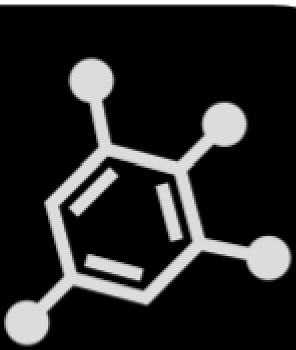
에코 프로젝트 AI 화학



February 2025 - 알렉산드로 Magana ~

Project Description

“Develop an AI model to predict waste volumes across South Korean cities. The model utilizes **clustering** to classify cities based on waste generation **patterns**, enabling more accurate **forecasting** of future waste production."

Project Assignment

UDEM ~ SAMSUNG

Authors and Researchers

Alejandro Garcia Magana

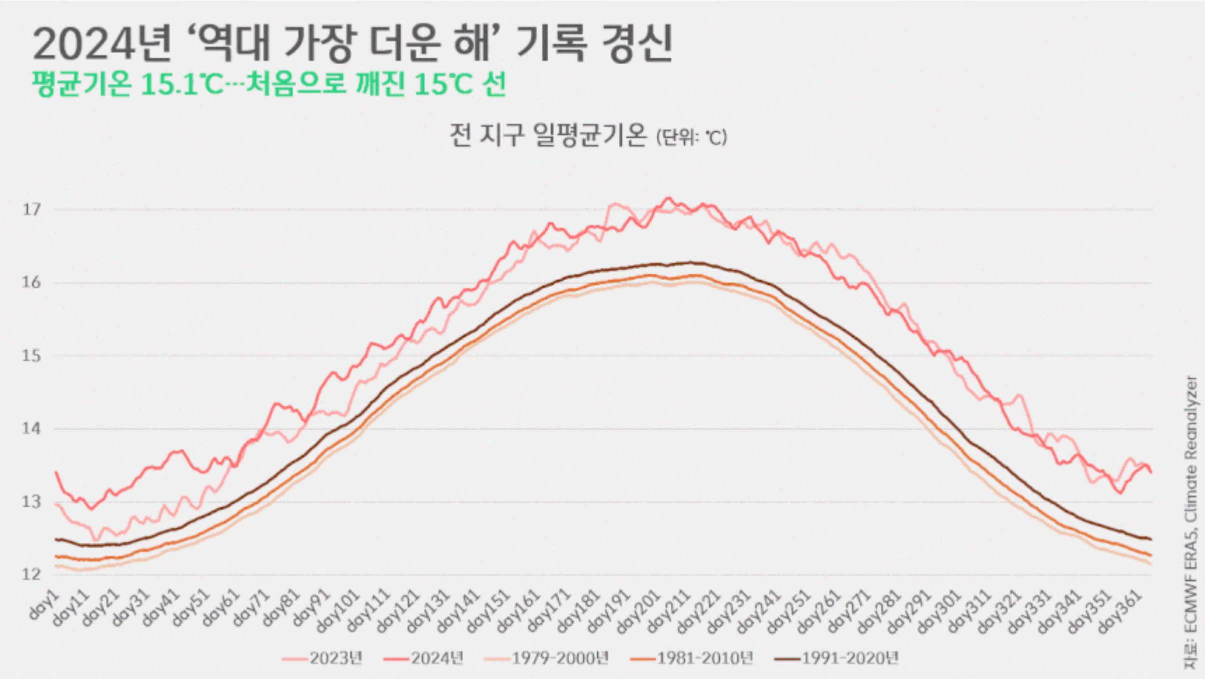
Involved University

CETI ~ Centro De Ensenanza Tecnica Industrial

Index

Introduction - 2024’s Climate Milestone: A New Global Temperature Era

2024 has been recorded as the hottest year on record. For the second consecutive year, the global average temperature reached a new high. In 2024, the global average temperature was 15.09℃, surpassing the 15℃ threshold for the first time and breaking the previous record of 14.98℃ in 2023. The annual average temperature record was also shattered, rising from 17.08℃ on July 6, 2023, to 17.16℃ on July 22, 2024. Even at the peak of summer, the average daily temperature across the entire planet—from the North Pole to the South Pole—has never been this high since temperature measurements began.



