Improving Air Pollution Detection

aws x DELTA

The Problem

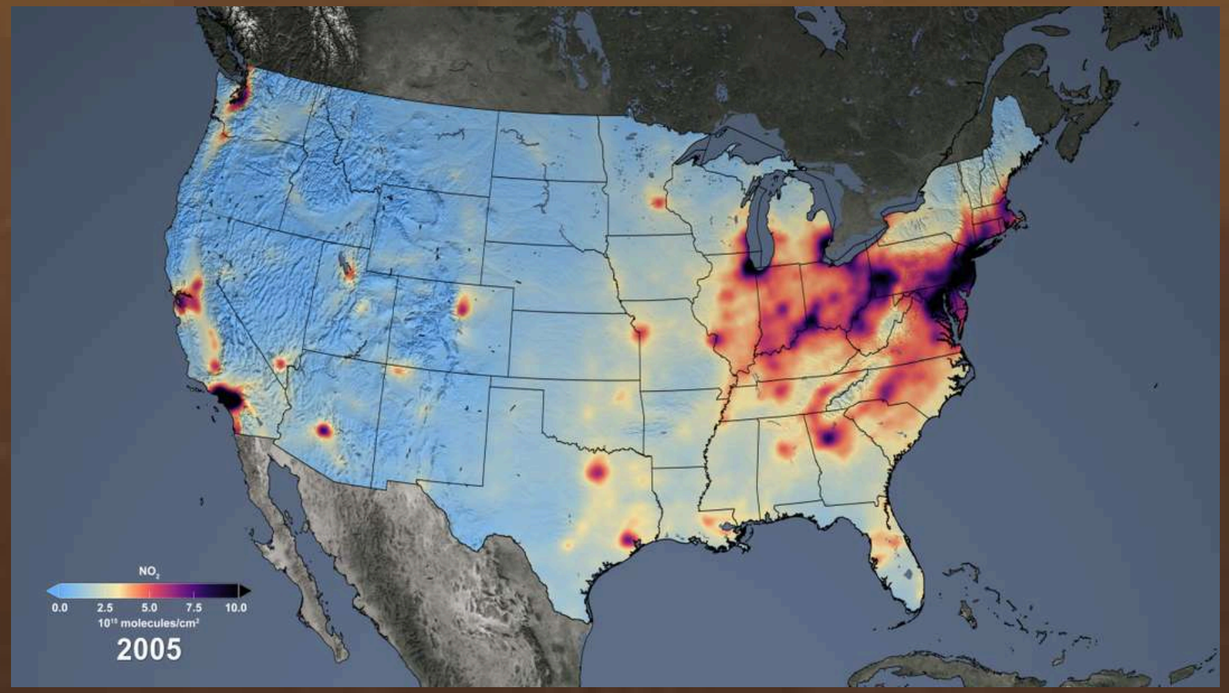
NO2 is an easy-to-produce, hard-to-detect pollutant that harms the respiratory and circulatory systems, causing inflammation, hindered lung development, increased susceptibility to infections, and even death.

NO2 contributes to environmental issues like acid rain, visibility degradation, and nutrient pollution.

The AQI (Air Quality Index) is a simple measure of air quality that helps quickly assess associated health effects and recommended actions.

OUR SOLUTION

We created an ML model to predict Nitrogen Dioxide (NO2) AQI levels using various environmental and pollutant-related features like Carbon Dioxide AQI. The model leverages historical data, including NO2 Mean, Max CO2 Value, location, and other relevant features.



WHAT OUR MODEL CAN ACCOMPLISH

SMART HOME SYSTEMS

Our model can be integrated into smart home systems that can control air purifiers and ventilation to improve air quality when higher levels are detected.

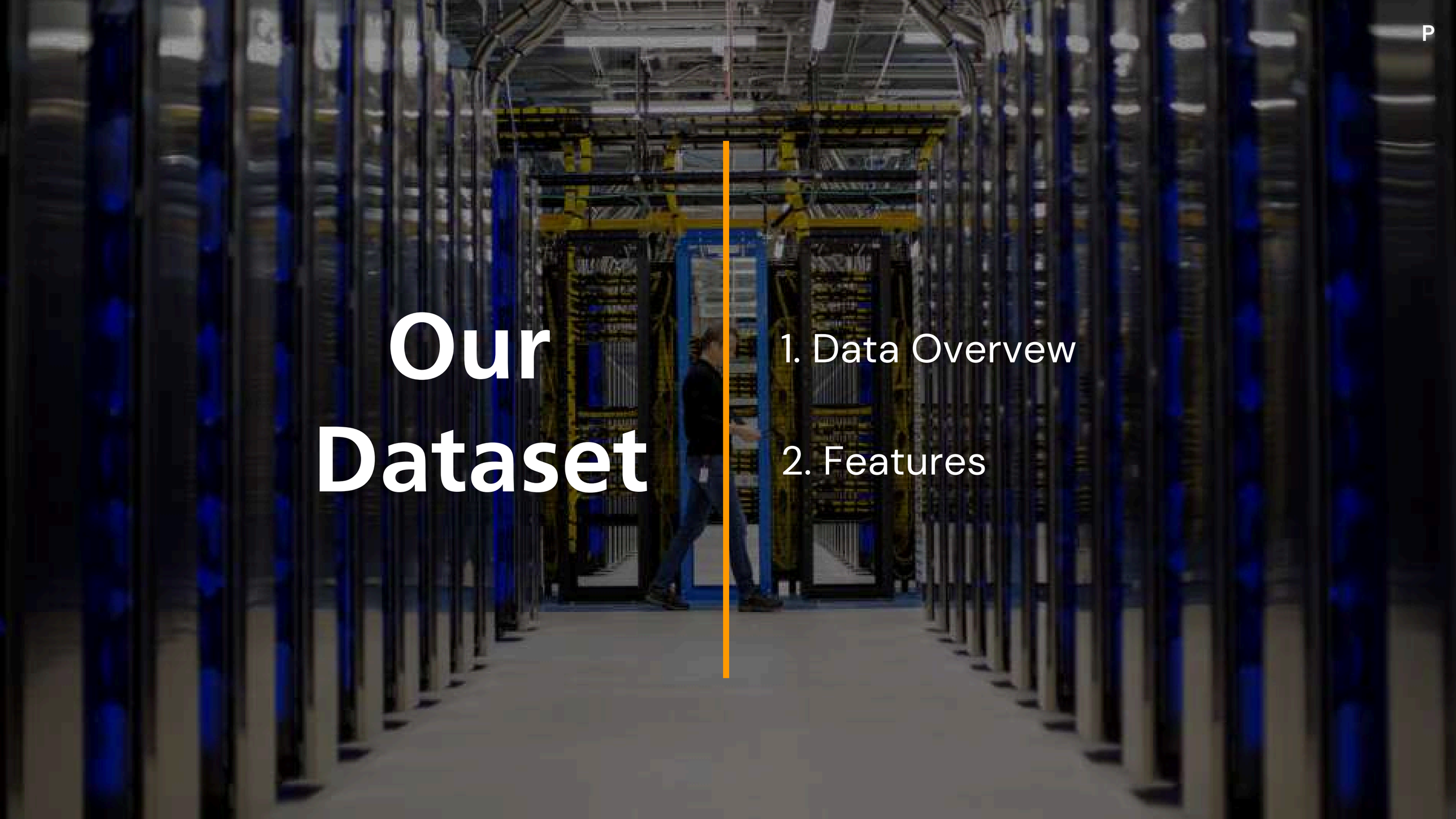
URBAN PLANNING

City planners may use our model to create real-time air quality maps based on predicted NO2 AQI levels

.

PUBLIC HEALTH

Local health departments may use our model to issue alerts via SMS or email about potential high pollution events.



Our Dataset

1. Data Overvew

2. Features

Dataset Overview

Our dataset was obtained from Kaggle and is very extensive spanning 47 states across the US and including 24 features.

