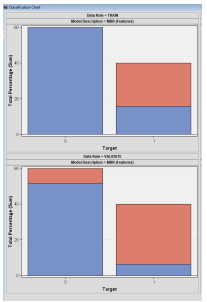
Can performance be improved?

Ensemble: As DT (Gini) is meaningless as the only insight available is already present in DT (entropy), I will not ensemble. Impractical to build Random Forest with 100 DTs in SAS EM Miner to dig for insights.
Scaling Features: makes values hard to interpret and Decision Trees don't benefit from scaling.

MBR

Train: Precision=1, Sensitivity=0.40, **Test**: Precision=0.43, Sensitivity=0.16
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. Model's precision and sensitivity are poor, suggesting **model is bad**.

Similar to the champion model for predicting high likelihood, MBR, testing precision is overfitted to the extent it is below 50% positive predictions are likely to be correct. Hence, <u>if MBR is chosen for this problem of predicting low likelihood companies, I have to disregard its training performance</u> which could make reliability and validity in claims questioned by businesses. Unlike previous MBR, there is no perks of taking this risk as training sensitivity is also below 0.5.



Model Selected for identifying low likelihood (high risk): DT (entropy)

Model Node	Model Description	Data Role	False Negative	True Negative	False Positive	True Positive
MBR7	MBR (Features)	TRAIN	1582	3956	.	1048
MBR7	MBR (Features)	VALIDATE	948	1457	241	181
Tree6	Decision Tree 3	TRAIN	2516	3915	41	114
Tree6	Decision Tree 3	VALIDATE	1098	1659	39	31
Tree7	Decision Tree (entropy)	TRAIN	2573	3947	9	57
Tree7	Decision Tree (entropy)	VALIDATE	1113	1686	12	16

Reason:

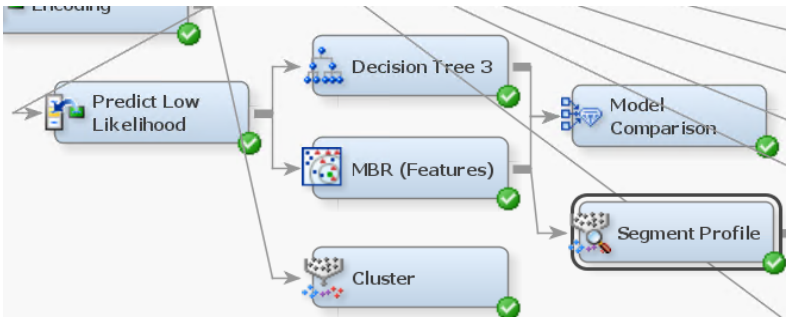
DT (entropy) is the **only model with a passable (but still poor) testing precision** of 0.57. Although MBR detects significantly more class 1 than DT (e), the precision is below 50%, making it more likely to generate false alarms by signaling a risk when there is none. The red flag could lead to unnecessary hysteria and potential damages when companies promptly take action trying to resolve a problem that doesn't exist.

The chosen Decision Tree aims for a balance, although unable to identify most red flags, but when it does there is a reasonable level of confidence that it isn't a false alarm.

Interpretation and Recommendation

I will now interpret how these models derive their predictions to understand how various factors influence likelihood of achieving sustainability.

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

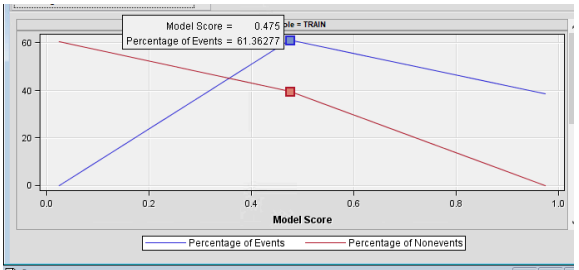
Variable	Worth	Rank
Unrenewable_Energy__MWh_	.002053926	1
Energy_Consumption_relative_to_i	.001776568	2
Eco_friendly_Product_Lines____	.001763858	3
Eco_friendly_Product_Lines_diffe	.001752369	4
Carbon_Emissions_relative_to_ind	.001727656	5
Energy_Tax_difference_from_indus	.001702645	6
Carbon_Tax_difference_from_indus	.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.



