**PROJECT REPORT:**

**Karachi Weather AQI Predictor**

Project Overview

This Project forecasts the Air Quality Index of Karachi for the next three days using the historical and pollutant data containing (PM\_2.5,PM\_10,NO,O3 e.t.c).This Project includes fetching of data set then feature engineer on that set ,training models with their evaluations using evaluation metrics and a web app to experience the predictions using modern data science ,machine learning tools and APIS.

**Modules and Features**

**1. Data Fetching:**

API Used: OpenWeather API (historical).

**Functionality:**

* Fetches raw weather and pollutant data for Karachi using co-ordinates lat and longitude data fetched is based on hours.
* Historical data backfilled for up to 100 days using start and end endpoints.
* Implements error handling for API rate limits and response validation.

**2. Feature Engineering**

**Raw Features:**

PM2.5, PM10, NO2, and O3 pollutant levels.

AQI values from the raw API response.

**Derived Features:**

Time-based features: Hour, day, month, day of the week, and weekend indicators.

Rolling statistics: Moving averages and standard deviations for PM2.5 and PM10 with the window of 3.

Lag features: Previous values of pollutants and AQI for modeling temporal dependencies.

Change rate: Day-over-day changes in pollutant levels.

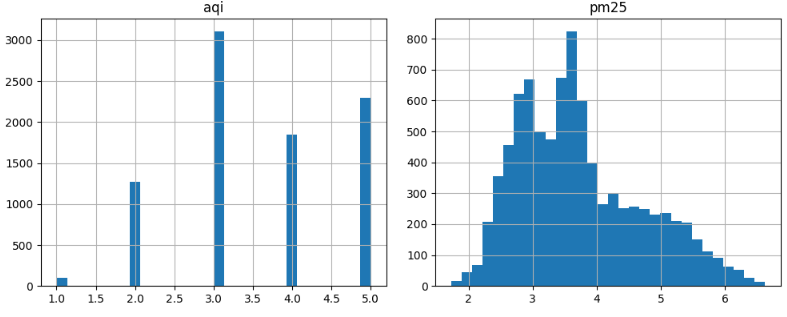
**Target Variable:**

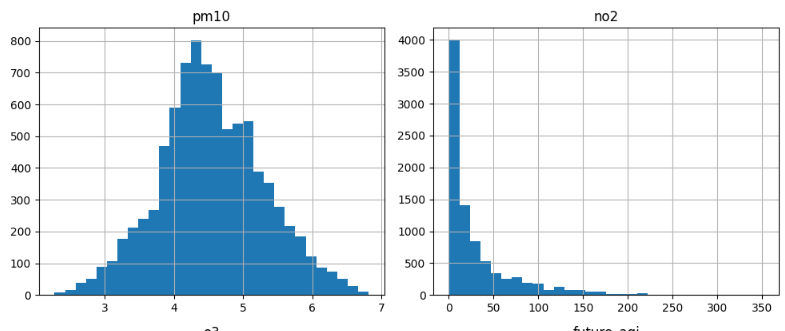
This keeps the value of the next 72nd AQI column means the forecast for the third day.

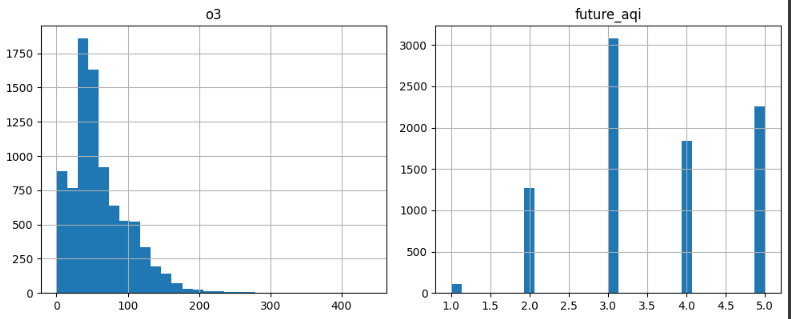
By this we’ll have last 72 rows that have future\_aqi null which will later on use to predict the aqi on those cells.thats why we donot drop all null values in the data set or fill with median.

