Big Data Analysis and Project – Part C

Air Aware: AQI Prediction and Analysis

Implementing data-driven techniques to predict air quality index and inform public health decisions

Problem Description

The project is focused on predicting Bangalore's Air Quality Index (AQI) by combining datasets from Tutiempo (Tutiempo 2024) and Weather Map (Weather Map 2024), containing various weather and air quality data from 2013 to 2018. The question posed is whether the air quality of one year, influences the air quality of the next year.

Pre-processing

The preprocessing combined meteorological data from HTML files and AQI data from CSV files (2013-2016), resulting in a unified dataset for environmental trend analysis. Hourly PM2.5 values (2013-2018) were averaged (Air Now n.d.) daily and monthly, then merged with the second dataset's features. Unimportant variables were excluded. Exploratory data analysis focused on PM2.5 levels. The final dataset of quantitative variables, with PM2.5 as the response variable, was used for regression modeling.

Testing different models

The dataset is split into training and testing sets, with 30% of the data reserved for testing and 70% for training (for every model). Linear Regression was applied but yielded moderate predictive performance, suggesting the need to explore more advanced models for better accuracy.

Ridge and Lasso Regression

In Ridge regression, the `alpha` parameter controls the strength of L2 regularization, which penalizes the sum of the squared coefficients to prevent overfitting and manage multicollinearity.

Using **`GridSearchCV`**, the best `alpha` value found was `40`, indicating strong regularization. This results in a model that is more stable and less sensitive to noise but does not eliminate any features. The performance was evaluated through **cross-validation**, yielding a negative **mean squared error** of approximately `-3663.34`, suggesting a good fit but not the best among the compared models.

Lasso regression applies L1 regularization, which penalizes the absolute values of the coefficients and can drive some of them to zero, effectively performing feature selection. The best `alpha` found was `1`, balancing regularization and model complexity. This model simplifies feature selection by excluding less relevant features. The **cross-validation** score yielded a negative **mean squared error** of about `-3665.66`, which is slightly better than

Ridge regression, indicating a marginally better performance with feature selection and regularization. The MAE was 44.51, the MSE was 3627.81 and RMSE was 60.23.

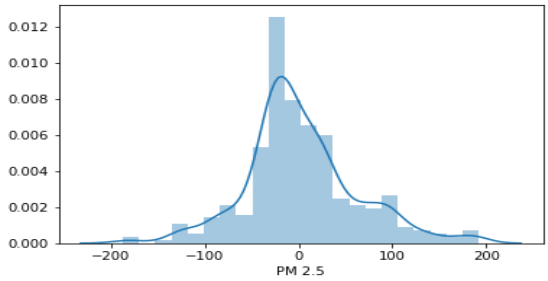

Fig 1. Residual visualization for Lasso Regression

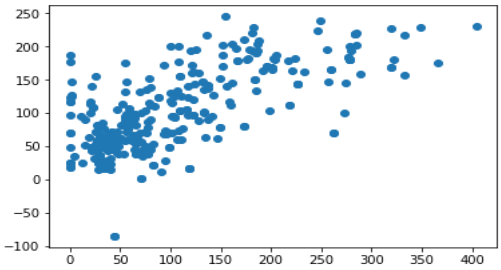

Fig 2. Scatterplot of the residuals

Decision Tree Regressor

The Decision Tree Regressor perfectly fits the training data (R^2 = 1.0) but shows lower performance on the test data (R^2 ≈ 0.69) and cross-validation (mean score ≈ 0.40). This suggests that while the model captures the training data very well, it overfits and does not generalize as effectively to new, unseen data. Hyperparameter tuning for the Decision Tree Regressor was performed using GridSearchCV with an extensive parameter grid. The cross-validated negative mean squared error was -3132.10. The model achieved an MAE of 40.14, MSE of 3171.81, and RMSE of 56.32 on the test set, indicating the best performance compared to the previously tested Ridge and Lasso regression models.