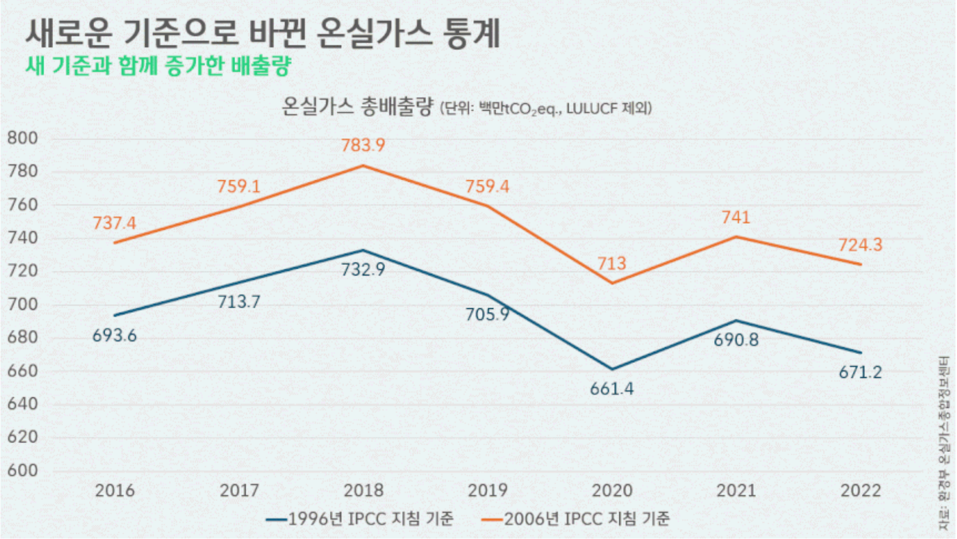
자료: ECMWF ERA5, Climate Reanalyzer

It wasn’t just the atmosphere that was heating up. The global sea surface temperature also reached an all-time high, breaking previous records.

Meanwhile, greenhouse gas emissions continue to pose a critical challenge. The Greenhouse Gas Information Center, under the Ministry of Environment, revealed a significant omission in national emissions data. During the verification process, it was discovered that some coal consumption figures in the Energy Statistics, the fundamental dataset for energy sector emissions, had been excluded. Since 2016, coal consumption from private coal power plants—amounting to 8.892 million tons—was left out of official records. When converted into greenhouse gas emissions, this omission accounts for up to 19.6 million tons of unreported emissions.

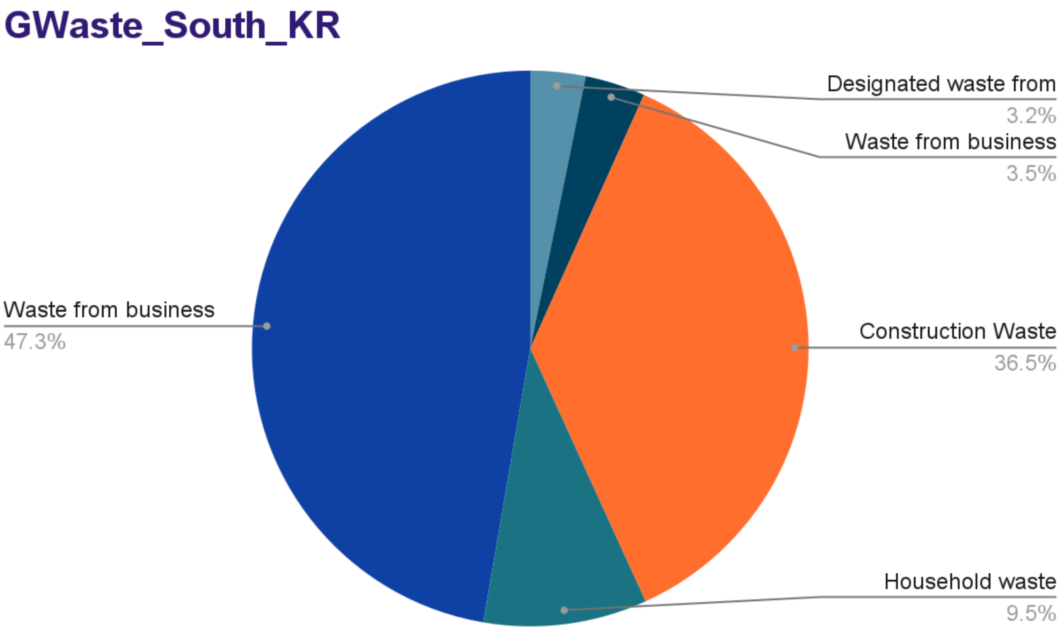
While it is commendable that authorities acknowledged this error instead of simply attributing changes to new calculation standards, it is concerning that such a critical oversight occurred in government-managed statistics used by international organizations. With the revised data, the previous peak emission record of 725 million tons in 2018 has now been corrected to 783.9 million tons. Despite recent reductions, annual emissions have yet to fall below 700 million tons.