Output

```
Data Role=TRAIN Target=REP_Sustainability_Goals_Segment Target Label=Replacement: Sustain

   False      True      False      True
Negative   Negative   Positive   Positive

     .        2392       1567       2627


Data Role=VALIDATE Target=REP_Sustainability_Goals_Segment

   False      True      False      True
Negative   Negative   Positive   Positive

    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

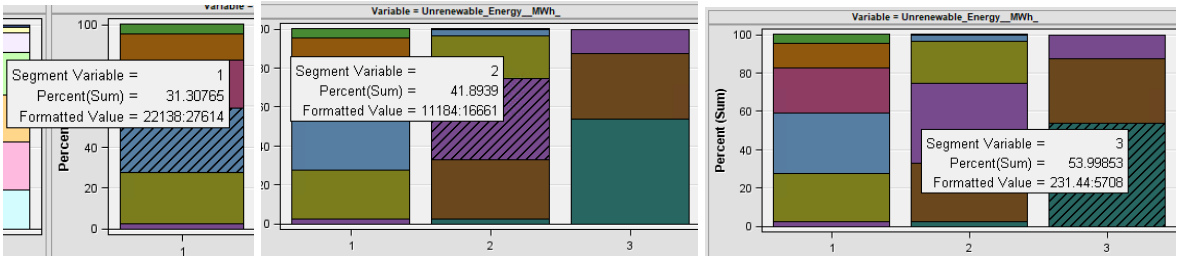
Variable	Worth	Rank
Waste_Generation_relative_to_ind	.002219084	1
_N2	.002022449	2
Carbon_Tax_difference_from_indus	.001892046	3
Water_Usage_relative_to_industry	.001806451	4
Renew_Energy_difference_from_ind	.001729958	5
Eco_friendly_Product_Lines_diffe	.001716946	6
Renewable_Energy_Use____	.001700829	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 with the goal of learning patterns in 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. The variables used are the same as MBR (Features). This allows me to understand exactly what conditions cause a high or low likelihood. I clustered using K=3, with the intention of splitting 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:

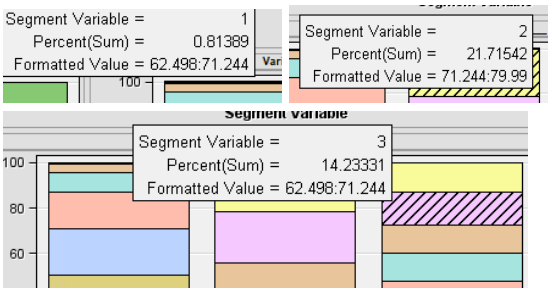


- **Low likelihood companies use at least 11,184 MWh of Unrenewable energy**.
- A company is at a **very high risk of falling into low likelihood if they exceed 22,138 MWh** because no noticeable 22138 (teal) instances in cluster 3, and **no noticeable instances in cluster 2 and 3 that exceed 22,138 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.

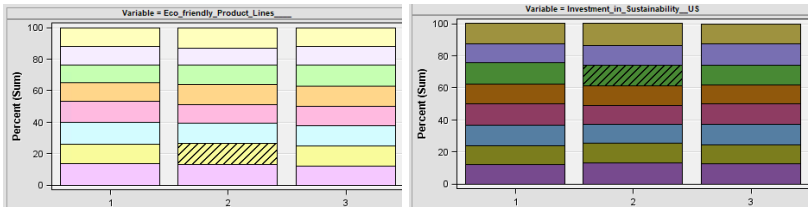
Renewable Energy (%) Impact:

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

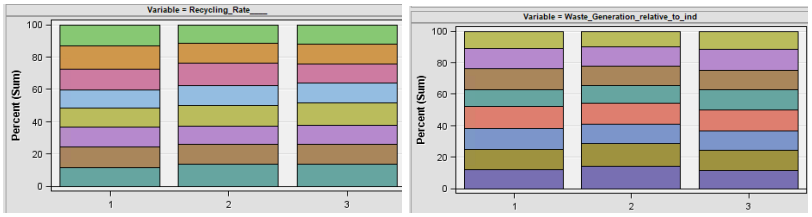
To address the need of heavy electricity usage required for operations, companies should try to have at least 62.5% of their energy consumption using renewable energy to avoid falling into the risk cluster that is not likely to achieve sustainability goals in the next 5 years.



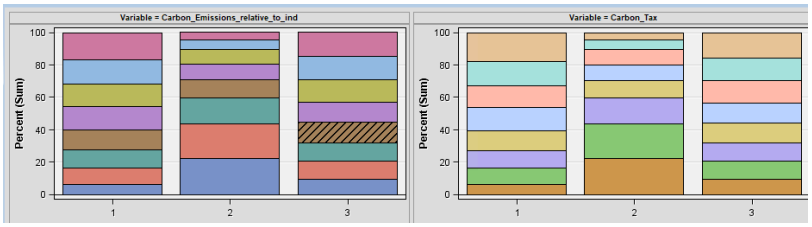
Eco-friendly product lines (%) and Investment in Sustainability:



Waste Management and recycling efforts:



Carbon Footprint:

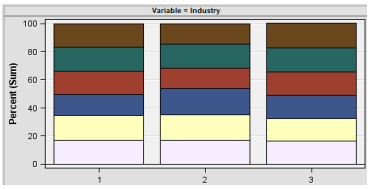


No pattern between Low and High clusters as proportion of each colour are closely similar, suggesting these factors are not impactful enough to distinguish low and high likelihood companies.

Conclusion:

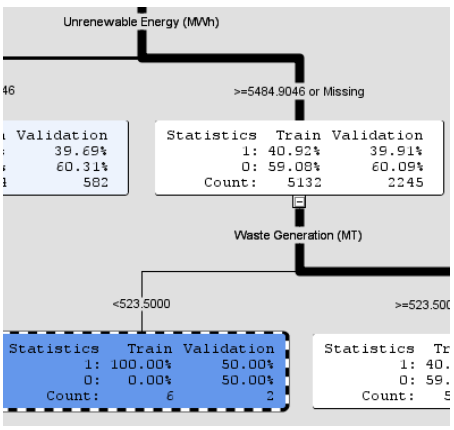
Energy Consumption seems to be the main factor that determines the likelihood of achieving low or high likelihood based on the Clustering model.

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

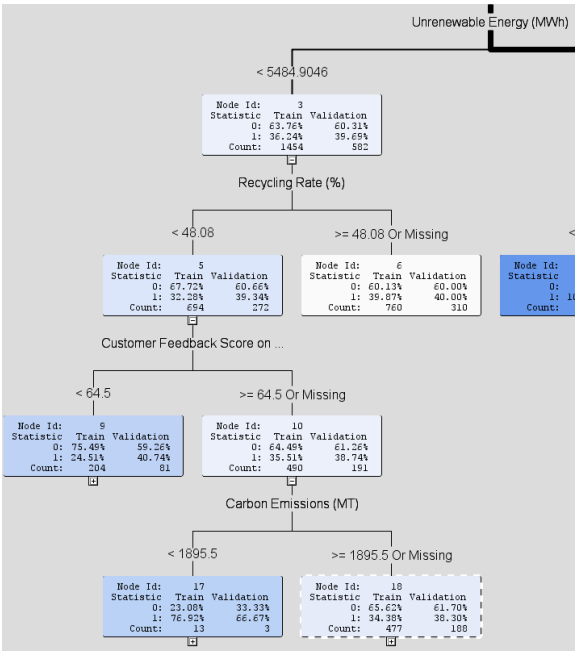
I will use strong paths in Decision Trees to justify the recommended plan of actions since I can use the train and test purity to show confidence in claims.

