**Data science assessment - Fraud data (36 pts total)**

**Answer the questions only in Python and upload your code (Jupyter notebook or a \*.py file(s)) to a repo on github and share the link**

**High-level problem statement**

E-commerce websites often transact huge amounts of money. Whenever a huge amount of money is moved, there is a high risk of users performing fraudulent activities, e.g. using stolen credit cards, laundering money, etc.

**Objective**

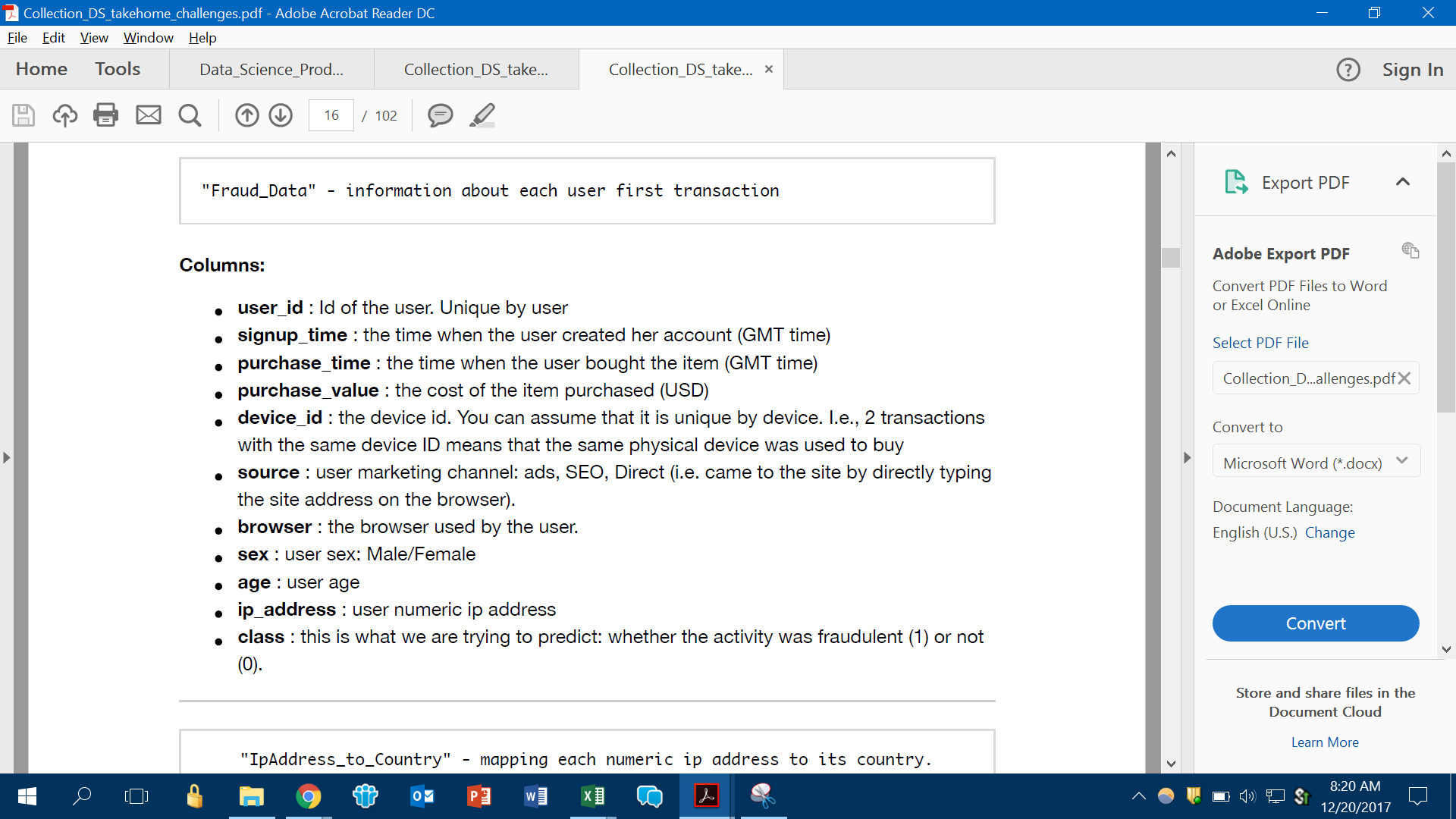
The goal of this challenge is to build a machine learning model that predicts the probability that the first transaction of a new user is fraudulent.