Additionally, the reclassification of emission sectors has highlighted trends in different industries. Previously categorized into six sectors—(1) energy conversion, (2) industry, (3) transportation, (4) buildings, (5) agriculture, forestry, and fisheries, and (6) waste—the new classification groups emissions into four major categories: (1) energy, (2) industrial processes and product use, (3) agriculture, and (4) waste. Notably, emissions from industrial processes and product use increased to 131.3 million tons in 2022, surpassing the 128.9 million tons recorded in previous estimates.



자료: 환경부 온실가스종합정보센터

Distribution of waste generated in South Korea in 2023 (Type)

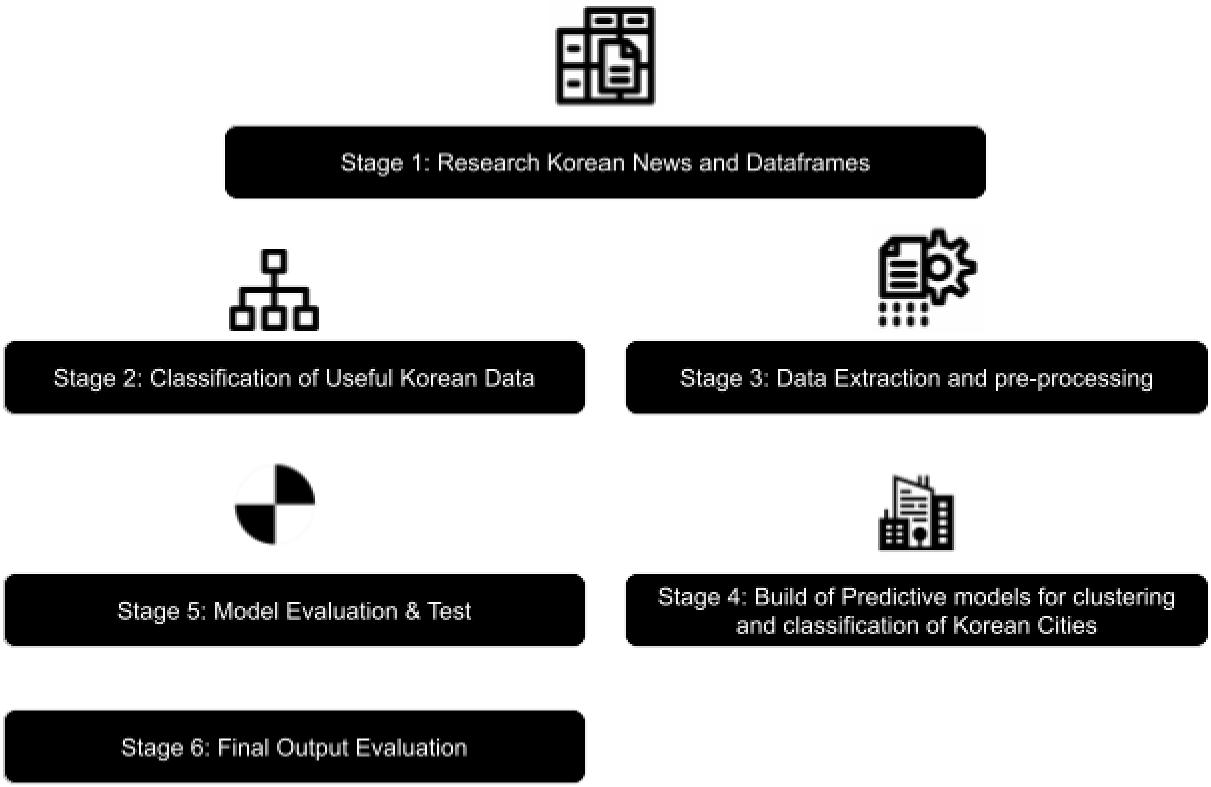


Objective of This Study

Climate change is not only about rising atmospheric temperatures; it is also influenced by industrial activities and **emissions** from cities and corporations. South Korea, like many other nations, faces increasing **environmental** challenges due to these emissions, which affect both neighboring countries and the global climate.

This study aims to **analyze** environmental emission patterns in Korean cities using data provided by the **Korea Environment Corporation**. By applying AI techniques such as K-Means clustering, the study will identify trends over time, classify cities based on their emission characteristics, and provide insights for policymakers and environmental planners. Understanding these **patterns** is crucial for developing more effective strategies for emission reduction and sustainable urban development.

Methodology - Project Workflow


Stage 1: Research Korean News and Dataframes
Stage 2: Classification of Useful Korean Data
Stage 3: Data Extraction and pre-processing
Stage 5: Model Evaluation & Test
Stage 4: Build of Predictive models for clustering and classification of Korean Cities
Stage 6: Final Output Evaluation

Stage 1: Research Korean News and Dataframes

At first glance, the main focus was on South Korea, so finding relevant news, articles, and datasets required searching through Korean websites, not just foreign sources. During this process, Alejandro and Aram led the efforts, one overseeing approvals and guiding the research, while the other accelerated the process in her native language, ensuring efficiency. They worked closely to verify the accuracy of the information, discarding irrelevant data and keeping only what was useful.