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.

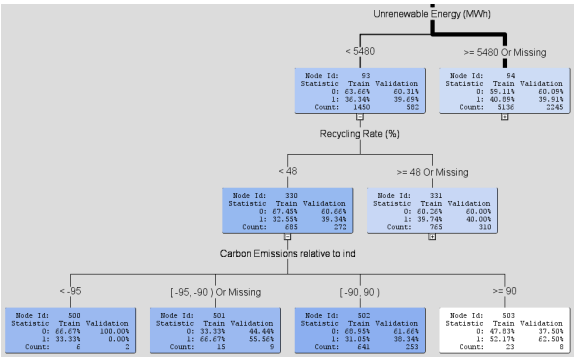
Hence, companies can try replicating these conditions to have high confidence that they can achieve their sustainability goals in the next 5 years by reducing reliance on unrenewable energy and emitting carbon, and take proactive steps to raise customer feedback scores.



[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.

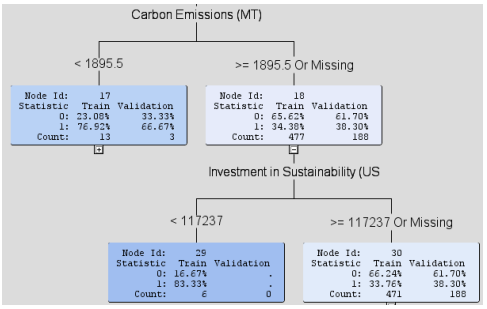
I **assume that high carbon emitters are large companies** since they generate a lot of carbon from their heavy operations, but the fact that they still fall in the minority group of low unrenewable energy users suggest that <u>as long as a company takes initiative to reduce non renewable energy usage,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

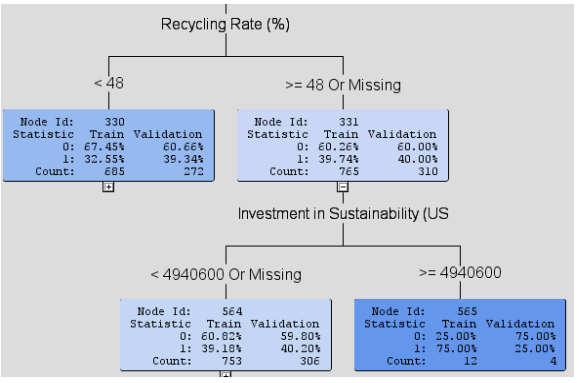


[Custom DT, Precision Train: 65%, Test: 50%]
Another interesting insight is that high carbon emitters (90% above industry average) are more likely to achieve high likelihood than mid-range (-90 to 89) carbon emitters.

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%]

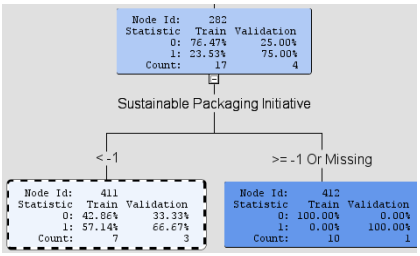
Hence, **companies should focus on taking measurable actions** like switching to renewable alternatives to reach their sustainability goals, instead of solely pouring money to invest in sustainability.

Companies with a customer feedback score below 84, despite lower water usage and reduced energy tax (lesser unrenewable energy usage)**, are still at risk of not achieving sustainability goals.**

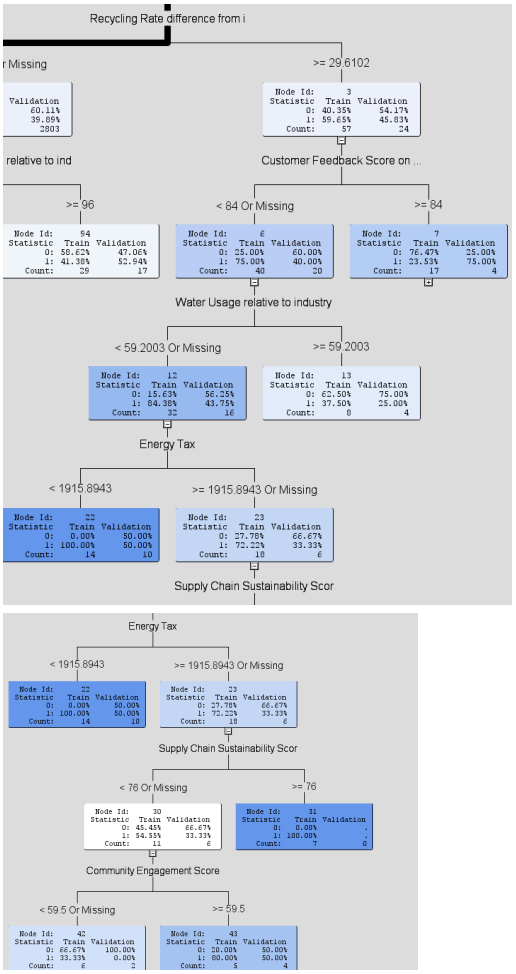
The critical threshold appears to be a feedback score of 84 and maintaining this level is crucial to mitigating the risk of falling short of sustainability goals.

This insight suggests the significance of customers' perceptions, as the score they give, as an indicator of whether a company is likely to achieve 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.

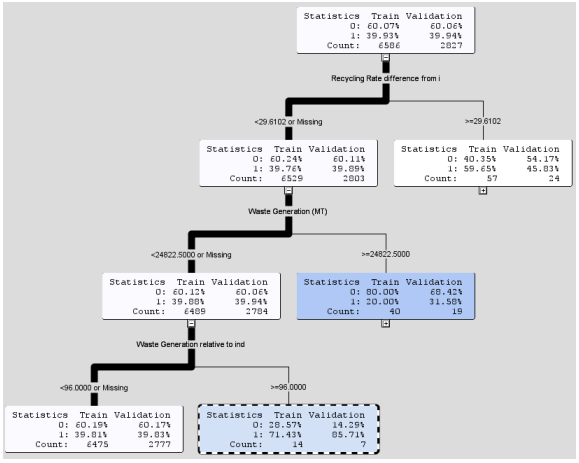


[DT predicting low likelihood as class=1]

Companies that fall within the top 5% of waste generation for their respective industry, yet produce less than 24,822.5 metric tons of waste, and exhibit recycling rates lower than 29.6% compared to the industry average are 70-80% at risk of not achieving sustainability goals in the next 5 years.

To mitigate this risk, companies should reduce waste generation by:

- Increasing recycling rates by placing my recycle bins in the workplace.
- Cut down on unnecessary waste.
- Reuse and upcycle whenever possible.



[DT predicting low likelihood + pruned split value]

