TWINS OF WINDS-SOI 2024

Perfectly Imperfect

Aiswarya M S

Pallavi

Aabha

Contents

Problem Statement

12. **Task**
    - To develop a Machine Learning Model which can take Sequential Data and Generate the Sea Surface Temperature. Sea Surface Temperature is one of the important factors in prediction of El Niño.

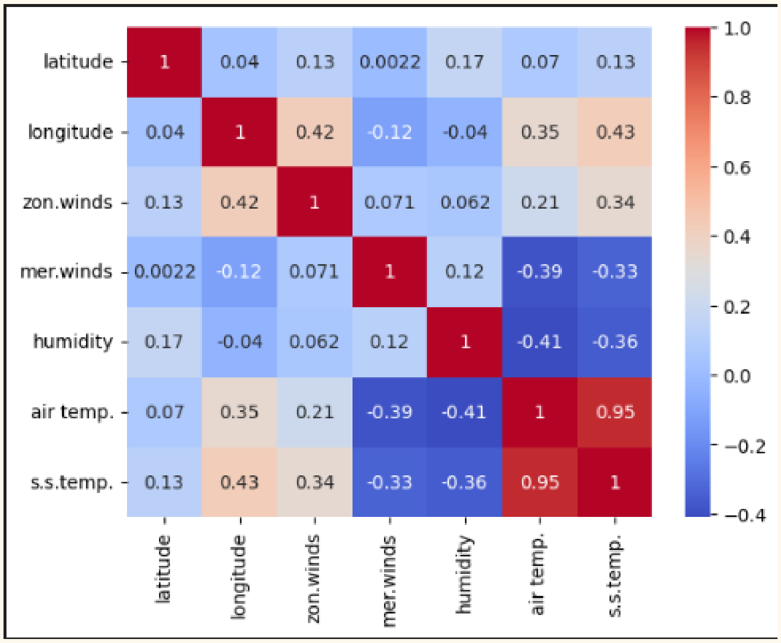
13. **Project Objective**
    - To train our Machine Learning Model on Labelled Data.
    - To use the ML model to predict the Unlabelled Data.

Data Collection and Preprocessing

1. **Data Source:**
   - As provided by the respective Team of SOI which includes train.csv, evaluation.csv and data_1997_1998 which contain latitude, longitude and various other parameters which determine sea surface temperature.
2. **Data Description:**
   - Data contains year, month , day, latitude, longitude, mer. winds,zon. winds, humidity, air temperature and sea surface temperature.
   - **Features:** Latitude, Longitude, Zonal Winds, Meridional Winds, Humidity, Air Temperature.
   - **Target Variable:** Sea Surface Temperature (s.s.temp.).
3. **Data Cleaning:**
   - Missing values i.e.... Nan values were filled by the mode of the respective column since mode showed better accuracy. When Nan values were filled with mode, model showed an accuracy of 96.60% while that of mean, it showed an accuracy of 96.48%
   - We have dropped unnecessary columns such as year, month, and day.
4. **Data Transformation:**
   - We have transformed the data by using Standard Scaler in Python.

Exploratory Data Analysis (EDA)

1. **Visualisation:**
   - We have visualized the data using heatmap.



Model Selection and Justification

1. **Model Selection:**
   - First we have chosen Polynomial regression. Then We have tried Neural Networks, and RandomTree Regression as suggested by our mentor and then we tried GB(Gradient Boosting) Model.
   - Finally We have chosen XGBM which has maximum accuracy.
2. **Model Architecture:**
   - We have imported xgboost(Extreme Gradient Boosting) which helped us to attain maximum accuracy of 96.60 when n_estimators is 1000 and 96.65 when n_estimators is 2000.

Data Preparation for Sequential Modeling

1. **Sequence Creation:**
   - Sequential Data was prepared by creating Time windows.
   - The dataset is split into training and testing sets (80%-20% split) using `train_test_split` from Scikit-learn.

Model Training

1. **Initialization:**
   - An XGBoost regressor (`XGBRegressor`) is initialized with hyperparameters (`n_estimators=1000`, `learning_rate=0.1`, `random_state=1`).
2. **Training:**
   - The XGBoost model is trained on the standardized training data (`x_train`) and corresponding target (`y_train`).

