

# MODEL EVALUATION

The screenshot shows a Jupyter Notebook titled "Sprint2.ipynb" in a web browser. The notebook is in "Editing" mode. The code is organized into three cells:

- Cell [56]: Imports `sklearn` and `datasets`, and `sklearn.model_selection` and `train_test_split`. It splits the data into training and testing sets using `train_test_split(X, y, test_size=0.2, random_state=42)`.
- Cell [4]: Imports `sklearn.ensemble` and `RandomForestRegressor`. It creates a `RandomForestRegressor` object with `max_depth=2` and `random_state=42`, and fits it to the training data using `rf.fit(X_train, y_train)`.
- Cell [15]: Predicts the training and testing sets using `rf.predict(X_train)` and `rf.predict(X_test)`. It then imports `sklearn.metrics` and `mean_squared_error`, `r2_score`. It calculates the training and testing Mean Squared Error (MSE) and R-squared (R2) scores. Finally, it creates a `pd.DataFrame` with the results and transposes it.

The notebook interface includes a menu bar (File, Edit, View, Insert, Runtime, Tools, Help), a toolbar with icons for file operations, and a status bar at the bottom showing the current temperature (30°C) and weather (Haze).

The screenshot shows the same Jupyter Notebook interface, but with additional code cells and output:

- Cell [26]: Prints the Mean Absolute Error (MAE) using `print('MAE:', metrics.mean_absolute_error(y_test, y_rf_test_pred))`. The output is `MAE: 0.05078368352128359`.
- Cell [24]: Prints the Mean Squared Error (MSE) using `print('MSE:', metrics.mean_squared_error(y_test, y_rf_test_pred))`. The output is `MSE: 0.00709044752030462`.
- Cell [16]: Imports `numpy` as `np` and prints the Root Mean Squared Error (RMSE) using `print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, y_rf_test_pred)))`. The output is `RMSE: 0.08420479511467634`.

The notebook interface remains the same, with the status bar at the bottom showing the current temperature (30°C) and weather (Haze).

# ACCURACY OF THE MODEL

## ACCURACY OF THE MODEL

```
0s metrics.r2_score(y_test, y_rf_test_pred)
```

0.9898546855830299

# SAVE THE MODEL

The screenshot displays a Google Colaboratory notebook interface. The browser address bar shows the URL: `colab.research.google.com/drive/158FIN9UnPTH32i2vXrGzNbp_FkV_jvlp#scrollTo=QnYSfjXM1qAo`. The notebook is titled "Sprint2.ipynb" and has a menu bar with options: File, Edit, View, Insert, Runtime, Tools, Help, and a status "All changes saved".

The notebook content includes a code cell with the following Python code:

```
import joblib
from sklearn.ensemble import RandomForestRegressor
from sklearn.datasets import make_regression
X, y = make_regression(n_features=4, n_informative=2, random_state=0, shuffle=False)
rfr = RandomForestRegressor(max_depth=3)
rfr.fit(X, y)
print(rfr.predict([[0, 1, 0, 1]]))
joblib.dump(rfr, "my_random_forest.joblib")
loaded_rfr = joblib.load("my_random_forest.joblib")
```

Below the code cell, the output is displayed as `[34.83545299]`. The notebook interface also shows a left sidebar with file explorer icons and a bottom status bar indicating "completed at 12:08 PM". The Windows taskbar at the bottom shows the date as 15-11-2022 and time as 12:40.