It is important to note that this stage took at least two days, as Alejandro needed additional time to find and translate relevant information. After the first day, a more effective strategy was developed to refine the search process and improve accuracy.

Without this structured approach, the research process could have been delayed and less precise. Finally, after reviewing multiple datasets, the most relevant one was selected for further work.

Stage 2: Classification of Useful Korean Data

During the process of gathering relevant datasets, an initial analysis was conducted through code to get a quick overview. This included describing the data, gathering statistics, handling null values, and examining each dataframe in detail, including:

- 한국환경공단_의료폐기물 소각시설별 연도별 소각량 – Korea Environment Corporation: Yearly incineration volume by medical waste incineration facility.

- 법인 – Corporations.

- 한국환경공단_사업장일반지정폐기물 배출 사업장 정보 – Korea Environment Corporation: Information on business establishments discharging general designated waste.

Additional datasets were also reviewed, but after grouping them, the key targets were identified.

Stage 3: Data Extraction and Pre-Processing

After classifying and selecting the datasets to be used, the next step was to explore each dataframe in greater detail. This involved examining all variables, applying filters to check if the data aligned with the planned training objectives, and analysing key statistics.

The process included calculating mean values, identifying maximum and minimum values, extracting the first and last records, and manually identifying potential patterns. Additionally, filtering was performed to ensure that only relevant data was retained for further processing.

Stage 4: Build of Predictive models for clustering and classification of Korean Cities

The initial step was challenging, as the variables did not clearly indicate a relationship. At first, a linear regression model was applied, but due to incorrect data and improper encoding of city names, the approach was flawed. The cities were converted into numerical values to apply the K-Means clustering method using annual waste measured in tons.

However, this approach proved ineffective, as encoding the cities led to illogical results. When the first clustering attempt was visualised, it failed to produce meaningful groupings, making it impossible to link the clusters back to real cities and regions.