1.7 Million+	Data Samples
15+ Years	2000 - 2016

OVERVIEW CONTINUED

Dataset Features

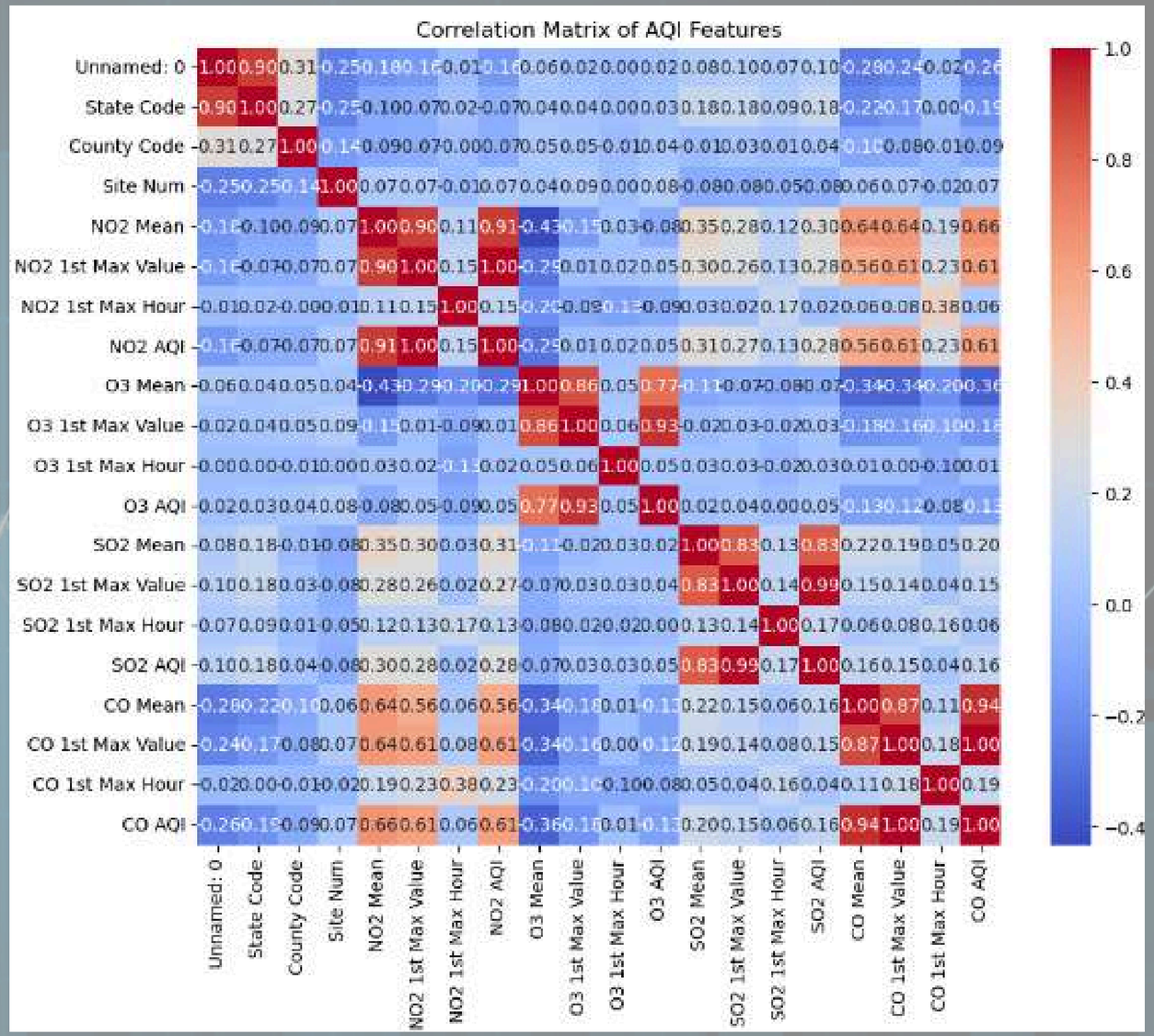
Demographics	Air Quality Metrics
Address	(Nitrogen Dioxide) NO2
State	(Ozone) O3
County	(Sulphur Dioxide) SO2
City	(Carbon Monoxide) CO
Local Date	Mean, 1st Max Value, 1st Max Hour, AQI

Exploratory Data Analysis

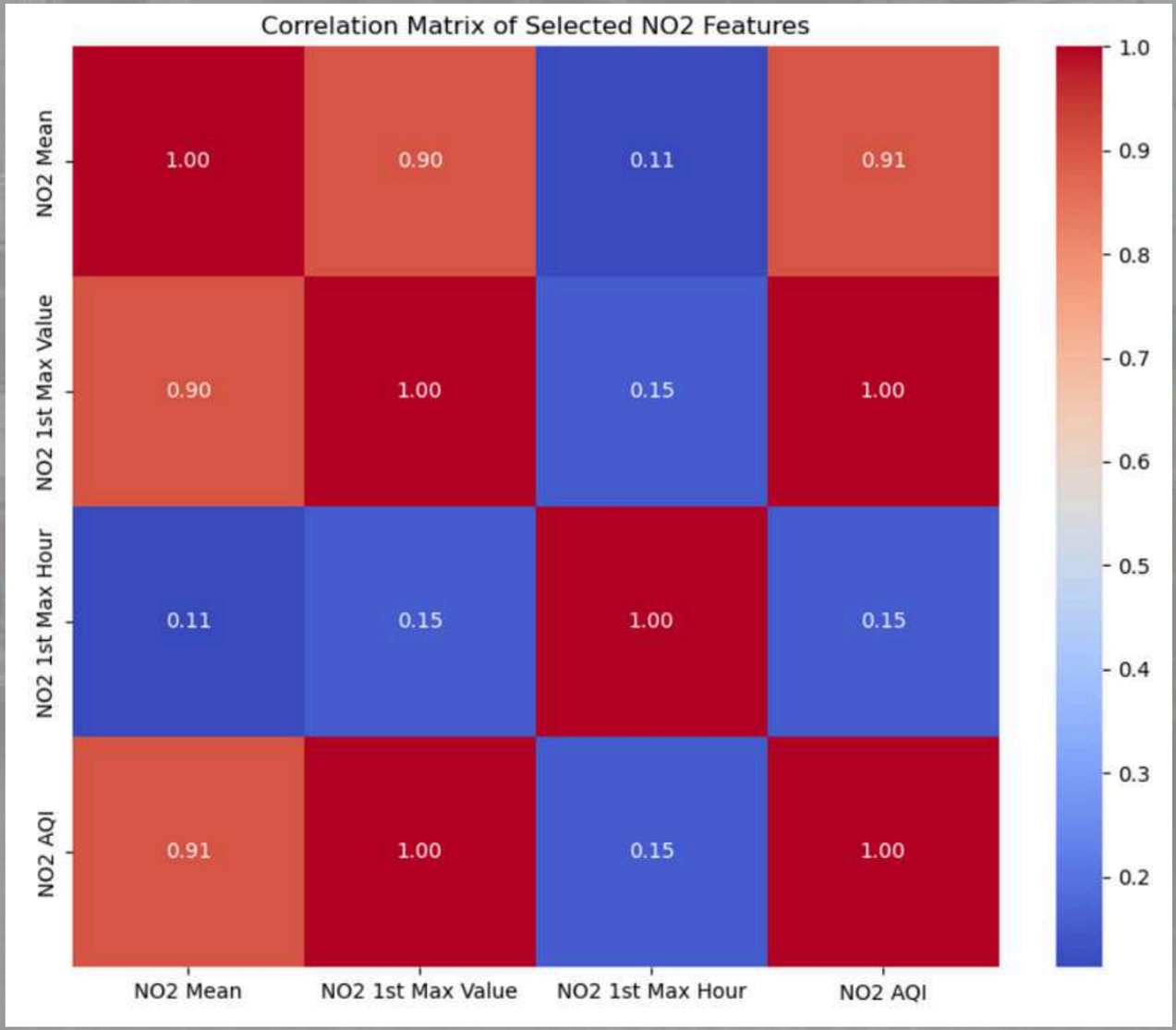
1. What Is It?
2. Box Plot
3. Correlation Matrix

What Is EDA?

