Build an automated machine learning system to predict the risk of loan applications

Introduction

As a machine learning engineer in MoneyLion, you are responsible to architect and build machine learning systems and pipelines. Machine learning systems are important because they can free up data scientists from maintaining existing models manually. Automated pipelines are also useful for enforcing machine learning governance as all newly created models are forced to adhere to required standards and best practices.

Assignment

In MoneyLion, we have a LightGBM machine learning model in production that predicts the risk of loan applications. This model has been in production for the last 3 months and data scientists have been manually writing scripts to tune and improve the model. Now, we want to build an automated machine learning system so that this loan model can get continuous updates.

For this assignment, we want you to build the initial model and develop the automated machine learning system for future improvements. This challenge is intentionally meant to be open-ended.

The following input data file (*loan.csv)* has been provided to help you kickstart building the model and subsequently the machine learning system. The model should be reproducible and the final system should be able to run from end-to-end independently when the assessment is submitted.

To help you get started, we break down this assignment into the following three parts:

Part 1 - Build a LightGBM machine learning model to predict loan risk

Prior to building the automated system, you should start by building the machine learning model to predict risk of loan applications. You should build this model using Python.

- Use the data provided to create this model and subsequently the automated system.
- We expect your thought process and justification on how you build this machine learning model.
- Perform any analysis or visualisations on the dataset to help you explain your thought processes.

Part 2 - Write documentation on the plan & design you will be following to build this automated machine learning system

Plan out how you would design and automate the machine learning system in production.

- You should write documentation on your design and plan for this system.
- You should draw a diagram on how this pipeline should be built.
- Discuss any considerations performed when coming up with this plan and for each step.
- You may refer online for technical information but **DO NOT** directly use any help from other people, sources, online forums, etc. Your submission should be solely your ideas and work.
- This documentation should be easily understood by other data scientists/machine learning engineers
- This documentation should not be more than 2 pages long.

Part 3 - Code out the automated machine learning system using the LightGBM model built in Part 1

With the machine learning model built in Part 1 and the plan from Part 2, now build the automated machine learning system using Python.

- You should use Python to build this automated machine learning system for production.
- You can use and modify the data provided to simulate different scenarios if necessary
- Remember to put in some thought on how to structure your files and folders while assuming this pipeline will be placed in production!
- You may also perform a demo on how this machine learning system works during the interview.
- Describe in detail every step you take while you are coding to build this system.
- Your system should be able to run from end-to-end from data ingestion to model deployment.

Tips

The Machine Learning Engineer position at Moneylion is extremely competitive and we receive many applications, so consider how you could **make yourself stand out**. Here are some skills that we're looking for:

- Data assessment and understanding on building machine learning models

- Detailed plan and considerations on how to design this machine learning system
- Clean coding style and reproducible code
- Reasoning and justifications on what you develop
- Clear communication of your thought process
- Back your considerations and assumptions with evidence

There’s no expectation on the amount of time you could or should spend on the challenge. That said, do share how much time you spent on it.

Deliverables

A zip file with contents grouped into the following sub-directories (you may omit empty directories):

- Data (only if you add any new ones; DO NOT send back the original assessment data!)
- Notebooks (we appreciate it if you **include a html file of your notebook** as well as the raw file). Any code in the notebooks should be able to be reproduced and used directly.
- Other Python files and folders (if you use any custom Python files)
- Documentation about the application and pipeline. Do include workflow/architecture diagrams if applicable.