To resolve this, the approach was adjusted by keeping the city names intact before and after clustering. Instead of modifying or encoding them, only numerical data—specifically the total waste per region—was used for clustering. This adjustment allowed the K-Means method to properly align and successfully classify cities along with their respective waste levels.

Following this improvement, the next step was to explore whether a Random Forest Classifier could effectively predict future waste levels for cities and determine if it was the most suitable method for this task.

Stage 5: Model Evaluation and test

The Random Forest Classifier proved to be a good choice after data processing, as the goal was to assign new incoming data to existing clusters, not just classify the current dataset.

To evaluate the model, tests were conducted by simulating "new cities"—i.e., new city data entries based on waste amounts corresponding to the predicted cluster classification.

After these tests, the model was also tested with existing cities to determine if it could correctly reassign them to their original clusters. Surprisingly, it achieved a perfect accuracy of 1.00, correctly classifying all cities.

Stage 6: Final Output Evaluation

To further validate the model's performance, additional tests were conducted using a **randomized** selection of cities. The goal was to track accuracy by comparing the original clusters with the predicted clusters to ensure consistency and reliability.

Outcome - Classification City Model

Regarding the final evaluations and results it's important to consider the dataset that were used.

Dataset : 한국환경공단_사업장일반지정폐기물 배출 사업장 정보 ~ Korea Environment Corporation_Information on business establishments discharging general designated waste

Registration Date: 2024-01-24
Date of revision: 2024-01-31

Whole rows: 16,330

Provider: Korea Environment Corporation

First Analysis

The dataset was entirely in Korean, which posed an initial but important challenge when working with Korean datasets. After reading the CSV (Comma-Separated Values) file, an expected error occurred:

"utf-8' codec can't decode byte 0xb1 in position 0: invalid start byte"

This happened because the dataset was encoded in CP949 (Expanded Complete), a character encoding developed to address issues with EUC-KR, which was commonly used in Unix-based systems. CP949 is also known as MS949 or WINDOWS949. When handling datasets from different countries, it is essential to consider encoding methods, as they vary between regions.

Preparing the Cluster

Here came one of the first challenges in the coding process—preparing the data for clustering. The dataset contained six columns:

기초시군구(관할관청) ~ Basic City / County / District (Jurisdiction): Lists all Korean cities.

업체명 ~ Company Name: Lists all company names.

연락처 ~ Contact Number: Contains company contact numbers.

폐기물구분 ~ Waste Classification: Specifies classifications related to waste generated in workplaces.

폐기물명 ~ Waste Name: Identifies the type of waste generated.

연간배출량(톤) ~ Annual Discharge (tons): Records the total annual waste in tonnes.

At this stage, different options were considered for organising the data and preparing it for model training. The waste classification column contained only two categories:

사업장배출시설계폐기물 ~ Workplace Discharge Design Waste

사업장생활계폐기물 ~ Workplace Living Waste

Since company names, contact numbers, and waste names were non-numerical data, they were unsuitable for training the model. Given that K-Means clustering only works with numerical data, this presented a limitation at first.

The initial approach involved encoding city names as numbers, but this resulted in illogical clustering and was completely discarded.

Upon recognising this, a better approach was formulated:

City data would be retained after clustering rather than encoded.

The model would focus solely on annual waste measurements, ensuring meaningful and accurate clustering.

Cleaning: Preparing the dataset

In most datasets, inconsistencies such as null values often exist, so it was crucial to clean the data before proceeding. After checking the output, the following missing values were observed:

```
기초시군구(관할관청)      0
업체명            0
연락처          480
폐기물구분          6639
폐기물명           0
연간배출량(톤)        0
```

Since the "연락처" (Contact Number) and "폐기물구분" (Waste Classification) columns were not needed for clustering at this stage, they were removed. However, considering their potential usefulness for future evaluations, the cleaning was performed using dropna(), and a separate copy of the dataset was created to avoid modifying the original dataset.

Experimentation: Finding a Suitable Clustering Approach

To apply clustering, only the most relevant columns were extracted from the original dataset, forming a new DataFrame containing:

기초시군구(관할관청) (Basic City / County / District) – Retains city names for reference.