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 when applicable. Opt for environmentally friendly appliances and products. Use only what's needed, upcycle waste if possible. Include **recycling** and composition bins to encourage recycling at the workplace.
3. Don't just invest in sustainability using money. **ACT on the goals** by doing step 1 and 2 above.
4. Take **customers' feedback as a gauge** on whether company is on the right track towards sustainability.
5. Implement sustainable packaging for products whenever possible to **increase sustainability efforts to raise customers' impression** on the company - thereby leading to increased customer feedback score on sustainability when customers see the company taking actions. Aim to reach a score of 84 or better.

Deployment plans:

Decision Tree (entropy) to predict low likelihood can be deployed as it has a passable precision of 0.57 for testing, suggesting fair ability to predict high risk companies with 57% accuracy. This model has low sensitivity of 0.1-0.2, suggesting that only 1-2% of high risk companies are identified, but is useful because if a positive class is found, the business knows that they require immediate action. They can consider the recommendations above, proceeding with the assumption that the general pattern the model runs on happens to be relevant and applicable. If a company wants to know if they have high likelihood, they can use Pruned DT and there is a 67% for a positive prediction to be correct. However, companies should not be disheartened by a negative prediction from this model, since 99% of high likelihood companies are not detectable by model.

Limitations of analysis

1. **Data Quality:**
   Accuracy of analysis depends on the data. 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 insights. I had to proceed with the assumption that those few companies reflect the actual trend if data size were to scale up. 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. **Not every ESG Metric assessed:**
   Due to lack of sufficient details in the dataset, such as green practices within a company's operations. Analysis is limited to basic level information like carbon emissions, recycling rate, etc, hence unable to uncover exact causes.
4. **External factors:**
   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 are not factored by models.
5. **Time/Location Sensitivity:**
   Practices and goals evolve over time, causing patterns in Low and High likelihood companies to change, and may vary across regions. The models did not account for such factors. Hence, the models may not be usable and accurate for all companies.
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. **Insights are not validated or backed up:**
   Models performed with poor accuracy when tested with a validation set. Hence, the patterns may vary in real-world scenarios due to poor generalizability of models, and will not always necessarily match with companies being tested.
