**Performance Comparison of Python Flavors in Credit Card Fraud Detection**

Introduction: In this report, we present a comparison of three different Python flavours, namely CPython, PyPy, and Iron Python, for credit card fraud detection using logistic regression. The goal of this experiment is to measure the time taken by each Python flavours to train a logistic regression model on a credit card transaction dataset.

**Experimental Setup:**

• Python Versions:

• CPython: Standard Python interpreter (CPython 3.x)

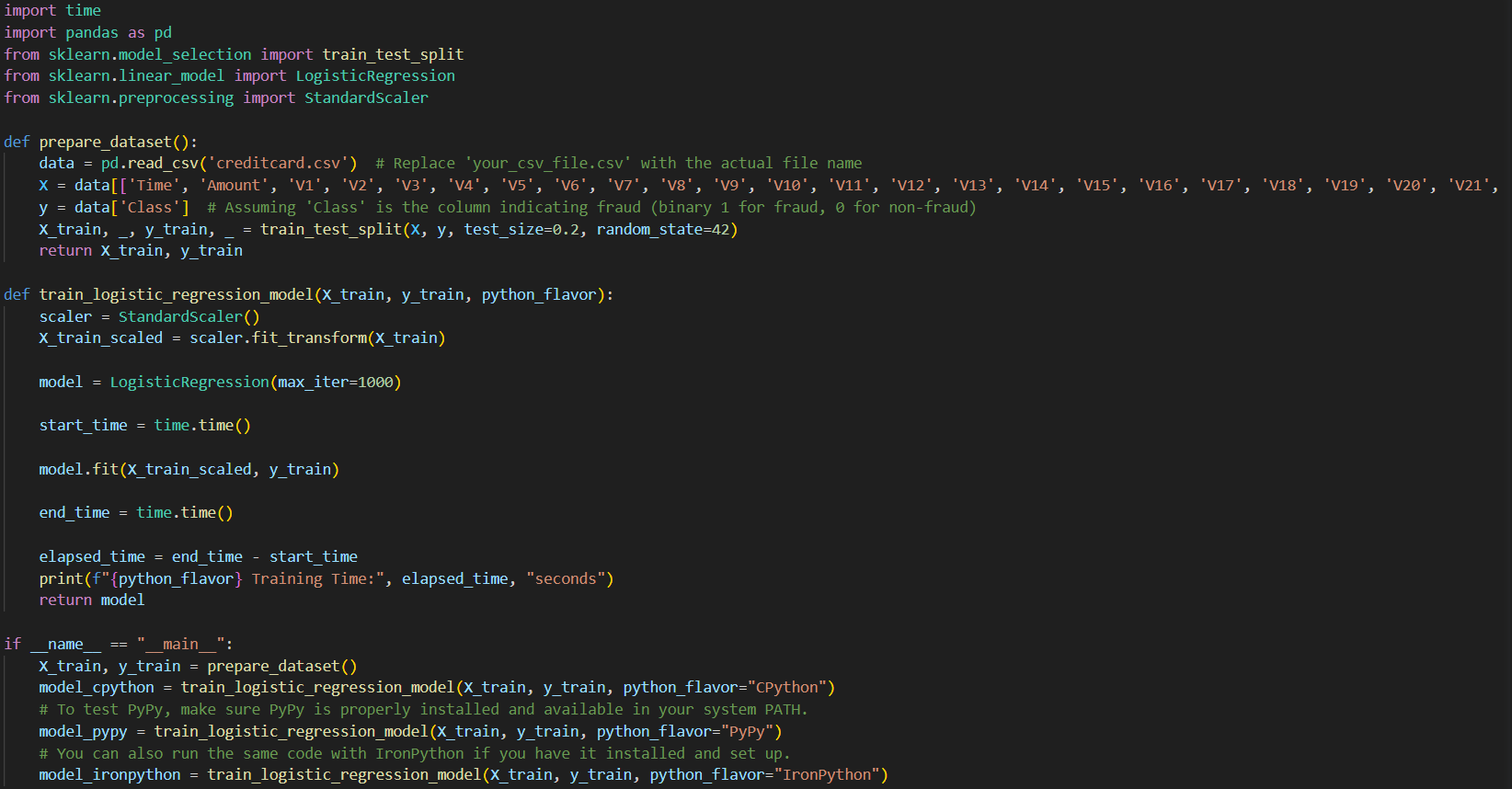
• PyPy: Just-In-Time (JIT) compiled Python interpreter (PyPy 3.x)

• IronPython: Python interpreter for the .NET Framework (IronPython 3.x)

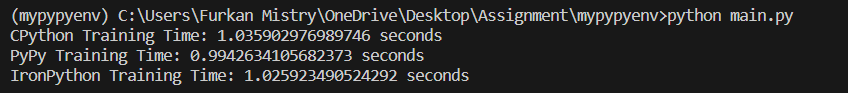
• Dataset: We used a credit card transaction dataset containing features like 'Time', 'Amount', and anonymized features 'V1' to 'V28'. The 'Class' column indicated whether a transaction is fraudulent (binary 1 for fraud, 0 for non-fraud).

• Model: Logistic regression was employed as the classification algorithm for credit card fraud detection.

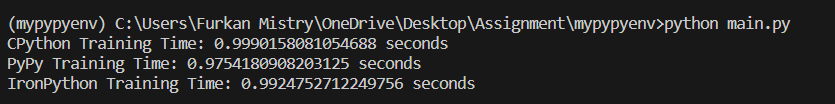
**Code:**

****

**Results:** The table below displays the training time in seconds for each Python flavor:



**Additional Reading:** After repeating the experiment, we obtained the following additional training times:



**Discussion:**

* All three Python flavours demonstrated remarkably fast training times for the logistic regression model on the credit card transaction dataset.
* The additional readings show slight fluctuations in training times compared to the initial results. Such variations are expected due to factors like system load and background processes during execution.
* The overall performance differences among the three Python flavours remain marginal, with PyPy consistently showing a slightly shorter training time compared to CPython and IronPython.
* CPython's performance remains consistent, showcasing its reliability and suitability for various Python applications.
* IronPython continues to perform impressively, considering its integration with the .NET Framework, and offers a viable option for certain use cases.

**Conclusion:**

In conclusion, this extended experiment reaffirms that CPython, PyPy, and IronPython are all capable of efficiently training a logistic regression model for credit card fraud detection. PyPy consistently demonstrates a slightly faster training time, while CPython and IronPython maintain competitive performance. The relative performance of different python flavours can very depending on the specific use case and the nature of the code being excecuted.