연간배출량(톤) (Annual Discharge (tons)) – Provides numerical data required for clustering.

The .copy() method ensures that this new DataFrame is an independent copy, preventing any unintended modifications to the original dataset. This approach ensures that clustering is performed solely on the cleaned dataset while keeping the original intact for future use.

Filtering and Validating City Waste Data

To ensure accurate data extraction, the first step involves filtering a specific city’s waste emission details. The code identifies whether the city exists in the dataset and, if found, prints the city's name along with its highest recorded waste emission in tons. If the city is not present, a message is displayed stating that it was not found.

Sorting and Viewing Maximum Waste Emissions

A new dataset is created by grouping waste data by city, extracting the maximum recorded waste emissions for each, and sorting them in ascending order. This allows for a better understanding of waste distribution across different regions.

Data Preparation for Clustering

A refined dataset (df_clusterC) is created, containing only the necessary columns:

City Name (기초시군구(관할관청)) – This keeps track of which city each data entry belongs to.

Annual Waste Emissions (연간배출량(톤)) – This numerical data is essential for clustering analysis.

Keeping City Names Separate for Reference

Since clustering algorithms work with numerical data, city names are stored separately in the city_names variable. This ensures that after clustering, city names can be re-associated with their respective clusters without affecting the training process.

Applying Clustering to Waste Data

1. Standardising the Data
The StandardScaler() function is used to normalise the annual waste emission values (연간배출량(톤)). Standardisation ensures that all data points are scaled to have a mean of 0 and a standard deviation of 1. This step is essential as clustering algorithms like K-Means are sensitive to differences in scale.

```
scaler = StandardScaler()
df_scaledC = scaler.fit_transform(df_clusterC[['연간배출량(톤)']])
```

2. Applying K-Means Clustering

The number of clusters (k) is set to 6, which is determined as the optimal number of groups for classifying cities based on their waste emission.

A KMeans model is initialised with n_clusters=6 and a fixed random_state=42 to ensure reproducibility.

The fit_predict() method is used to assign each city to a specific cluster.

```
optimal_k = 6
kmeans = KMeans(n_clusters=optimal_k, random_state=42, n_init=10)
clusters = kmeans.fit_predict(df_scaledC)
```

3. Assigning Clusters to the Original Data

The identified clusters are added as a new column ('Cluster') in df_clusterC, allowing us to see which cluster each city falls into.

```
df_clusterC['Cluster'] = clusters  # Assign clusters to original DataFrame
```

4. Creating the Final Clustered Dataset
A new DataFrame, df_clusterC_final, is created to retain city names, their original waste values, and their assigned clusters.

This ensures that even though clustering was performed on numerical values, city names remain for easy reference.

```
df_clusterC_final = pd.DataFrame({
    '기초시군구(관할관청)': city_names,
    '연간배출량(톤)': df_clusterC['연간배출량(톤)'],  # Original values
    'Cluster': clusters
})
```

5. Displaying the Results
The first few rows of the final clustered dataset are printed to verify that the clustering process was successful.

Analyzing Clustered Data: Maximum Waste Per City

Once the clustering process is completed, the next step is to analyze and extract insights from the results. The goal here is to identify the city in each cluster with the maximum waste emission and display the results in an organized way.

1. Finding Maximum Waste Per City in Each Cluster
The dataset is grouped by Cluster and City (기초시군구(관할관청)).

The maximum waste emission (연간배출량(톤)) for each city is calculated and stored in a new DataFrame.

```
max_values_by_city = df_clusterC.groupby(['Cluster', '기초시군구(관할관청)'])['연간배출량(톤)'].max().reset_index()
```

2. Displaying Results for Each Cluster
We iterate through each unique cluster, sorting them for a structured output.

For each cluster, the corresponding cities and their maximum waste emissions are printed.

```
for cluster in sorted(df_clusterC["Cluster"].unique()):
    print(f"\n🔹 Cluster {cluster} - Max Waste per City:")
    cluster_data = max_values_by_city[max_values_by_city["Cluster"] == cluster]

for _, row in cluster_data.iterrows():
        print(f"City: {row['기초시군구(관할관청)']}, Max Emission: {row['연간배출량(톤)']:.2f} tons")
```

Visualizing Waste Emission by Cluster using Scatter Plots

After successfully clustering the cities based on their waste emissions, the next step is to visualize the results.