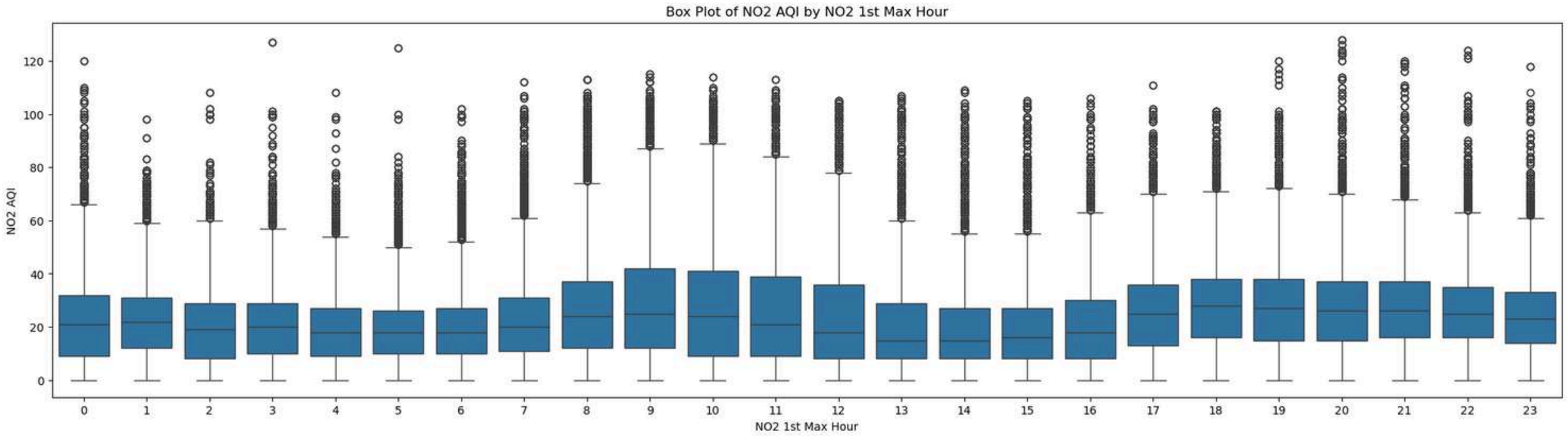
Exploratory Data Analysis (EDA) was key to identifying patterns and correlations between features using graphs and metrics. We focused on NO2- and CO2-related features to select those most predictive of AQI levels for our model.



Correlation Matrix



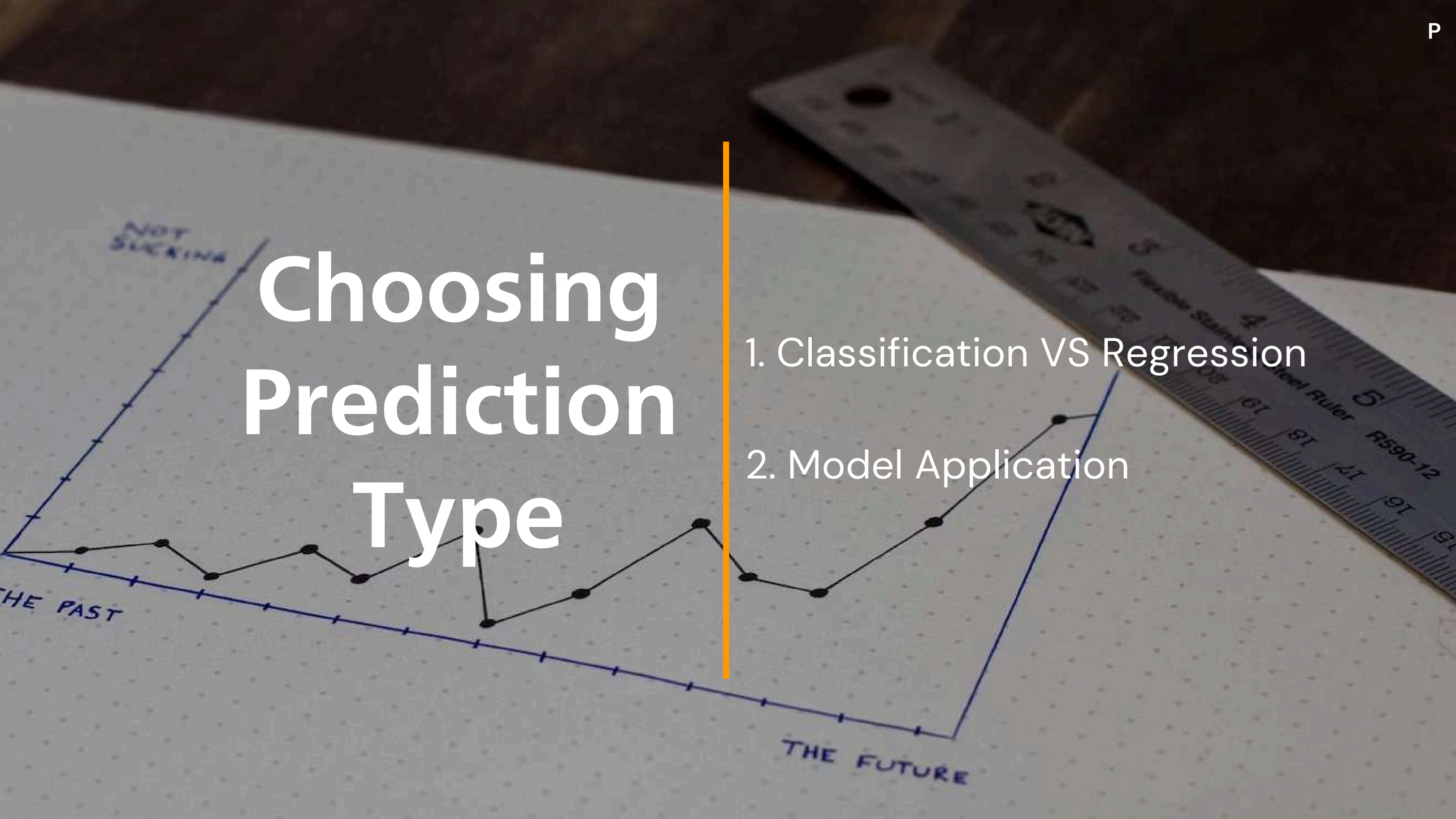
Box Plot


Box Plot of NO2 AQI by NO2 1st Max Hour
NO2 AQI
0
20
40
60
80
100
120
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
NO2 1st Max Hour

