Problem Statement

1. Background

Every day financial institutions consume large amount of historical data for financial analysis and risk management. The data sources are myriad, and often provided by third parties. One of the challenges faced by financial institutions is data anomaly, and the considerable effort spent in cleaning data. Here we share a typical problem where we have a dataset with some anomalies, with a degree of (not necessarily complete) data cleaning already done. We call for innovative models to identify anomalies in an uncleaned dataset. The solution you provide should work not only on the data shared, but also on an ongoing basis on a general dataset.

2. Introduction and Problem Statement

The given dataset contains quarter end data for 76 variables categorized into 8 Asset Classes (Equity, FX, Real Estate etc.) from 2007Q1 to 2018Q4. Certain data corruption events have created anomalies in this dataset. Some of the anomalies detected include, but not limited to, *missing values*, *outliers* and *change of unit (for example 85bps reported as 0.85bps).* In the dataset we have corrected all anomalous data points for the period 2013Q1-2015Q4.

1. The problem is to come up with a model/methodology which can be programmed / coded and detects all types of anomalies, including the ones mentioned above, that can potentially corrupt a dataset.
2. We want the algorithm to provide probabilities of a given value to be an anomaly. The probabilities can be specified as **classification problem (0/1 flag)** or could be a **computed probability**. We would accept both forms of solutions.
3. Contestants are allowed to use any programming platform as long as the source code is provided along with the submission.
4. Contestants should keep in mind to avoid overfitting the algorithm to sample test data provided. In other words, the algorithm should work well on validation data set as well (not disclosed as part of this problem).

3. Input Data

The description of the data in the attached csv file is as follows:

Column 1	Contains details of the quarterly timeline of data
Last Column	A flag that indicates if the data contains anomalies. CHECKED indicates that the data has been checked and that there is no anomaly in the data for that particular row. NOTCHECKED implies that the data has not been checked there may or may not be an anomaly in any of the variables during that quarter.
Row 1	Contains the asset class
Row 2	Contains the name of each variable
Row 3	A flag that indicates if the data contains anomalies. CHECKED indicates that the data has been checked and that there is no anomaly in the data for corresponding variable. NOTCHECKED implies that the data has not been checked there may or may not be an anomaly in any of the quarters for that variable. This flag is applicable to data points at the variable level.
Row 4-51	Contains data points for each variable corresponding to the given quarter in Column 1

	AC1	AC1	AC1	AC1	AC2	AC2	Status
	V11	V12	V13	V14	V21	V22	
	CHECKED	NOTCHECKED	NOTCHECKED	NOTCHECKED	NOTCHECKED	CHECKED	
2007Q1	0	13.27266775	83.51927346	45.18211193	60.60621062		NOTCHECKED
2007Q2	50.32835421	13.70141772	95.79003277	41.48823608	57.58440182	19.15583357	CHECKED
2007Q3	41.73649004	123456	80.70383625	43.61355634	66.75907421	9.926658282	NOTCHECKED

a) Assumptions you can make

- You can assume that for any given asset class not all variables are anomalous for a given quarter

- Actual production data at our end consists of many more variables for a much more granular timeframe than the training data that has been shared, and the code/logic submitted should be robust and scalable enough to run on a much larger dataset.

4. Output Data

a) The output should be a csv file with the same structure as the input data file. You have to overwrite the cells containing the time series data (rows 4 to 51) with probabilities of data error/ anomaly in the respective cells (or, a 1/ 0 flag for data anomaly). Do not delete, add, or reorder any rows or columns.

A sample of the output file is as follows for reference:

	AC1	AC1	AC1	AC1	AC2	AC2	Status
	V11	V12	V13	V14	V21	V22	
	CHECKED	NOTCHECKED	NOTCHECKED	NOTCHECKED	NOTCHECKED	CHECKED	
2007Q1	0.78	0.08	0.19	0.03	0.02	1.00	NOTCHECKED
2007Q2	0.00	0.00	0.00	0.00	0.00	0.00	CHECKED
2007Q3	0.18	1.00	0.11	0.07	0.15	0.39	NOTCHECKED

- In this example, the probability that the 2007Q1 data of VAR11 is anomalous is 0.78, the probability that the 2007Q2 data of VAR11 is anomalous is 0.00 and so on
- The output file having only binary outputs i.e. 1 or 0 would also be accepted

4. Required Submissions

4.1 Each team needs to submit 3 files as part of their Round 1 evaluation.

- A csv file with the output for the training data provided along with the problem statement
- The underlying code (with clear comments) which can consume a .csv file in the same format as explained above and produces the output in the required format. This will be used to run the algorithm on multiple sets of test data. The code can leverage any programming platform which the contestants are comfortable with
- An executive summary in word document (not exceeding 2 pages) explaining the algorithm used in plain language. It should also mention the coding language used.

4.2 Each shortlisted team needs present their approach in the second round. Further details of the presentation will be shared with the successful teams shortlisted from Round 1

4.3 The exact team name (without any underscores or special characters) should be the name of the output files submitted to us. For instance, it the team name is “Three Mavericks”, the files submitted to us should be “Three Mavericks.csv”, “Three Mavericks.doc”, etc.

5. Success Criteria

- Round 1 submissions will be assessed on the basis of:
    1. accuracy of anomaly detection on training data and test data;
    2. clarity and elegance of underlying code
    3. clarity of submitted executive summary

6. Useful Supplementary Information

- There may be other anomalies too apart from the above mentioned ones (Missing values, outliers and change of unit).

- The model should work even if the column Status contains only NOTCHECKED values.

- The model should work even if the Test Data contains only NOTCHECKED variables within an asset class.

- Method of evaluating the performance of a model will be discretionary to Credit Suisse.