The best initial approach was to plot each cluster individually. This method allows for a clearer **comparison** of different clusters and helps in verifying whether the **printed** data from previous steps aligns with the **graphical** representation.

1. Preparing the Visualization

Markers & Colors:

- Each cluster is assigned a unique marker and color for better distinction.

- Example:

  ```
  markers = ['.', 'o', '*', 'x', 'd', 'h']
  colours = ['#272221', '#cb0303', '#00fbff', '#880079', '#780858', '#05d6c0']
  ```

2. Generating the Scatter Plot for Each Cluster

The process is iterative through all 6 clusters.

For each cluster: Filter cities belonging to that cluster. Extract the city names and their max waste values, then plot the scatter graph with customized markers and colors.

```
for cluster_num in range(0, 6):  # Iterate through all clusters
   cluster_data = df_clusterC[df_clusterC["Cluster"] == cluster_num]

   # Extract max waste per city in the cluster
   max_values_cluster = (
      cluster_data.groupby("기초시군구(관할관청)")["연간배출량(톤)"]
      .max()
      .reset_index()
   )

   # Extract values
   cities_cluster = max_values_cluster["기초시군구(관할관청)"]
   max_waste_cluster = max_values_cluster["연간배출량(톤)"]

   # Scatter plot
   plt.figure(figsize=(20, 10))
   plt.scatter(
      cities_cluster, max_waste_cluster,
      label=f'Cluster {cluster_num}',
      marker=markers[cluster_num],
      color=colours[cluster_num]
   )

   # Formatting
   plt.xlabel("도시명 / City", fontproperties=font)
   plt.ylabel("Max Emission (톤 / Tons)", fontproperties=font)
   plt.title(f"Max Waste / 폐기물 x Korean City", fontproperties=font)
   plt.legend()
   plt.grid(True)
 plt.xticks(ticks=range(len(cities_cluster)), labels=cities_cluster, fontproperties=font,
rotation=90)
   plt.show()
```

After running tests to check if the data matched each cluster individually and confirming the accuracy, the code was updated to merge all clusters into a single plot using the following lines:

```
 all_cities.extend(cities_cluster)

plt.scatter(cities_cluster, max_waste_cluster,
         label=f'Cluster {cluster_num}',
         marker=markers[cluster_num],
         color=colours[cluster_num])
```

```
plt.xticks(ticks=range(len(all_cities)), labels=all_cities,
        fontproperties=font, fontsize=8, rotation=45, ha='right')
```

With this update, instead of generating separate scatter plots for each cluster, all clusters are now displayed together in a single figure.

Revised Version

On the other hand, after successfully creating the clustering and classification, the next step was to handle regression for modeling and predicting continuous numerical values.

Starting with the clustering data, the following assignment was made:

X = 연간배출량(톤) ~ Annual waste

Y = Cluster

Then the model training began with the following code:

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
y_pred = clf.predict(X_test)
accuracy_score = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy_score:.2f}")
print(classification_report(y_test, y_pred))
```

Output:

```
Accuracy: 1.00
           precision    recall  f1-score   support

        0       1.00      1.00      1.00      1860
        3       0.93      1.00      0.96        13
        4       1.00      0.67      0.80         3

accuracy                          1.00      1876
macro avg       0.98      0.89      0.92      1876
weighted avg       1.00      1.00      1.00      1876
```

A random Forest Classifier was used as the first approach to compare performance and see the results:

```
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)
```

Surprisingly, the model achieved an accuracy of 1.00. With these results, the model was ready for future tests.

Testing Model Performance & New City Predictions

After successfully training the model, tests were conducted to evaluate its performance. This included introducing a "new city" to see how the model would classify it.