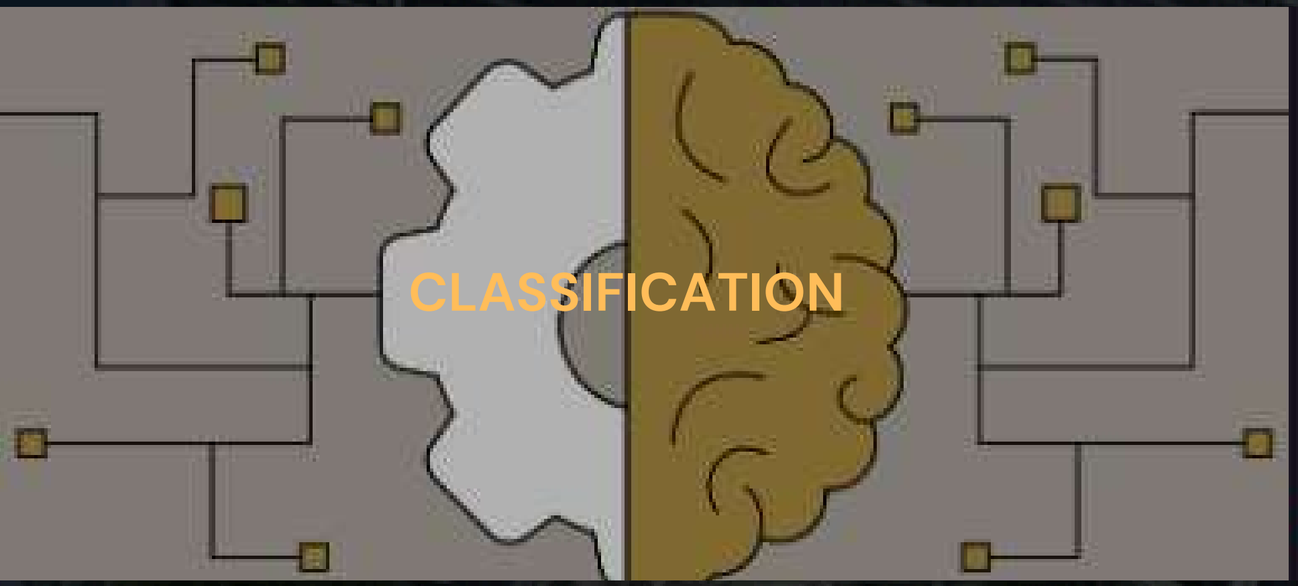
Choosing Prediction Type

1. Classification VS Regression

2. Model Application



Prediction Type



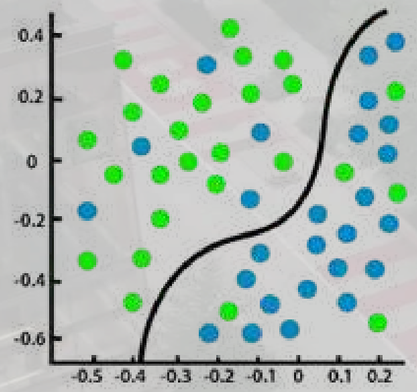
Classification predicts discrete categories (e.g., YES/NO) and outputs categorical values, which are evaluated using metrics like Accuracy and F1-Score. It is commonly used for tasks that require distinguishing between different classes or labels.

Regression predicts continuous numerical values (e.g., prices) and outputs continuous values, which are evaluated using metrics like Mean Squared Error and R-squared. It is ideal for modeling relationships and trends in data to predict future values.

Algorithm Application

Classification

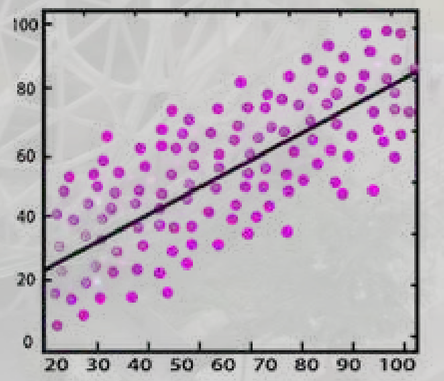
- Classify daily NO2 AQI into air quality categories like Safe, Moderate or Unhealthy
- Identify the predominant pollutant in specific locations, such as NO2 or O3
- Categorize seasons by typical air quality patterns



Classification

Regression

- Predict exact NO2 AQI levels for specific days based on historical data
- Estimate continuous NO2 concentration levels over time in a city
- Models the relationship between NO2 and and features to predict



Regression

Data Processing

Feature Deletion

Deleted features include described units used to measure each gas and any other NO2 related features

Label Encoding

Used Scikit-learn's LabelEncoder converted missing values back to NaN

Imputation

Using scikit-learn's SimpleImputer for mean imputation

Scaling

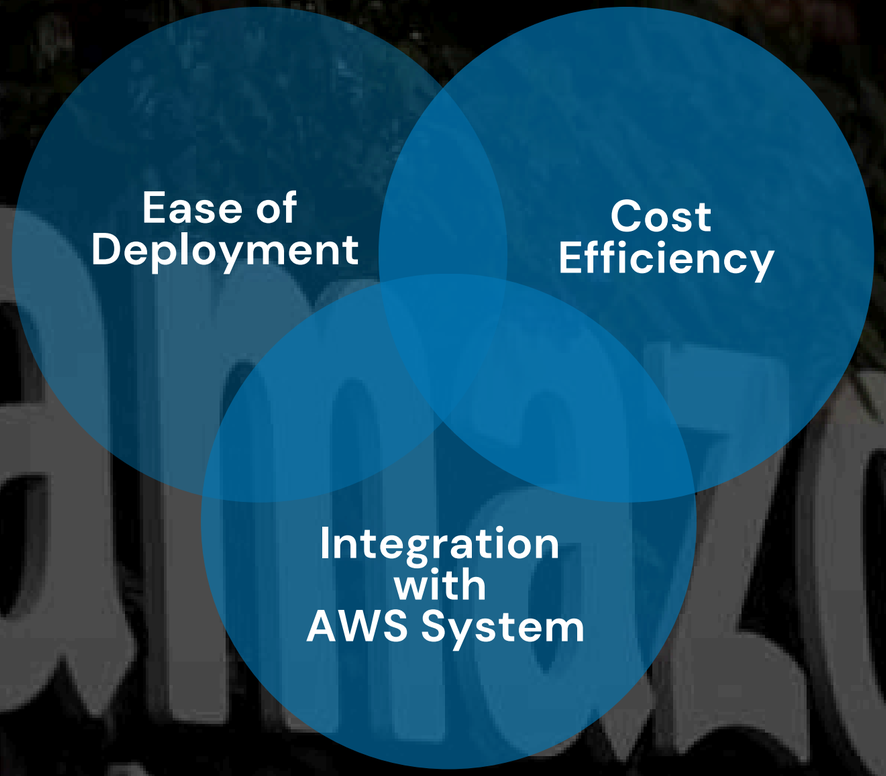
Using Scikit-learn's StandardScaler to normalize the dataset for consistent scaling across features

Data Balancing

Not Necessary Due to Continuos Data Set

Why We Chose AWS Sage Maker

SageMaker provides a very straightforward way to develop new ML projects within a few clicks. It eliminates the complex and manual setup of server configuration, introducing brevity. Not only that, it supports the entire cycle of an ML project, reducing the time and effort needed from data preprocessing all the way to model monitoring.



Amazon offers an AWS free tier, allowing us to utilize services such as S3 storage and notebooks without incurring any costs. This was incredibly substantial for us, as it allowed us to explore different methods and develop our model entirely risk-free.

