RAYCA

Evaluation Task: Designing a TPOT-2 AutoML Microservice

Resources:

TPOT2: https://github.com/EpistasisLab/tpot2 | http://automl.info/tpot/

Dataset: https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction

---

Objective

You are tasked with designing a microservice that employs TPOT-2, an autoML library, to generate a classification model based on specific input data. The microservice must allow users to specify the target and predictive features, and utilize TPOT-2 as AutoML engine to try out different configurations on model training; please implement required algorithms such as data normalization and feature selection that properly handle the input data and pass it through the rest of the ML pipeline, including training, model selection, and hyper-parameter tuning.

Microservice Development

1. Design and implement a microservice that utilizes TPOT-2 as the AutoML engine to try out different configurations on model training.
2. The service should allow users to input data and specify the target and predictive features.
3. Automate the ML pipeline including:
    a. Data preprocessing, feature selection, and model training.
    b. Hyper-parameter tuning and model selection using TPOT-2.

Data and Testing

1. Utilize the heart failure prediction dataset from Kaggle.
2. Test the microservice's functionality on this dataset, ensuring the user can input specific features for classification, and the microservice generates a predictive model.

Deliverables

1. The codebase hosted on a private GitHub repository. Please invite internship@raycabio.com for access.
2. Comprehensive documentation detailing the microservice design, setup instructions, and endpoints.
3. Reports highlighting the trained model's performance, including evaluation metrics, insights into model selection, hyper-parameter tuning, and scenario-specific analysis.

Evaluation Criteria

1. Functionality: Does the microservice execute feature selection, training, and model selection through TPOT-2 effectively?
2. Code Quality: Is the codebase well-organized, readable, and maintainable?
3. Documentation: Are the instructions clear for deployment and usage?
4. Model Performance: How well does the model perform on the provided heart failure dataset?

5. Insights and Reporting: Does the report provide meaningful insights into model selection and hyper-parameter tuning?

Submission:

Candidates are required to submit the source code (GitHub private Repository) for the microservice along with any necessary documentation, setup instructions, and API documentation to internship@raycabio.com

Confidentiality

It's imperative to maintain the confidentiality of this task; candidates must not include the results in their portfolios, use it for demonstrations, or share it for any other purposes, regardless of the outcome of their application.