```
new_city_df = pd.DataFrame([[5000]], columns=['연간배출량(톤)'])
predicted_cluster = clf.predict(new_city_df)
```

In a dummy test, "Seoul" was used as an example with 5000 tons of waste, and the model successfully assigned it to a cluster. Further tests were performed by manually inputting new city entries to see how the model classified them.

```
new_cities_data = pd.DataFrame({
    '기초시군구(관할관청)': ["Seoul", "Busan", "Incheon"],
    '연간배출량(톤)': [1400, 700000, 9000]
})
```

To validate the model’s consistency, these new cities were introduced into the dataset, and the model predicted their respective clusters.

```
new_cities_data = df_clusterC[['기초시군구(관할관청)', '연간배출량(톤)']].copy()
new_cities_data['Predicted Cluster'] = clf.predict(new_cities_data[['연간배출량(톤)']])
print(new_cities_data.head())
```

At first glance, the model **assigned** the clusters correctly. However, since it was using existing dataset values, further verification was needed to ensure accurate classification for new future data.

Evaluating Prediction Accuracy with Randomized Data

To test how well the model generalized, a larger number of trials were performed using randomized data. Out of 50 manual tests, the model misclassified data only twice, proving its high accuracy.

Here’s how the randomized tests were conducted:

```
random_cities = df_clusterC_final.sample(n=5)
random_cities['Predicted Cluster'] = clf.predict(random_cities[['연간배출량(톤)']])
print(random_cities)
```

Below is an example where the model misclassified one entry:

#	기초시군구(관할관청)	연간배출량(톤)	Cluster	Predicted Cluster
8163	대전광역시 대덕구	3.33	0	0
68	강원특별자치도 동해시	289029.35	4	3
13952	충청남도 서산시	71.82	0	0
1417	경기도 안산시	1.74	0	0
3365	경상남도 김해시	7.22	0	0

This minor discrepancy confirms that while the model is highly accurate, there can still be small errors during prediction, especially for **edge** cases.

Discussion and Conclusions - Final Thoughts

Through this project, we discovered valuable resources and datasets that can enhance future AI models. As demonstrated throughout this research, data-driven insights play a crucial role in improving predictive models.

One of the key objectives of this project was to expand beyond conventional datasets, addressing the challenge of data encoding issues that often limit accessibility. By leveraging available data sources, we uncovered patterns in waste generation that could be instrumental in identifying solutions to reduce waste and minimize environmental impact, not just in South Korea, but in other regions as well.

Beyond this specific case, the project highlights the importance of understanding trends in different countries. In South Korea, for example, cultural tendencies may influence environmental factors, and recognizing these patterns could lead to more effective waste management strategies.

Additionally, working with diverse datasets provides valuable learning experiences for future developers, equipping them with the skills to handle complex data challenges. Thanks to advancements in AI models and supervised learning, these insights pave the way for better, data-driven solutions in the future.

References

S-저널, & 석편이규. (2017, September 1). 환경부, 자원순환 규제 개선. . . 28일부터 ‘폐기물관리법 시행규칙’ 개정안 시행. S-저널. Retrieved March 4, 2025, from https://www.s-journal.co.kr/news/articleView.html?idxno=26104

박상욱. (2025, January 6). [박상욱의 기후 1.5] 온실가스 통계 촌극과 더 복잡해진 전기차 보조금. Naver News. Retrieved April 4, 2025, from https://n.news.naver.com/article/437/0000425377?sid=102

김은경. (2023, October 18). [단독]“탄소로 항공유를”. . .LG화학, 세계 첫 ‘CCU 실증 플랜트’ 구축. Retrieved April 4, 2025, from https://n.news.naver.com/article/018/0005598885?sid=101

Fischer, H. (2025, February 19). SK chemicals to establish Waste Plastic Recycling Innovation Center in Korea. https://petpla.net/2025/02/19/sk-chemicals-to-establish-waste-plastic-recycling-innovation-center-in-korea/

Allbaro & 한국환경공단 ~ Korea Environment Corporation. (2024, January 24). 한국환경공단_사업장일반지정폐기물 배출 사업장 정보_20221231. 공공데이터포털. Retrieved March 24, 2025, from https://www.data.go.kr/data/15126438/fileData.do