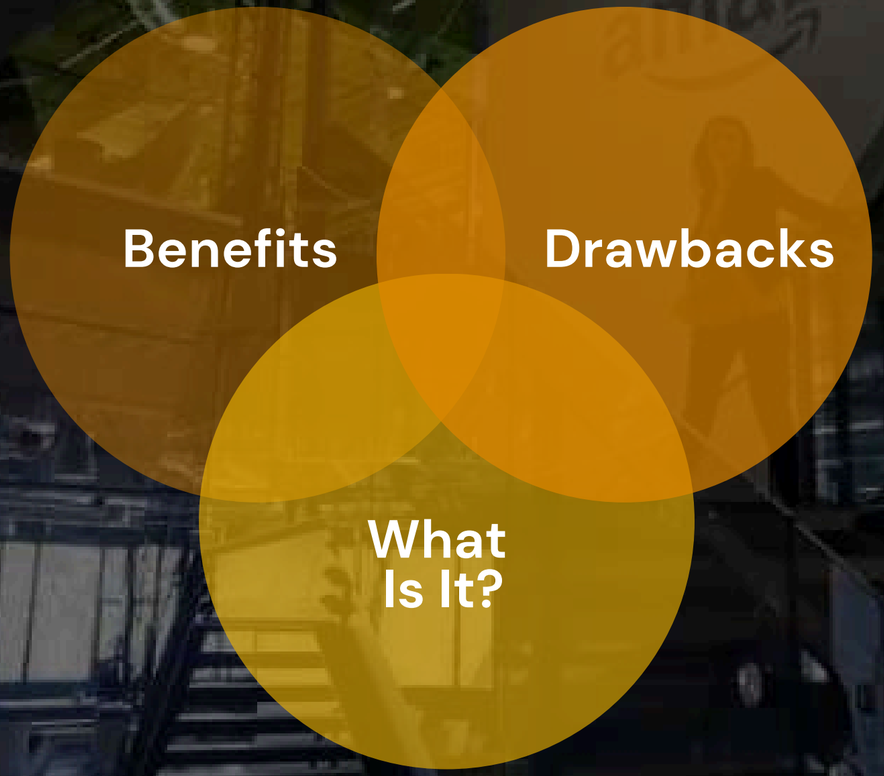
Sagemaker's seamless integration with the AWS environment allows us to use services such as S3 to efficiently store and retrieve large datasets. This integration makes it easy to load datasets directly into SageMaker and train ML models without worrying about storage limitations.

KNeigbours Regressor

1. What Is KNN?
2. Our Model
3. Efficiency

KNEIGBOURS REGRESSOR

What Is KNN?



Simple and intuitive to implement. Effective for non-linear relationships without requiring a complex model structure. Adaptable to dynamic datasets, as the model automatically updates with new data points

Sensitive to the choice of the number of neighbors (k) which can impact accuracy. Computationally expensive for large datasets, as it requires training data points for each prediction

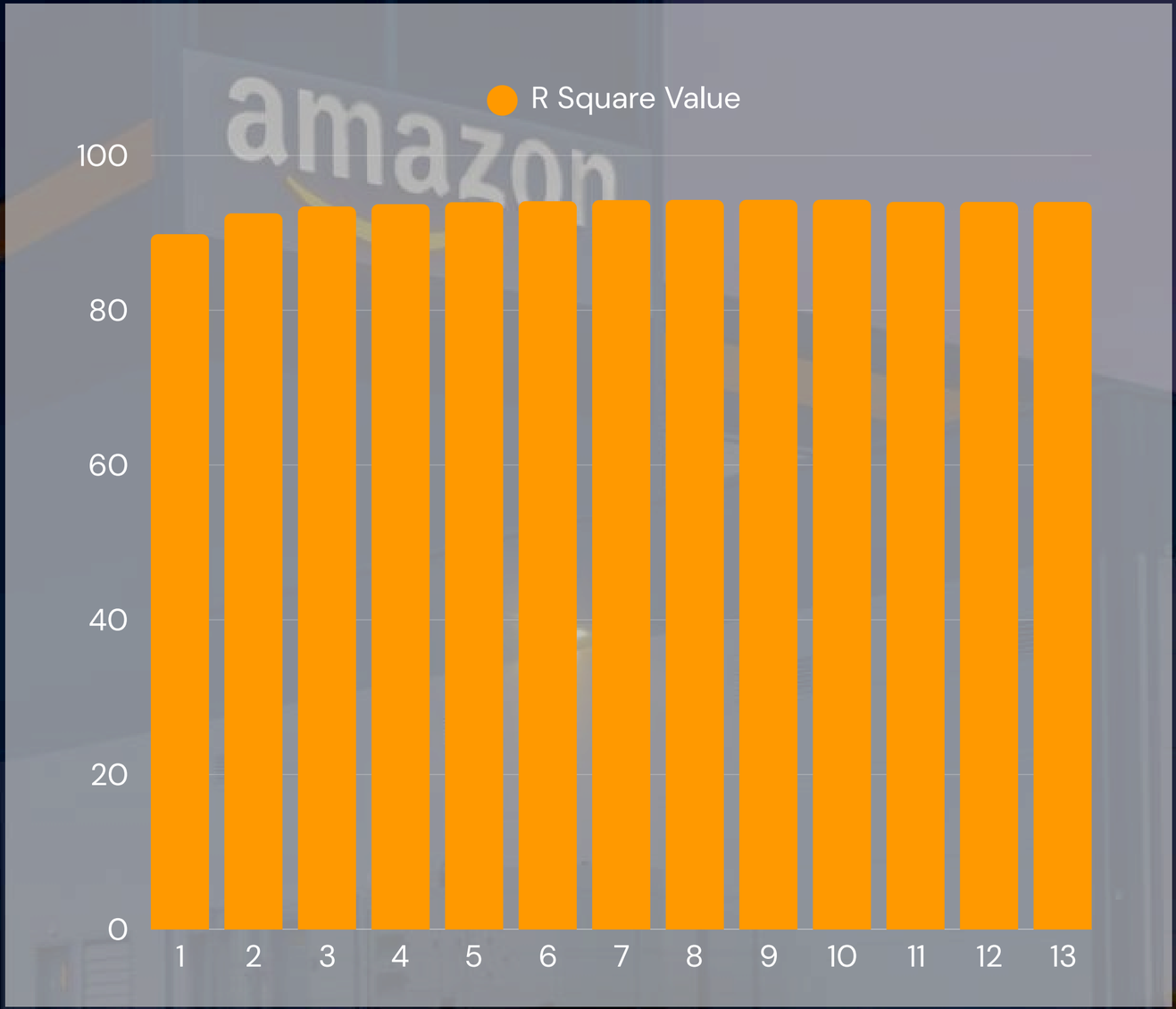
KNeighbors Regressor is a machine learning algorithm that predicts target values by averaging the values of the nearest neighbors in the feature space