**Details**

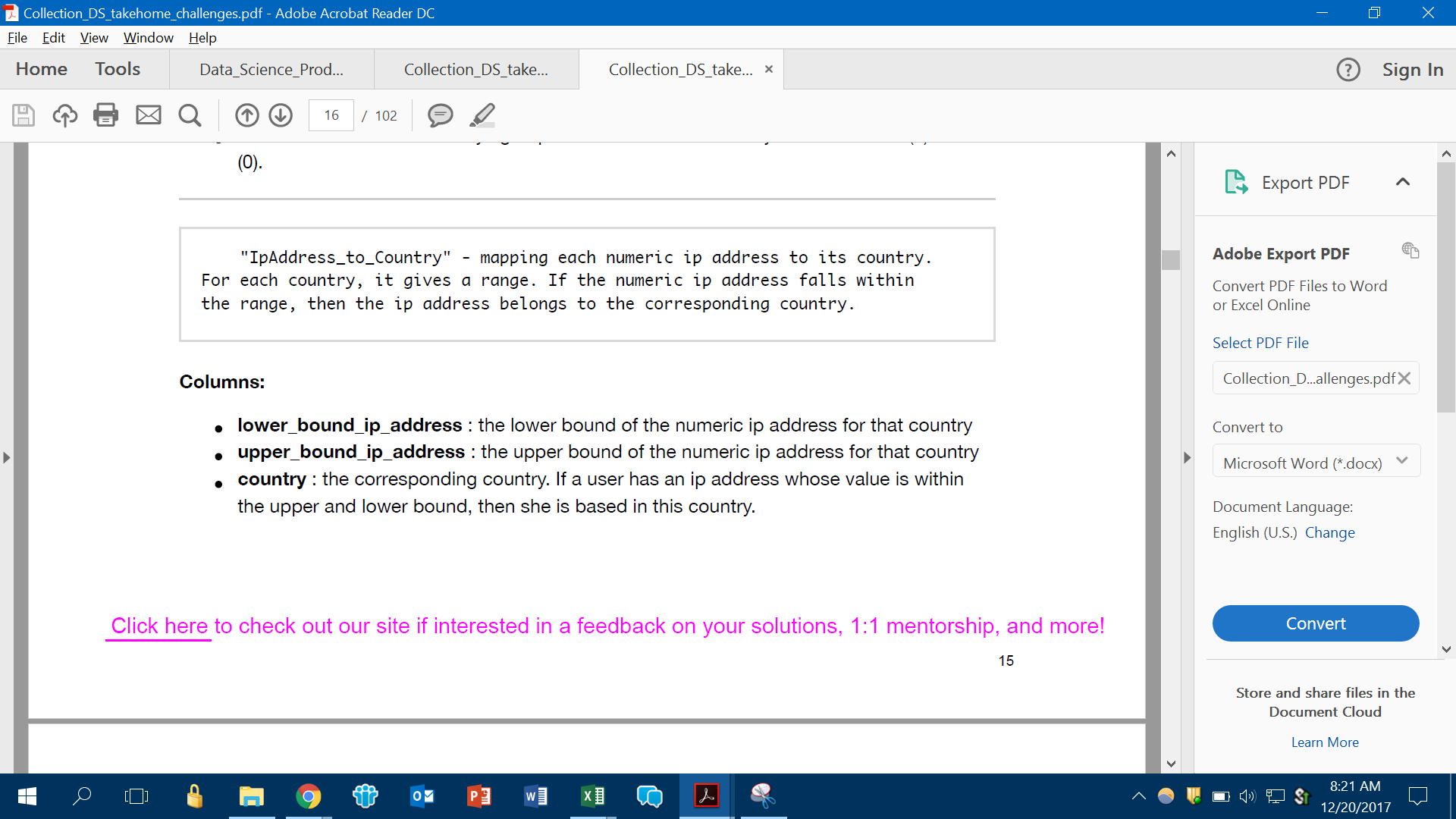
Electronica is an e-commerce site that sells wholesale electronics. You have been contracted to build a model that predicts whether a given transaction is fraudulent or not. You only have information about each user’s first transaction on Electronica’s website. If you fail to identify a fraudulent transaction, Electronica loses money equivalent to the price of the fraudulently purchased product. If you incorrectly flag a real transaction as fraudulent, it inconveniences the Electronica customers whose valid transactions are flagged—a cost your client values at $8.

|  |  |
| --- | --- |
| **24 points for showcasing your abilities** | **12 points for the performance of your model** |
| - Explore the data  - Keep all your work  - Comment your code  - Explain any features that you engineer  - Indicate which model is final  - Points will be given for originality / exploration of different model types  - Feel free to include “next steps” in the comments of your code  - You may be given partial credit for incomplete analyses | - We have withheld 20% of the dataset, with which we will evaluate the out-of-sample performance of your model.  - **Ensure you build a model MVP!** Once you have your MVP, then iterate to improve performance. You can’t get any credit for this section if you don’t have at least one model! |

**Scoring:**

****

**IP address to country mapping**

****