**Data Cleaning (EDA):**

* Missing values represented by -9999 were replaced with NaN in the dataset.
* The columns except Future\_AQI were identified, excluding the target variable future\_aqi
* Missing values in these features are filled using **median values**, as the data is skewed.

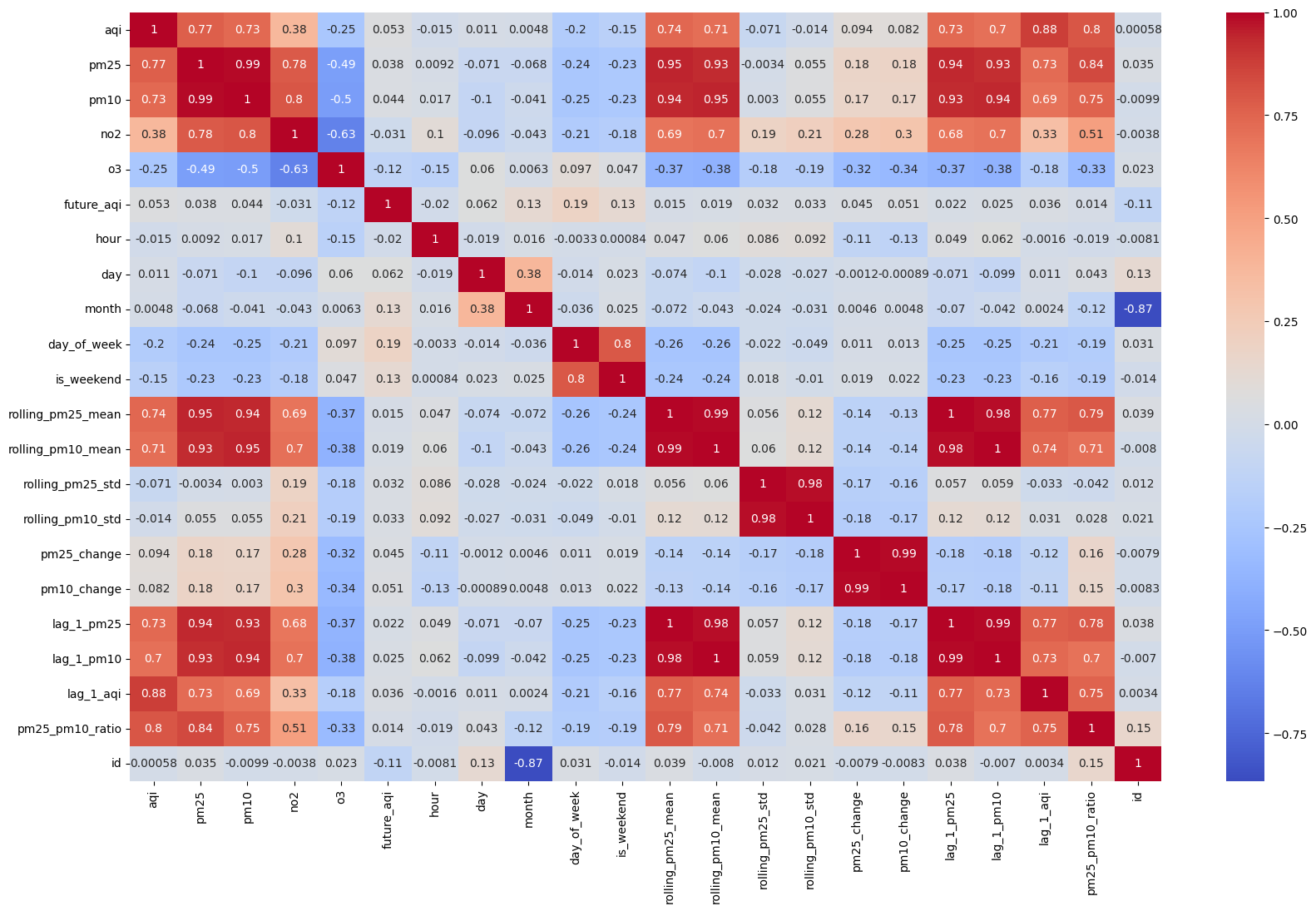






* **Z-Score Method** was applied to identify and remove extreme outliers in the pm25 and pm10 columns, rows with **z-scores greater than 3** as they are the 3 s.d away from the mean.