Our Model

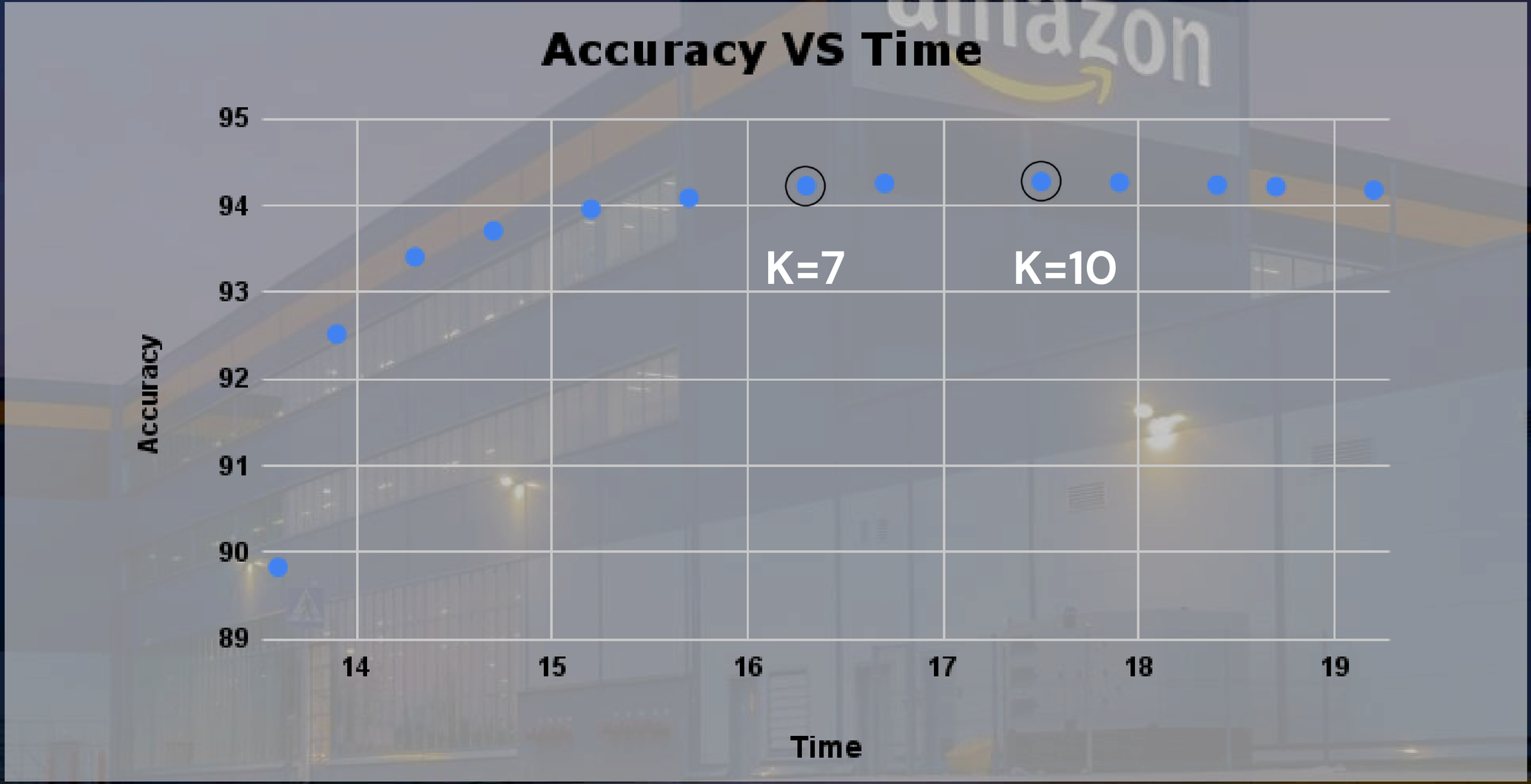
Best Performing K Value at 10

Max R^2 of 0.9428

Total Time Of 176 Seconds



Efficiency


Accuracy VS Time
95
94
93
92
91
90
89
Accuracy
14
15
16
17
18
19
Time
K=7
K=10

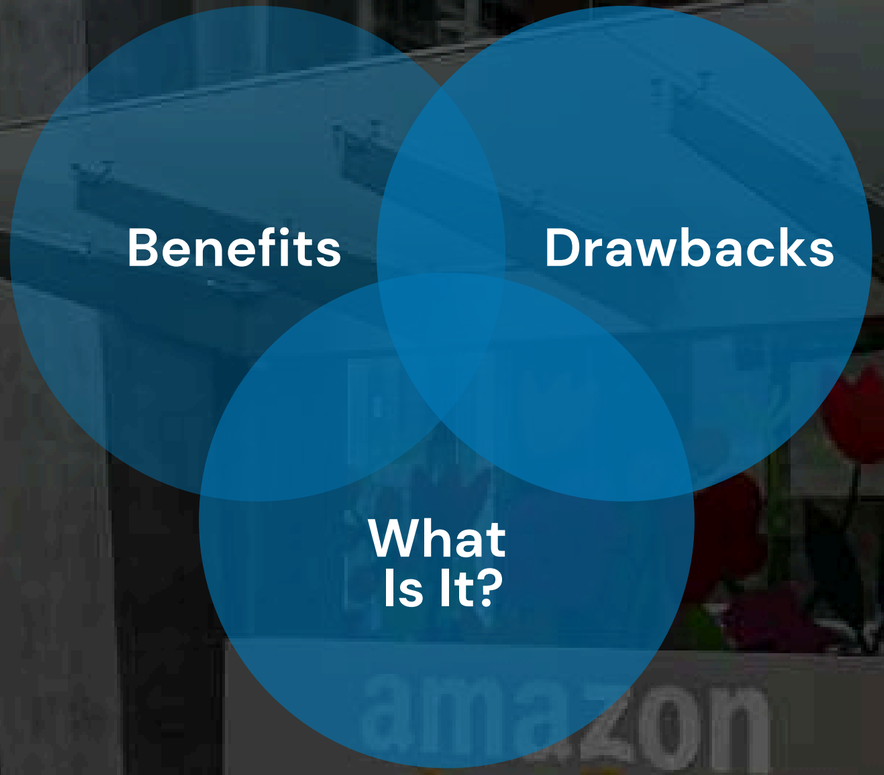
Decision Tree Regressor

1. What are Decision Trees?

2. Our Model

DECISION TREE REGRESSOR

What Are Decision Trees?



Decision Trees are highly interpretable, handle both numerical and categorical data, and capture non-linear relationships without complex transformations.

Decision Trees are prone to overfitting, sensitive to data variations, and require careful tuning of parameters like tree depth to avoid poor generalization.

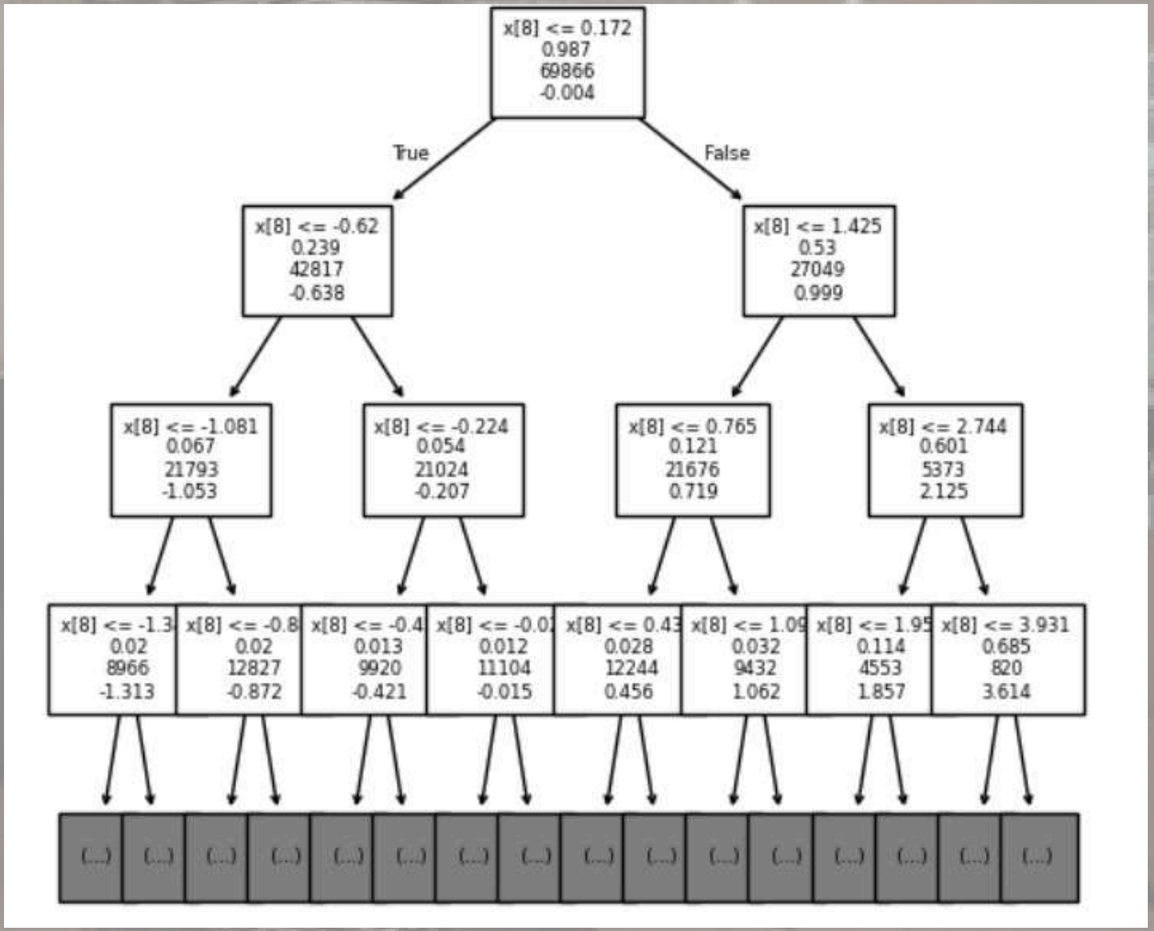
Decision Trees are a machine learning algorithm that uses a tree-like model of decisions, where features are split recursively to predict outcomes.

