Submission: 22-Jun, Monday 4:20pm

**Retail Incident Prediction**

**Challenge Overview**

Welcome to the Retail Incident Prediction data science challenge!  
  
Our customer is a retail chain interested in leveraging data science and machine learning techniques to forecast maintenance incidents in their stores based on available historic data.  
   
We will focus on six of the most frequent incident types, labeled as cash\_drawer, pin\_pad, register, coffee, frozen\_drink, and soda\_fountain. As these labels suggest, the first three denote different malfunctions of the store counter equipment, and the latter three refer to malfunctions of coffee, frozen-drinks, and soda vending machines installed in the stores.  
  
The following data are available for the analysis:

* The historic incident log, which includes store location IDs, incident types, timestamps, durations, and other details.
* The log of related transactions with a per-store breakdown. It includes coffee, soda, and frozen drink purchases in vending machines, and overall sales in each store.

Challenge Objective

* Your goal is to come up with an approach able to predict future incident counts and average durations based on the historic data.
* You will provide both a code implementation of your solution, and a write-up explaining it, and validating its performance.

There is a twist. We are working with real data from real stores. During our pre-launch testing, we determined it was possible (but not confirmed) that there may not be enough data for machine learning alone. The aim of this challenge is not to simply perform the best fitting with ML techniques, the task is to develop as strong a prediction logic as possible. This should involve ML and statistics but can also include heuristics, additional data, etc, as well. Therefore we are hosting this marathon as a research problem, with a novel prize structure as described below.

**COMPETITOR PACK DETAILS**

The [Competitor Pack](https://apps.topcoder.com/forums/?module=Thread&threadID=957406) provided in the challenge forum (available after the registration to the challenge), includes the following content:

* /data/all-data - Entire dataset available for the analysis, training, and benchmarking of your solutions. It consists of:  
  + ���/data/all-data/sales - Sale transactions with per-store breakdown, covering the time range July 1, 2019 - March 31, 2020. Provided in the form of CSV files, each containing one month worth of data.
  + /data/all-data/aggregated\_sales.csv - Aggregated sale transactions for the time range July 1, 2019 - March 31, 2020. These are generated from data/all-data/sales files by summing up daily transaction counts over different stores. The reason to have this file is explained in the scoring section below.
  + /data/all-data/incidents.csv - The full incident log covering the time range July 1, 2019 - March 7, 2020.
  + /data/all-data/locations.txt - The list of all location IDs in the dataset.
* /data/inputs - A subset of data to use as inputs for benchmarking. It covers July 1, 2019 - December 31, 2019, and the assumed goal of your solution is knowing only these data to forecast the incident counts and durations for the period January 1, 2020 - March 7, 2020. See further explanations in the scoring section below.
* /data/ground-truth/gt.csv - Incident counts and average durations (where known) for the period January 1, 2020 - March 7, 2020, with per-store and per-incident breakdown. This data will be used as the ground truth for benchmarking.
* /scorer - NodeJS scoring code which will be used to benchmark your solutions. It also includes a setup for running with Docker.

**SCORING DETAILS**

The Top-3 placements in this challenge will be awarded based on the objective performance of your solutions in the benchmark explained below, assuming that your submission includes a reasonable write-up explaining your approach, and does not attempt to cheat the benchmark rules.  
  
The 4-6 placements will be awarded subjectively, to other competitors, based both on the results and on the write-ups.  
  
Also, trivial solutions won’t be eligible for any placement; *e.g.* forecasting no future incidents at all receives relatively high score in our benchmark, because it is not a bad guess, the incidents are relatively rare.   
  
For the benchmark purposes we assume we are at December 31, 2019, and we want to forecast future incident counts and average durations for the period January 1, 2020 - March 7, 2020. Being on December 31, 2019 we only know the subset of data prior to this date, provided in the /data/inputs folder of Competitor Pack. These data include: the log of past incidents, and detailed transaction data for July 1, 2019 - December 31, 2019. It also includes the aggregated transaction data /data/inputs/aggregated\_sales.csv for the larger period July 1, 2019 - March 31, 2020. The reason for this is the following. You will see from the dataset that transactions of interest are clearly season-dependent (*e.g.* frozen drinks are consumed a lot more in summer compared to winter time). It is a problem as the entire dataset covers less than one year of data, thus it is impossible to predict seasonal trends. As a workaround we assume that a forecast of future transaction trends is readily available in /data/inputs/aggregated\_sales.csv. As mentioned before, we generated this file from the actual transaction data for January 1, 2020 - March 31, 2020 by summing up the counts over different stores.  
  