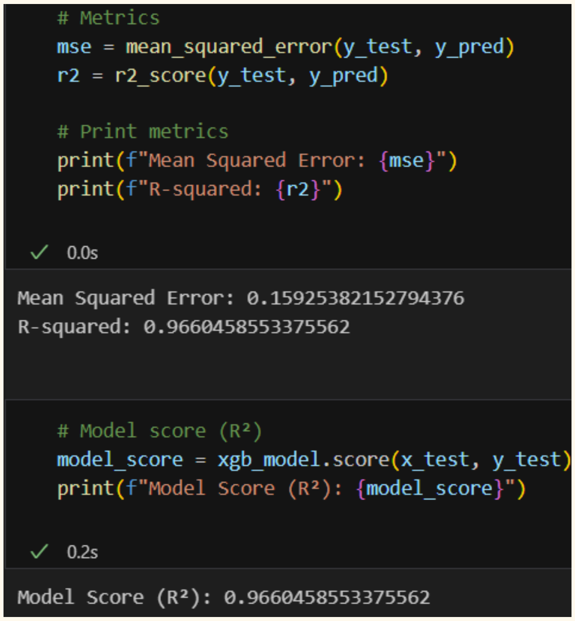
Model Evaluation

1. **Prediction:**
   - The trained model predicts SST values on the test dataset (`x_test`), and metrics such as Mean Squared Error (MSE), R-squared, and Model Score (R^2) are computed to evaluate model performance.
   - For n_estimators = 1000,

     MSE = 0.1592,

     R^2 = 0.9660

     Model_score = 0.9660



```
# Metrics
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

# Print metrics
print(f"Mean Squared Error: {mse}")
print(f"R-squared: {r2}")
```

✓ 0.0s

```
Mean Squared Error: 0.15925382152794376
R-squared: 0.9660458553375562
```

```
# Model score (R²)
model_score = xgb_model.score(x_test, y_test)
print(f"Model Score (R²): {model_score}")
```

✓ 0.2s

```
Model Score (R²): 0.9660458553375562
```

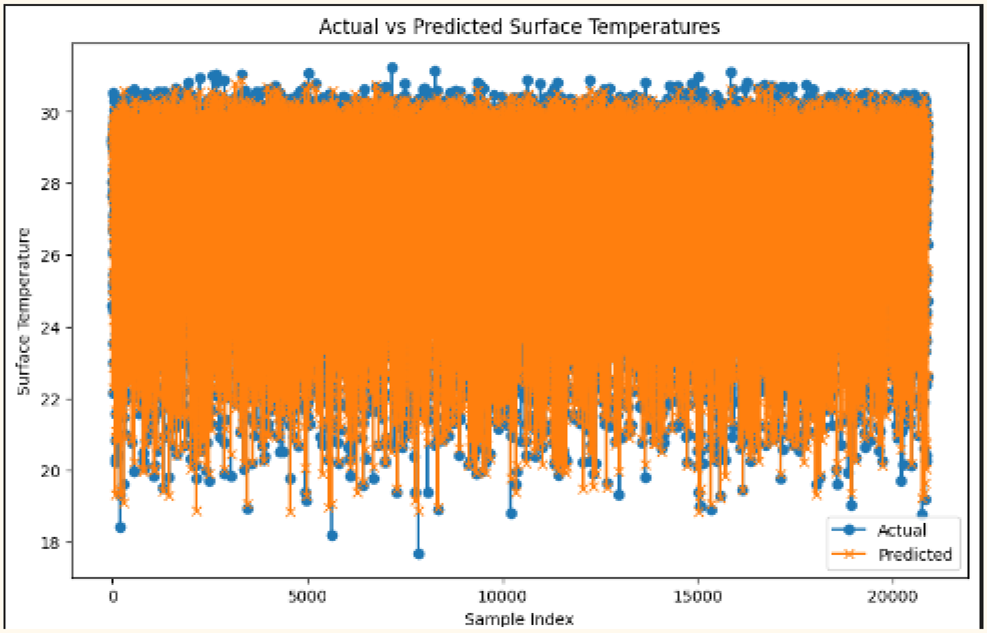
2. **Results:**
   - Predicted and actual SST values are compared and displayed in a data frame (`results_df`).

Prediction for New Data

1. **Prediction of given files:**
   - The trained model is used to predict SST for data_1997_1998.csv and evaluation. which is first scaled using the same `StandardScaler`.This predicted value of SST is separately stored in output1(data_1997_1998).csv and output2(evaluation).

Result Visualization

1. **Visualization:**
    - A plot compares actual vs. predicted SST values for visualization (for testing and training data).



Conclusion

The XGBoost model demonstrates effective prediction of sea surface temperatures based on selected meteorological parameters. It achieves a high R^2 score, indicating strong predictive performance.

References

References to data sources, and libraries used (Pandas, NumPy, Scikit-learn, XGBoost, Matplotlib, Seaborn).

THANKYOU