Brain Stroke Prediction Using Machine Learning

-Anujkumar Yadav   MIP-ML-18

Project Overview and Objective

- **Objective:** To build a machine learning model to predict the likelihood of a patient experiencing a stroke based on demographic, health, and lifestyle attributes.
- **Overview:**
1. Motivation: Stroke is a critical health condition with significant mortality/morbidity rates. Early identification can enable proactive intervention.
2. Goal: Develop a predictive model for early identification of high-risk individuals to support healthcare decision-making.

Data Description

Dataset Features:

Total records and key attributes used in the prediction:

- gender: Male, Female
- age: Patient age
- hypertension: Presence (1) or absence (0)
- heart_disease: Presence (1) or absence (0)
- avg_glucose_level: Continuous
- bmi: Body Mass Index (with missing values imputed)
- smoking_status: Categories like formerly smoked, never smoked, smokes
- Target Variable: stroke (1 = Yes, 0 = No)

Data Preprocessing:

- Imputed missing values in BMI
- Encoded categorical variables (gender, ever_married, work_type, etc.)
- One-hot encoding for multi-class features
- Standardization of continuous features

Model Building and Evaluation

Model Used:

- Random Forest Classifier (tuned using GridSearchCV)

Evaluation Metrics:

- Accuracy
- Precision
- Recall
- F1 Score

Results:

- Hyperparameter tuning results and best parameters found
- Performance metrics on the test set:

```
in stroke/main.py
Best parameters: {'max_depth': None, 'min_samples_split': 2, 'n_estimators': 100}
Best F1 score: 0.010526315789473684
Accuracy: 0.949119373776908
Precision: 0.0
Recall: 0.0
F1 Score: 0.0
(base) PS E:\anuj\mentorness 2\machine learning\brain stroke>
```

Key Findings and Interpretability

- Feature Importance:

Feature Importance in Predicting Stroke
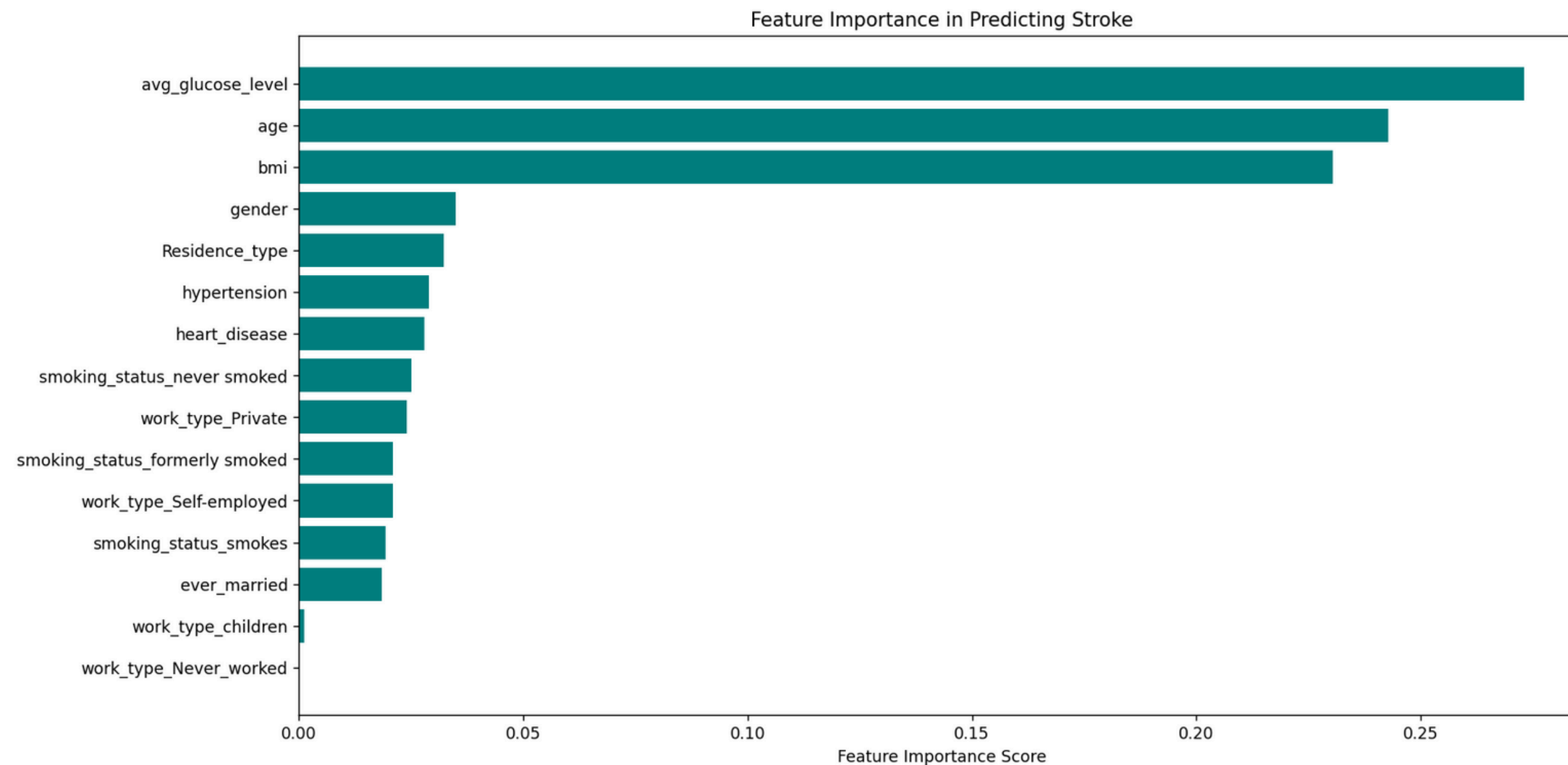
Conclusion and Next Steps

Conclusion:
- Successfully built a predictive model to identify stroke risk with evaluated performance metrics. This can aid healthcare practitioners in early intervention.

Next Steps:
- Improve model performance with more data and hyperparameter tuning.
- Consider incorporating more advanced models (e.g., boosting algorithms).
- Deploy the model for real-time prediction using a web interface or API.
- Further exploration of explainable AI methods to improve transparency.

Thank You