Homework 1
ECE 498 DS, Spring 2020

Issued: Feb 10, 2020
Due: **Feb 17, 2020 @ 11:59 PM on Compass2G**

What you will learn

1. Parsing and cleaning data (which is an important part of data analysis)
2. Basic Python

Instructions

For security and efficiency reasons, an operating system will divide up the available physical RAM into units called "pages". A page fault occurs on your computer when the memory address accessed by the CPU is not valid (i.e. has not been mapped to a page). In this assignment, you will parse & analyze a log of page faults reported by the operating system of a home desktop computer.

You need to modify the given Jupyter Notebook template to (i) convert the raw log file to a CSV file , (ii) sort/filter/slice the resulting dataframe and (iii) make some basic predictions.

Included Homework Files

Filename	**Function**
pf.log	Dataset
hw1.ipynb	Jupyter Notebook template
pf_example.csv	Example CSV file (partially complete)
HW1_template.pptx	Template for HW answers

Log Format

The raw page fault entries in ***pf.log*** are formatted as follows:

```
<timestamp>:<process>:<pid>:<address>:<read/write>:<major/mino
r>:<time to resolve>
     <library+offset/function addr>
     <library+offset/function addr>
     <library+offset/function addr>
```

where

- `<timestamp>` indicates the Unix time when the page fault occurred
- `<process>` indicates the name of the process causing the page fault
- `<pid>` indicates the process ID of the process causing the page fault. Note that each process may be spawned multiple times, resulting in multiple PIDs…
- `<address>` indicates the address (in hex) where the page fault occurred

- `<read/write>` indicates whether the page fault was caused by a read or write access
- `<major/minor>` indicates whether the page fault was major or minor
- `<time to resolve>` indicates the amount of time (in milliseconds) the operating system took to resolve the page fault

Additionally, a backtrace is provided for each page fault with one or more entries:

- `<library>` indicates the name and version of the library in the trace entry
- `<offset>` indicates the offset of instruction within the function causing the page fault
- `<function addr>` indicates the name of the process causing the page fault

Dataframe Format

You are required to convert ***pf.log*** to an intermediary CSV (Comma Separated Values) file with the following headers:

1. ***index*** – an auto incremental value unique for each page fault
2. ***time*** - timestamp of page fault
3. ***proc_name*** - name of process causing page fault
4. ***pid*** - process ID of process causing page fault
5. ***pfaddr*** - page fault address (converted to int)
6. ***rw*** – read/write access
7. ***major_minor*** – major/minor access
8. ***resolve_time*** – time in milliseconds to resolve page fault
9. ***lib*** – full path of the library causing the page fault
10. ***addr*** – address of function within backtrace (converted int)
11. ***offset*** – offset (within function) of instruction causing page fault (converted to int)

For page faults with multiple backtrace entries, write out one line per trace entry, while keeping items 1-8 constant. Values should be separated by tabs ('\t') and lines should be terminated with a single new line delimiter ('\n') [Hint: consider the *`sep`* parameter in Pandas' *`to_csv`* function…]. Save your file as ***pf.csv***.

An example of the required CSV file is provided in ***pf_example.csv***. It contains entries for the first two page faults in ***pf.log***.

Making sense of your dataframe

Import ***pf.csv*** to a pandas dataframe using the `read_csv` function from the Pandas library. Remember to convert timestamps to Pandas datetimes and to set the dataframe index appropriately.

Before you begin answering questions below, you may find it useful to play around with your dataframe using functions covered in this tutorial: http://pandas.pydata.org/pandas-docs/stable/tutorials.html.

Questions

Answer the following questions based on your analysis above. Format your answers in a slide show presentation with bullet points. We have provided you with a template for submitting your answers. Remember to include a title and axes labels for each plot!

A. **Show the data structure that you used to parse in the raw log file in terms of python dictionaries, lists, sets, etc.**

B. **Data analysis**
   a. What time range does this data cover?
   b. How many unique processes were executed over this period? How many times was each process executed?
   c. Compare the number of major and minor page faults for each process. Plot a bar chart with two categories (major and minor) to demonstrate your results.
   d. Plot the histogram for the **time to resolve** page faults. Label the axes. For each process, report the mean and standard deviation of the time to resolve page faults. Plotting and calculations should be done for major and minor page faults separately.

C. **Making Predictions**

Suppose that you want to be able to predict which process is the most likely cause of a page fault given certain information about the fault.

   a. Assume that the process variable is the class. Calculate the priors for all the classes.
   b. Given that the page fault was major, which process was it most likely caused by? Use the MAP rule.
   c. Given that the page fault came from a read access, which process was it most likely caused by? Use the MAP rule.
   d. What model taught in class could you use for classifying the process given information about (i) the severity of the fault (major/minor) and (ii) the access type of the fault (read/write)? Write your answer in two sentences or less…

What to turn in

Submit (i) your ipynb notebook and (ii) the PDF of your slides via Compass2G. You do not need to submit your CSV or log file.