Our Model

Max R^2 Of 0.99998

Total Time Of 0.989 Seconds

Max Depth Of 45



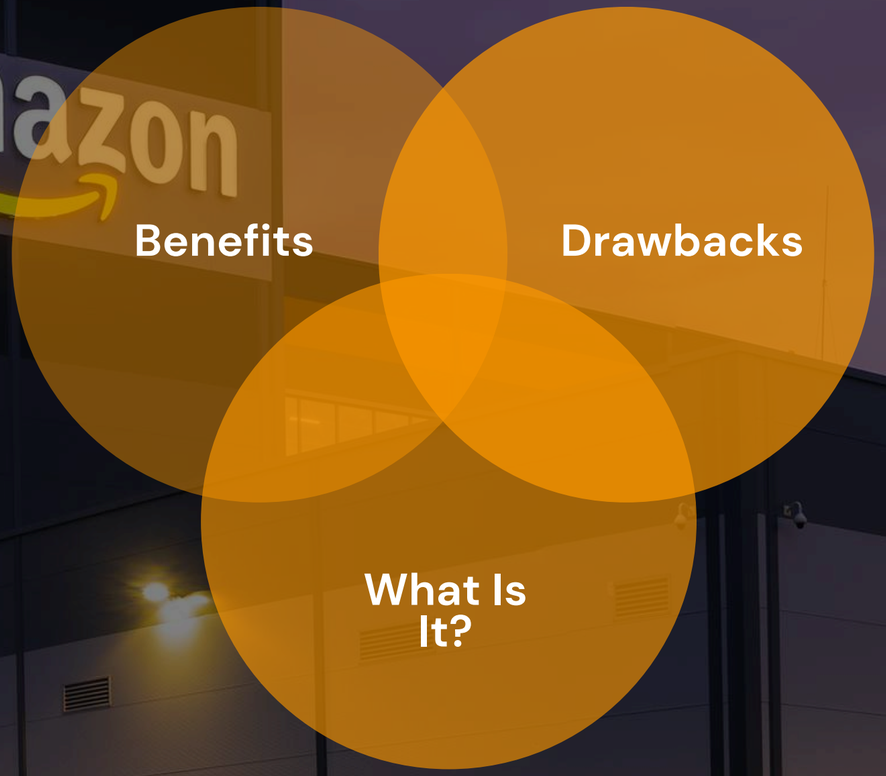
Linear Regression

1. What Is Linear Regression?

2. Our Model

What Is Linear Regression?

Linear regression is quite simple to implement. Not only that, it's easy to interpret and find strong insights into the relationships between variables. Additionally, it requires minimal computational resources and is very efficient for larger datasets.



Linear regression assumes that there is a linear relationship between variables, which is not always true. Also, it is sensitive to outliers, which will significantly skew the results.

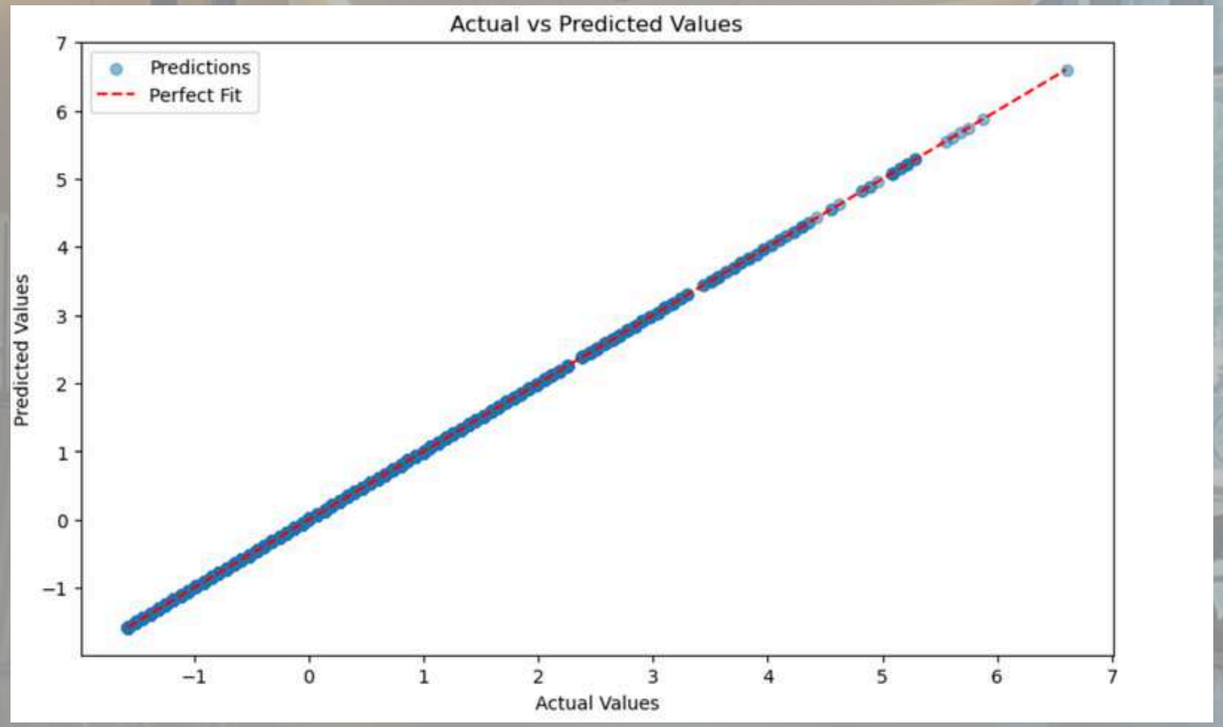
Linear regression models the relationship between a dependent variable and multiple independent variables by fitting a linear equation through observed data. The goal is to minimize the error, or the difference, between predicted values and actual values. by adjusting the line of best fit.

Our Model

Max R^2 Of 1.00

Total Time Of 2.49 seconds

Mean Absolute Error: 6.72e−16



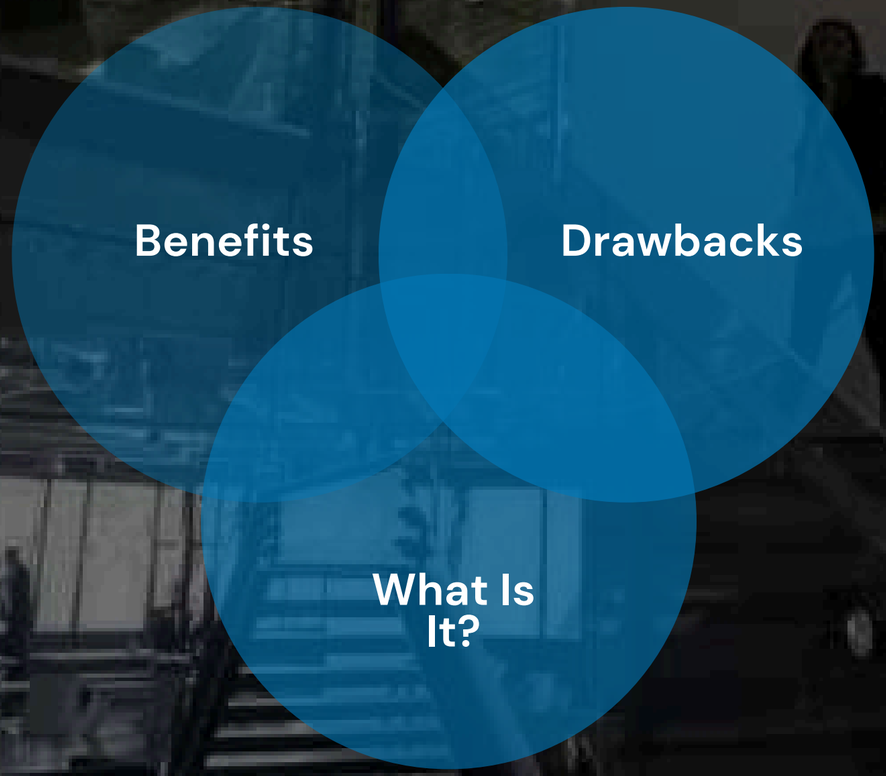
Actual VS Predicted

MLP Regressor

1. What Are Neural Networks?

2. Our Model

What Are Neural Networks?



