Practical Machine Learning Exercise

Amazon Development Centre Scotland

This is a short exercise to test your ability to construct and evaluate machine learning models. You should produce a (simple) machine learning model and an evaluation metric. The goal of this exercise is not to produce a state-of-the-art machine learning model. If your chosen model performs poorly by your selected metric, do not worry—this is not what we are testing. Which model you use, and how you evaluate, is up to you. The choice of model is not important (although we will assume that when you choose a model, you understand what it is and how it works). Your solution should be simple, but sensible: you should be able to explain why it tests something of impact to the problem.

1. Data

This exercise will use the UK government's land registry data. The goal of the exercise is to predict how much a house will sell for. This is similar to many problems Amazon has in online advertising and merchandising, where we try to predict what value a customer has for an item. You can find and download the data at:

https://bit.ly/2MzkQYW

A description of the columns in the data is available at:

http://bit.ly/1hh97JI

The site has many options: you should download the option labelled "the complete Price Paid Transaction Data as a CSV file", which contains all registered purchases since 1995 as comma separated values.

2. Instructions

Each row in the file contains a house that was purchased, the price that was paid, and features of the house and purchase. You should try to predict the price that was paid from (at most) three simple features: the lease duration, the property type, and whether or not the property is in London. In the data:

- Rows are separated by newlines
- Columns are separated by commas
- Column 1 contains a unique ID for the purchase and can be ignored
- Column 3 contains the date of the purchase
- Column 5 contains the property type (meanings of the codes can be found in the link above)
- Column 7 contains the lease duration
- Column 12 contains the town or city in which the property was located. You can judge a property to be in London if this field contains the word "London"

- You can ignore remaining columns

2.1 Train/Test split

Any purchases prior to January 1st 2019 should be used as training data. Purchases after this date should be used as test data. You can either do this split dynamically, or split the one large file into two files as a pre-processing step.

3. Tips and Clarifications

- We are not looking for a model that performs well: we are looking to see that you can build a sensible model with a sensible evaluation.
- You should aim to spend about 2 hours on the problem, and submit a solution withno more than a couple of hundred lines of code. Please do not spend more time than this, you will not receive extra credit for very elaborate or thorough solutions.
- If you are struggling to make something work with the volume of data present, you can subsample (for instance, look at a month or a year's worth of data). If you are struggling to implement something that deals with this volume of data, do you know of a way to deal with it in theory?
- If you are having trouble extracting features, can you submit an evaluation of a sensible baseline on the test data?
- You can use any programming language you like to solve the problem: pick a language suited to the task, and one you are comfortable with.
- You are strongly encouraged to make use of third-party libraries for model building and evaluation, rather than writing your own, unless you specifically need to do something with no library support.
- If you do not understand something, or have questions, please contact us!

4. Submitting Your Solution

Please email the code of your solution, and a single-paragraph summary of your model and evaluation result, to your recruiter. You do not need to include any of the data, or your model's predictions, with your submission. If you do any pre-processing to the data, please also include the script you use to do this (or a list of the commands run).