**Correlation Between Features:**

****

**3. Feature Store Integration**

Platform: Hopsworks.

Description:

Stores computed features and targets in a feature store named air\_quality\_features\_cleaned.

4. Model Training and Evaluation

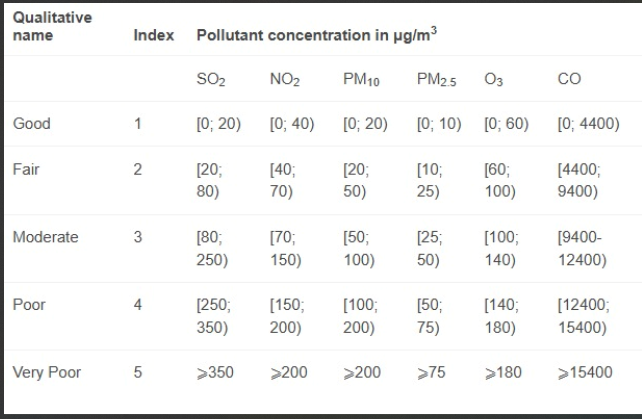
Models Used:

Logistic Regression.

Ridge Classifier.

Random Forest Classifier.

I used Classification models instead of Regrssssion because here the aqi values are not continuous.8619 rows only contains 5 values which is the evidence that it is the multiclass classification.



Evaluation Metrics:

Classification Metric.

Confusion Metric.

Accuracy Score.

Precision Score.

Findings:

Random Forest Classifier performed well with an accuracy score of 81.

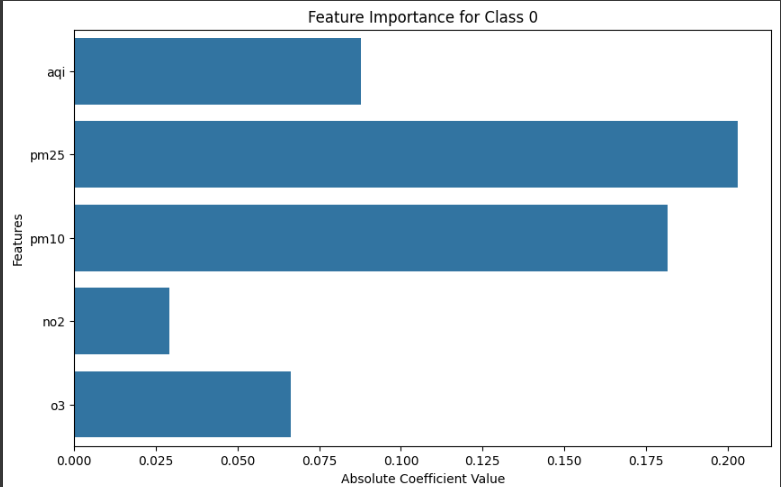
5. Explainability Tools

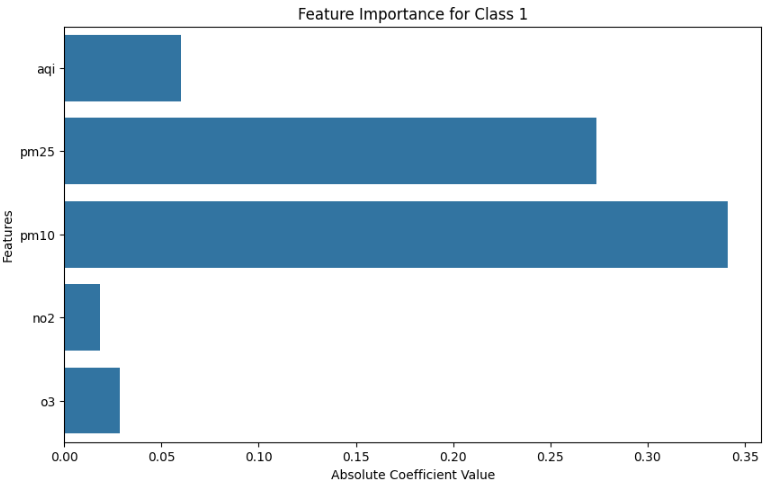
SHAP:

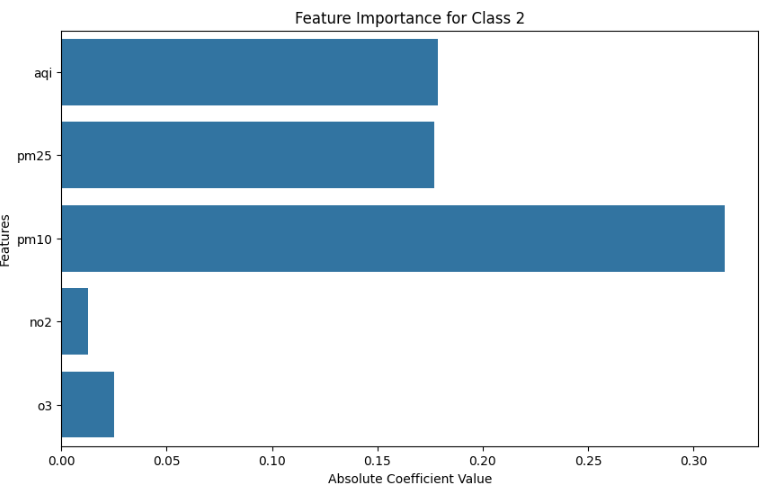
Visualized feature importance and their impact on AQI predictions.

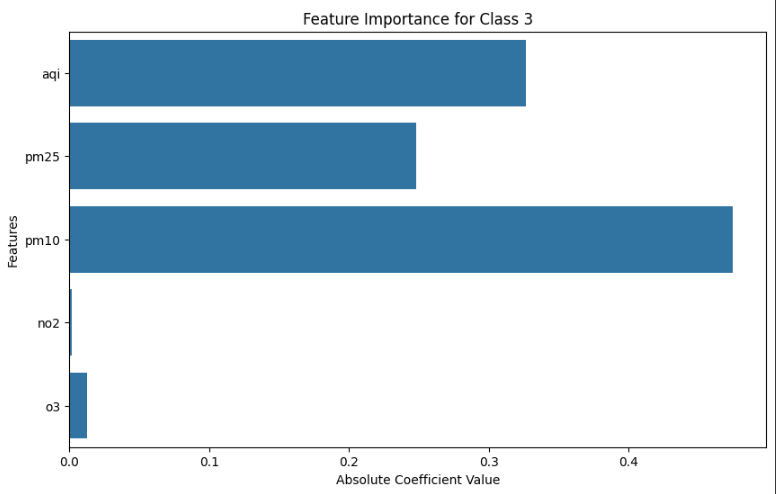
Classes depends upon the dataset having nuber of unique aqi values.

Logistic Regression:

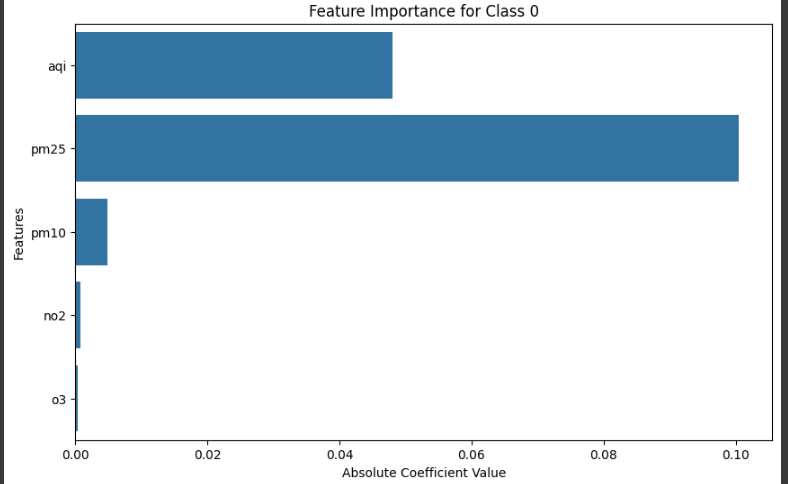


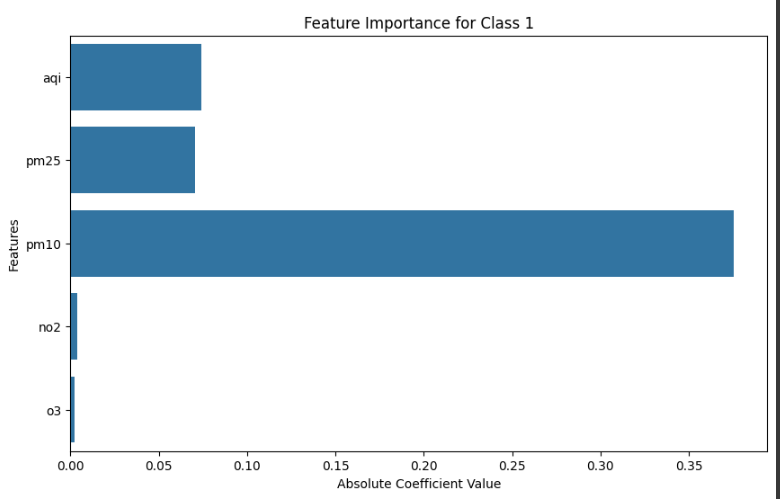


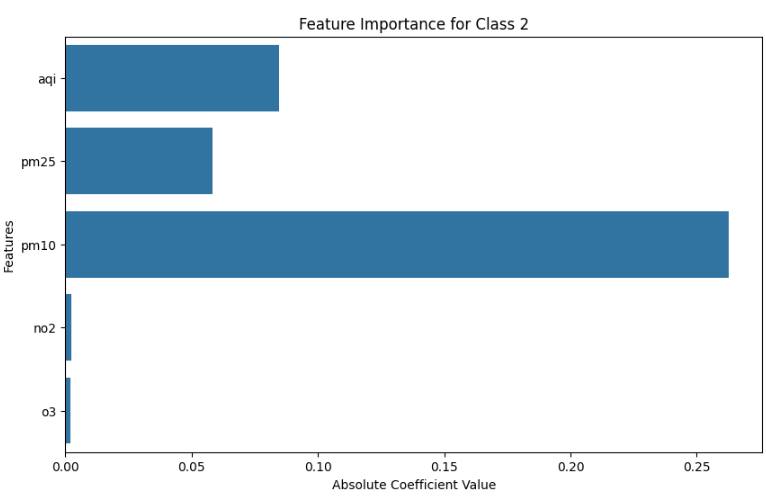


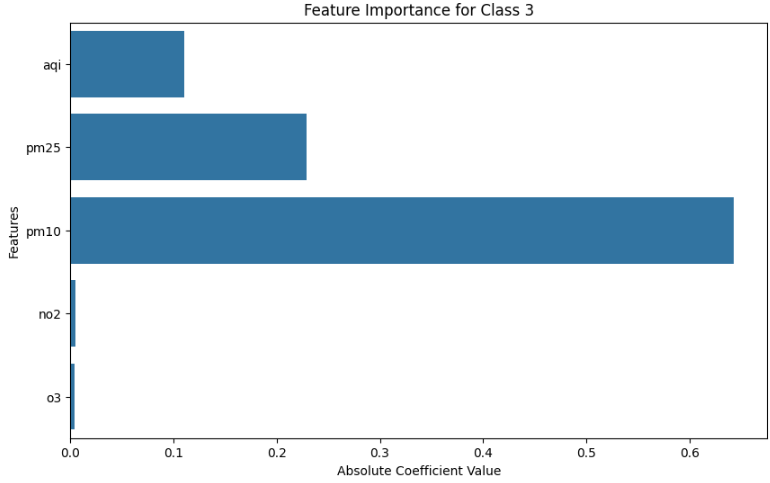


Ridge Classifier:

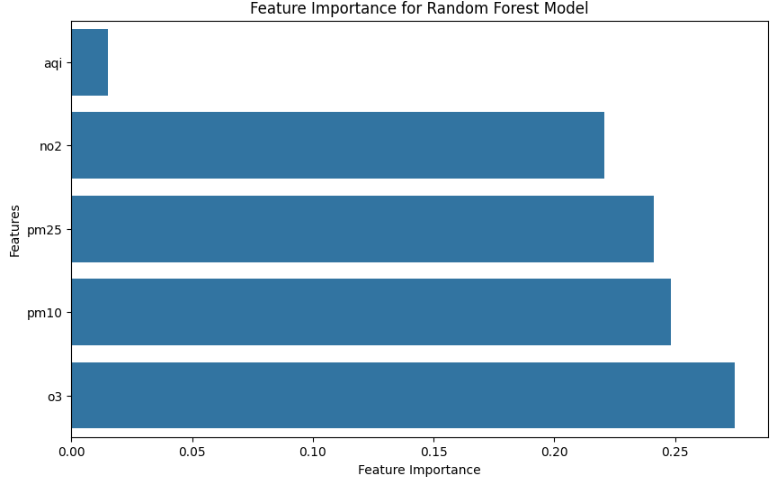








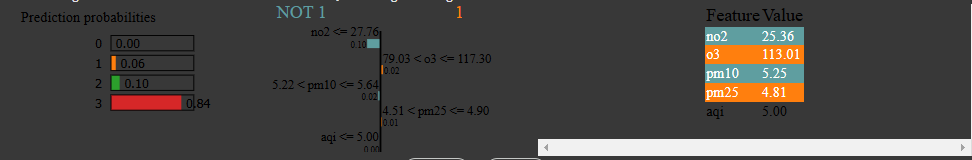
Random Forest:



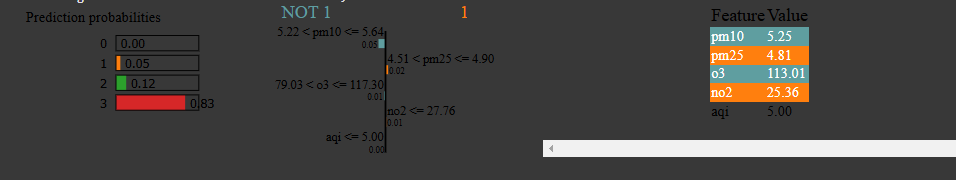
LIME:

Provided explanations for individual predictions.

Logistic Regression:



Random Forest:



**6. Interactive Dashboard**

Frameworks: Streamlit.

Functionality:

* Displays real-time AQI and show current weather and forecasts for the next three days.
* Visualizes trends using plots and charts.
* Alerts users for hazardous AQI levels.

**7. Automation Pipelines**

CI/CD Tools: GitHub Actions.

Workflow:

Feature extraction pipeline runs hourly.

Model training pipeline runs daily 24 hours.

When model training pipeine runs it splits the data set having future\_Aqi not null rows and having future\_aqi null model train on the data set having not null rows and then the null row dataset use for prediction in the app this null values dataset also updated hourly on the feature store.

Ensures timely updates to predictions and model retraining.

**8. Deployment and Scalability**

Containerization: Docker.

**Conclusion**

This project successfully demonstrates a scalable, automated pipeline for AQI prediction in Karachi. The integration of feature stores, advanced feature engineering, and an explainable model makes it user-friendly. Future enhancements, such as incorporating deep learning models (ANN) can make increase its accuracy.