For these inputs your solution will generate a CSV files with forecasted incident counts and average durations for January 1, 2020 - March 7, 2020, which will follow this format:  
  
location\_id,incident\_type,count,av\_duration\_hours  
00186317dbb69b00acffd2984b96194e,cash\_drawer,0,  
00186317dbb69b00acffd2984b96194e,coffee,1,  
00186317dbb69b00acffd2984b96194e,frozen\_drink,2,3.75  
00186317dbb69b00acffd2984b96194e,pin\_pad,0,  
00186317dbb69b00acffd2984b96194e,register,0,  
00186317dbb69b00acffd2984b96194e,soda\_fountain,0,  
002046bedb961740983264904b961947,cash\_drawer,0,  
002046bedb961740983264904b961947,coffee,0,  
002046bedb961740983264904b961947,frozen\_drink,0,  
002046bedb961740983264904b961947,pin\_pad,0,  
002046bedb961740983264904b961947,register,1,  
002046bedb961740983264904b961947,soda\_fountain,0,  
0038bdf4dbe69b40983264904b96192f,cash\_drawer,0,  
  
Important:

* For some incidents, durations were not recorded into the client database; for them you will find empty values inside the incident\_duration\_hours column of incidents.csv file. If you output an empty value or 0 in the last (av\_duration\_hours column) of your solution, it is treated as the average duration cannot be forecasted for this incident type and store.
* Your output must have a record for each expected store and incident type, and to not have any records for unknown stores and incidents.

The scorer, provided in the Competitor Pack, will score your outputs against the ground truth using the following approach:  
  
Let us define the following parameters for a selected store location and incident type:

* *nF* - the forecasted number of incidents of selected type in the selected store. It can be a fractional, non-negative number.
* *nG* - the observed number of incidents of selected type in the selected store. It is an integer number.
* *tF* - the forecasted average duration of the incident. It can be a fractional, non-negative number, expressed in hours; it also can be an empty value.
* *tG* - the observed average duration of the incidents of the selected type in the selected store location (in hours). It can be unknown for some incidents and locations.

The score *SST* of your prediction for this selected store & incident type is calculated the following way:

* If *nF* < 0 or *tF* < 0 The forecast is invalid by definition and the entire solution is failed.
* *SST*= 0.8 *SS*+ 0.2 *ST* otherwise, where the two terms are calculated as:
  + 
  + *ST* = 1.0 if *tG* is unknown for this incident & store.
  + 
* The overall score of your solution will be calculated as the average of *SST* over all known stores and incident types.

Note that the trivial forecast, no incidents and unknown durations for all incident type and store pairs scores quite high in this benchmark and might be not that easy to outperform. This is expected as while there are many incidents happening in the entire retail chain, for specific stores the incidents are relatively rare.

**Final Submission Guidelines**

Submit a ZIP file containing:

* Your solution code along with detailed usage instructions. Please [Dockerize](https://docs.docker.com/engine/) your solution to facilitate the verification, and further use and development of the code (if you have any problems with Dockerization, don’t hesitate to ask for advices in the challenge forum). Should your solution rely on machine learning we need instructions on the model retraining, as we’ll need to re-train your model to verify that it is trained on the input data only, without access to the ground truth data subset.
* PDF write-up (research article) explaining your approach and solution. It may include additional details of your data analysis. For example, if your solution does not perform great on our benchmark, but you believe it is still providing forecast value, which is not captured by the benchmark, you are welcome to analyse it. We’ll consider this, especially for the subjective 4-6 placements of this challenge. If you do use a different ranking metric than the one provided, be sure to explain illustrate your reasoning in close detail.  
    
  The write-up should consist of the following sections:
  + Overview: describe your approach in “laymen’s terms”
  + Methods: describe what you did to come up with this approach, eg literature search, experimental testing, *etc.*
  + Materials: did your approach use a specific technology? Any libraries? List all tools and libraries you used
  + Discussion: Explain what you attempted, considered or reviewed that worked, and especially those that didn’t work or that you rejected.  For any that didn’t work, or were rejected, briefly include your explanation for the reasons (*e.g.* such-and-such needs more data than we have). If you are pointing to somebody else’s work (eg you’re citing a well known implementation or literature), describe in detail how that work relates to this work, and what would have to be modified
  + Data: What other data should one consider?  Is it in the public domain?  Is it derived?  Is it necessary in order to achieve the aims?  Also, what about the data described/provided - is it enough?  
    Assumptions and Risks: what are the main risks of this approach, and what are the assumptions you/the model is/are making?  What are the pitfalls of the data set and approach?
  + Results: Did you implement your approach?  How’d it perform?  If you’re not providing an implementation, use this section to explain the EXPECTED results.
  + Other: Discuss any other issues or attributes that don’t fit neatly above that you’d also like to include
* Editable sources of your PDF write-up.