8. **Recommendation only applicable for small subset of companies:**
   Only few companies fall under the specific conditions as the Decision Tree models. This means the majority of companies that do not have scenarios resembling the models' would find the recommendations unrelated hence inapplicable to their business.

Analysis could be improved if:

1. **More information about companies in list** (such as country, year of observation) as sustainability practices and goals vary across time and regions. This could explain why MBR had poor generalizability.
2. **All necessary information to fully understand the Target is provided** (such as how it was measured or calculated). If more details regarding Target were provided, such as the methodology for deriving the value, it could be beneficial in improving this analysis as I will have a clearer understanding of what to focus on during features engineering, specially constructing features catered to predicting target.
3. There is a **diverse range of features and information** about the company's operations to explore and delve into as the features provided in the dataset may be relevant to the problem statement but may not be related or influence the Target. I am also unable to integrate much external data into this analysis as there is insufficient information provided for me to establish a connection and link to other sources.
4. The **"Sustainability Goals" that the companies set are provided**. Models were built with the assumption that these goals set are generally related to the ESG pillars of sustainability. If companies provided what their sustainability goals are, this analysis would be more complete as I will have better understanding and more information to work with to tailor analysis towards companies to resolve their unique sustainability issues and challenges for their specific context.
5. **Origin of the dataset** provided(how observations were recorded). I dropped all Healthcare and Finance records as I found an issue in a column for these rows to **ensure veracity in data** as I suspected the whole row was inaccurate. Given confirmation that those errors only lie within that column, or if those values represented something true, I would have imputed NaN or 0 under those columns instead.
6. **More data** for 60-20-20 partition while ensuring sufficient data in each set, for extra layer of testing to be certain a model is well-generalized and able to perform consistently when deployed.

Models can be improved if (year and country) or sustainability goals of the company were provided, as I can integrate features in this recent dataset about SDGs (highly relevant to sustainability) into the modelling dataset. Given the opportunity, I would reach out to companies to collect more data about their operations and practices or collaborate with sustainability organizations to obtain relevant data with features to supplement this analysis.