Can learn complex patterns and relationships -> can capture nonlinearities
& Versatile
& Easily scalable

Prone to overfitting
& Computationally expensive
& Performance is sensitive to hyperparameters
& Susceptible to gradient vanishing or exploding -> hinders convergence

Neural network that has multiple layers of interconnected nodes (neurons) including an input layer, one or more hidden layers, and an output layer

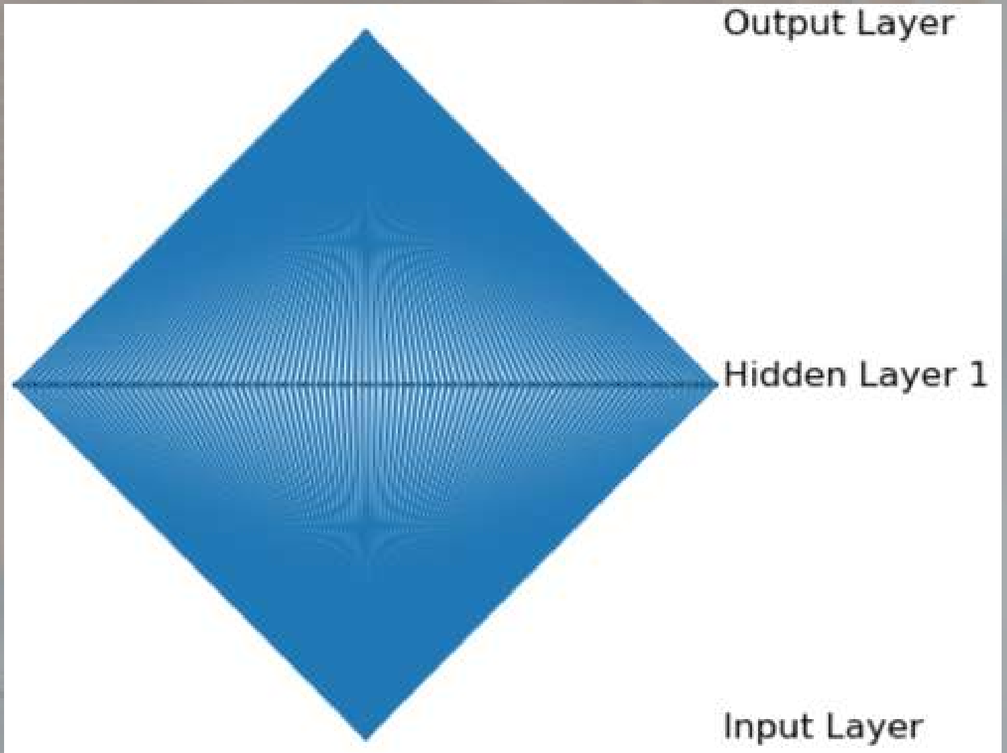
Our Model

Max R^2

Of 0.99988

Total Time Of

7.619 Seconds



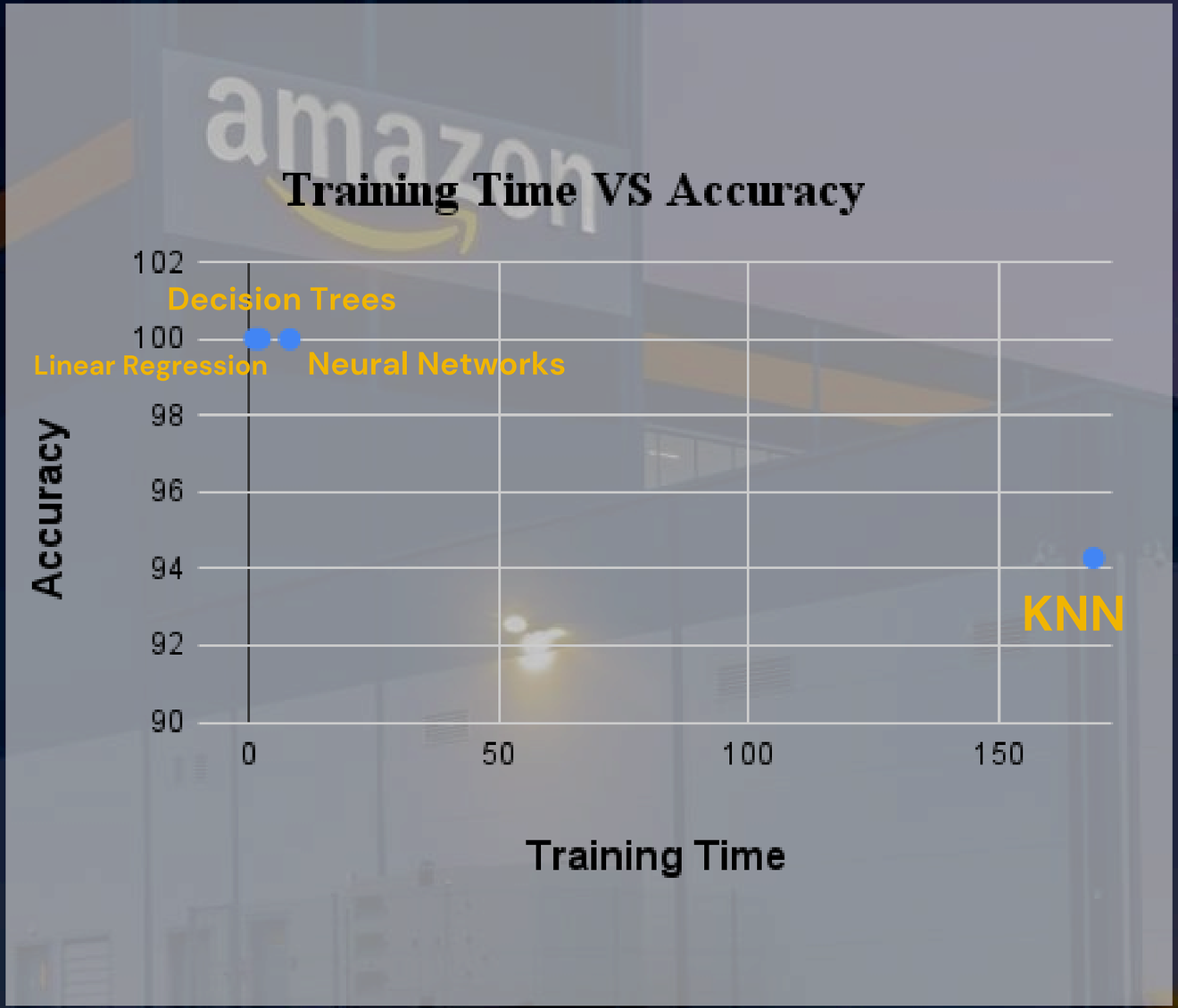
Neural Network Architecture

Final Reccomenation

Decision Trees

~100% Accuracy

Least Training Time



Prerak Mahajan

Michael Zhang

Sophie Liu

On a more personal note...

Dear AWS & Delta Careers,

We are sincerely thankful for the opportunity to delve into the machine learning industry and contribute to its innovative future.

Thank you,

Prerak, Michael, Sophie