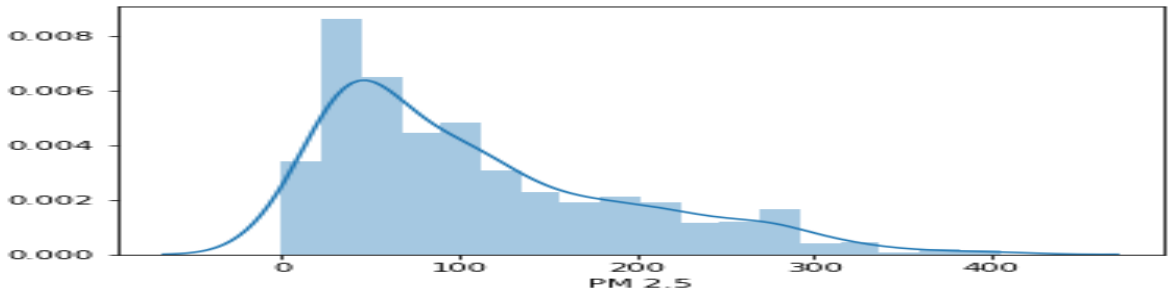

Fig 3. Residual visualization for Decision Tree Regressor

XGBoost Regressor

The XGBoost Regressor was trained and evaluated, showing an R^2 score of 0.8606 on the training set and 0.7211 on the test set. The mean cross-validation score was 0.6583, indicating good overall performance but with some overfitting. The residuals' distribution plot further helps to understand the model's prediction errors.

The XGBoost Regressor, after hyperparameter tuning with RandomizedSearchCV, achieved the best parameters with a subsample of 0.8, 800 estimators, a minimum child weight of 4, a maximum depth of 10, and a learning rate of 0.05. The model's evaluation metrics on the test set showed an MAE of 18.66, MSE of 1281.45, and RMSE of 35.80, indicating strong performance. The XGBoost model outperformed the Decision Tree model.

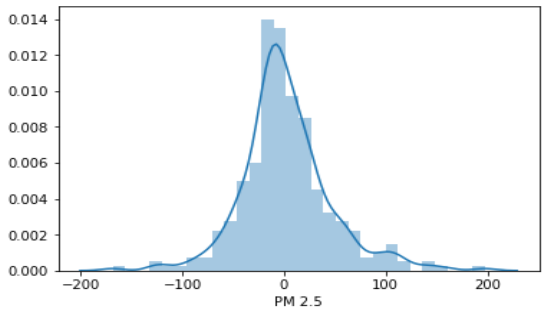

Fig 4. Residual visualization for XGBoost Regressor

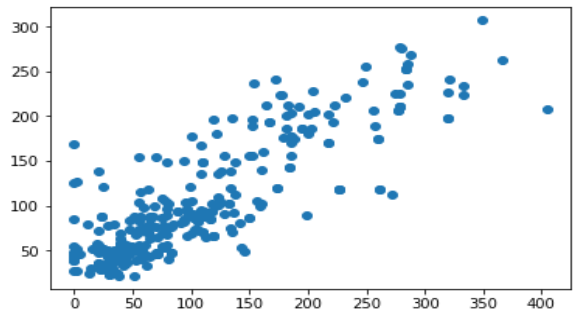

Fig 5. Scatterplot of residuals

Rationale and Results

- Hyperparameter tuning with `GridSearchCV` systematically tests a range of alpha values or parameter grids (e.g., for the Decision Tree Regressor).
- For the XGBoost Regressor, which includes parameters such as `n_estimators`, `learning_rate`, `max_depth`, `subsample`, and `min_child_weight`, `RandomizedSearchCV` was used to reduce computational complexity by sampling from the hyperparameter space instead of evaluating all possible combinations (Kharwal 2023).
- Cross-validation ensures robust model performance, providing reliable estimates (Kharwal 2023).
- The models were evaluated using MSE, RMSE, and MAE, which are the most common evaluation metrics and are easy to understand (Chugh 2020).

Conclusion:- The XGBoost Regressor performed the best, indicating that air quality trends from one year can influence the next. Further refinement of the question is not necessary currently.

References

Tutiempo 2024, World Weather, Tutiempo 2024, viewed 7 June 2024, <https://en.tutiempo.net/>

Aman Kharwal 2023, *Air Quality Index Analysis Using Python*, The Clever Programmer, viewed 7 June 2024, <https://thecleverprogrammer.com/2023/09/18/air-quality-index-analysis-using-python/>

Air Now n.d., Air Quality Index (AQI) basics, viewed 7 June 2024, <https://www.airnow.gov/aqi/aqi-basics/#:~:text=Think%20of%20the%20AQI%20as,300%20represents%20hazardous%20air%20quality.>

Akshita Chugh 2020, *MAE, MSE, RMSE, Coefficient of Determination, Adjusted R Squared — Which Metric is Better?*, Medium, viewed 25 July 2024, <https://medium.com/analytics-vidhya/mae-mse-rmse-coefficient-of-determination-adjusted-r-squared-which-metric-is-better-cd0326a5697e>