Considerations

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

Scope of Analysis: The data is lacking sufficient patterns as the true key drivers that influence Target are not in the dataset, hence the patterns learnt by models are weak and unable to hit acceptable benchmarks (precision and sensitivity >= 0.7) by industry.

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).

Summary

Several supervised machine learning models were built to solve 2 classification problems:

1. Whether a company would have high likelihood (>60) to achieve its sustainability goals in the next five years
2. Whether a company would fall at risk of not being likely to achieve its sustainability goals in the next five years (low likelihood < 40).

Prospective Decision Tree and MBR models were then interpreted and analyzed, to understand what steps the model takes to derive its predictions. The analysis focused on finding explanations to justify how reducing non-renewable energy usage, and other environmentally sustainable practices could help a company improve their odds of achieving their sustainability goals.

Generally, the models created were significantly overfitted by around 20% as the precision in predicting positive instances were inconsistent between training and testing despite attempts to resolve overfitting issues. This suggests that either the true key drivers with significant patterns that affect Target are not in the dataset, or the Target is unpredictable and not in the scope of a ML problem, or the missing information of time and region of the companies distorted the patterns between features and sustainability goals hence poor and inconsistent performance. Every model faces lack of generalizability and either extremely poor sensitivity (unable to detect positive class) or poor precision in training (50-60% of positive class predictions are wrong).

Despite poor model performances, there were still meaningful insights within the data that were successfully mined and constructed into actionable recommendations. The credibility and validity of these recommendations should be assessed using the Decision Tree's individual branch that suggests these recommendations. The validity can be assessed by judging the training and testing purity to gauge how reliable these recommendations are.

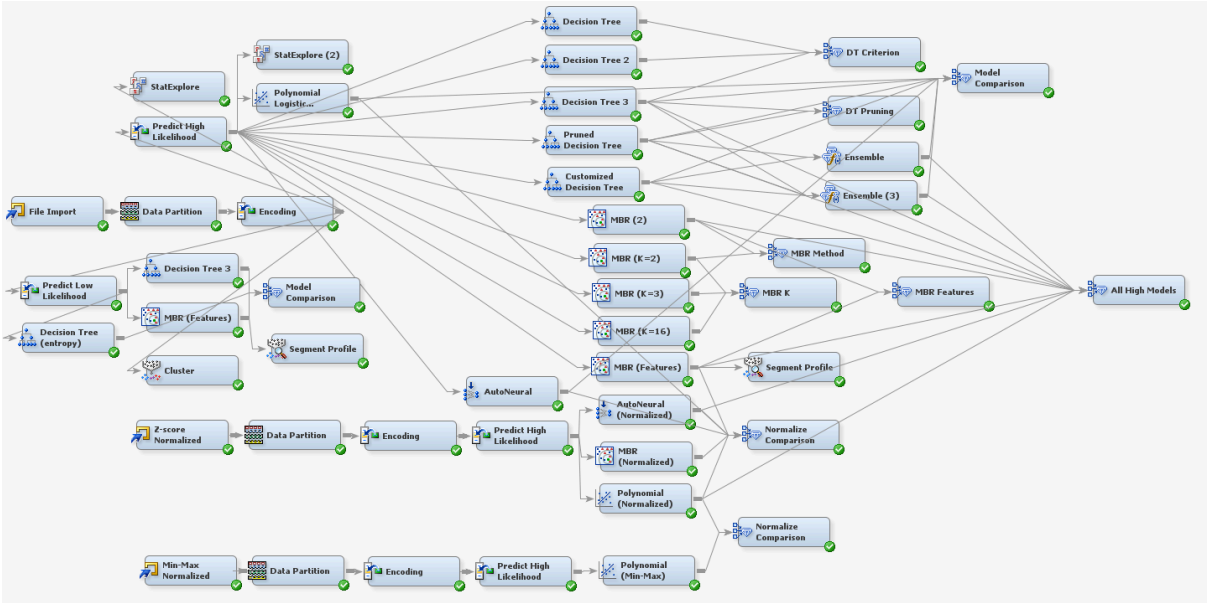
Conclusion

The pursuit of sustainability goals requires a commitment from companies to make sustainable practices. But even after following common practices, certainty for a company to achieve their sustainability goals based on operational characteristics proves to be elusive. For companies seeking to improve their sustainability practices to align with their sustainability goals, the recommendations from this analysis could be beneficial if applicable, but there is no guarantee of foolproof success given the modest sample size tested on these claims.

Predicting sustainability attainability through Machine Learning encounters complexities due to the nuanced nature of a company's goals. The intricacies of each organization's journey demand a thoughtful, context-specific comprehension that transcends the abilities of machine learning algorithms. In navigating the intricate landscape towards sustainability, a holistic approach becomes imperative. Besides following traditional sustainable practices like the 3Rs and reducing carbon emissions, water usage, and waste generation, companies should employ a comprehensive strategy grounded in logic that aligns their sustainability efforts to address their own unique goals instead of conforming to a conventional approach.

In a company's pursuit of achieving sustainability goals, machine learning serves as a supporting reference, not the primary guide. Predictive models can facilitate informed decision-making, but companies should ultimately rely on understanding their specific goals to determine the logical steps to take to achieve it.

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]

External analysis to supplement data engineering: https://ourworldindata.org/emissions-by-sector

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]

Various evaluation metrics for classification: https://iq.opengenus.org/precision-recall-sensitivity-specificity/

Deeper research into more types of evaluation metrics to determine relevant metrics to use:

https://neptune.ai/blog/how-to-deal-with-imbalanced-classification-and-regression-data

https://neptune.ai/blog/ml-model-evaluation-and-selection

Understanding MCC:
https://towardsdatascience.com/matthews-correlation-coefficient-when-to-use-it-and-when-to-avoid-it-310b3c923f7e

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

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 [Understanding configurations]

Research on using MBR: https://documentation.sas.com/doc/en/emref/15.3/n1fevkin0iu4cxn1khv8ow9r8pmj.htm [Learn how the configurations work]

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

External data sources which could be integrated to analysis given more info in given dataset:
https://www.kaggle.com/datasets/sazidthe1/sustainable-development-report?resource=download&select=sustainable_development_report_2023.csv [Dataset about likelihood of achieving SDGs 1-6 for countries from 2020-2023]

About SDGs: https://sdgs.un.org/goals [Sustainable Development Goals by United Nations]