**Outlier Detection**

This project is developed to automate the process of detecting outliers in agricultural data. The data has been collected from AgMarknet site. It's an enormous dataset spread across 7 years. Each year dataset contains 12 columns and approximately 30 lakh rows. Therefore, the code has been time and space optimised. Code has been written in Jupyter notebook.

**Method 1:**

The data is filtered using loops and lists; we filter using Commodity type and Market Centre. We define a band for normal data points using the Coefficient of Variation and the difference between the modal price of each data point from the moving average.

If the difference between the modal price and its corresponding moving average is found to be greater than the coefficient of variation, then the data point is marked an outlier.

**Method 2:**

Once again, the data is filtered using loops and lists; we filter using Commodity type, Market Centre and the State. For each filtered set we calculate the 7-window centred moving average, the absolute of the difference between the moving average and the modal price, the standard deviation and the mean.

We define a range such that:

Lower Band: Mean - (Standard Deviation) \* K

Upper Band: Mean + (Standard Deviation) \* K

Where K = 1, 2 or 3

If the difference lies outside the range where k=1 then flag1 is marked 1 to represent an outlier. Otherwise, flag1 is marked 0 to represent a normal data point. Similarly, we calculate flag2 and flag3.

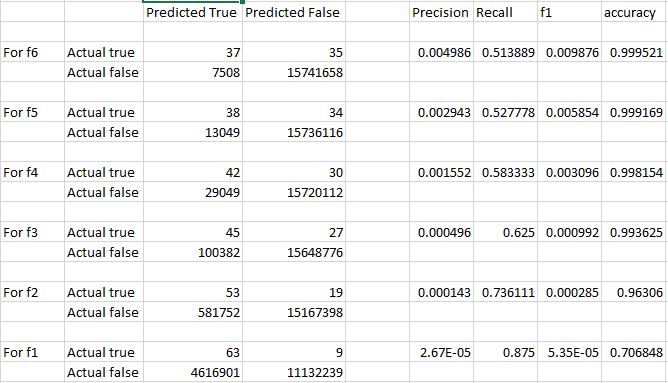
**Method 3:**

The data is grouped according to its Commodity type, State and Market Centre. Within these groups we have created 6 brackets of outliers that vary by a multiple, ‘K’, of the standard deviation. The groups that have a single data point are not evaluated because the standard deviation is meaningless in these cases. Methodology (The general formula):

Mean of Modal Prices - (K \* Standard Deviation of Modal Prices) < Modal Price < Mean of Modal Prices + (K \* Standard Deviation of Modal Prices)

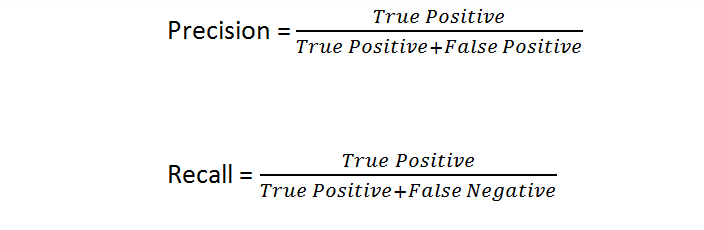
We have processed the data for flags for K= 1, 2, …, 6 The range defined as such represents all data points within K standard deviations. Any data points that lie outside this range are ‘flagged’; where 1 represents an outlier and 0 represents a normal data point.

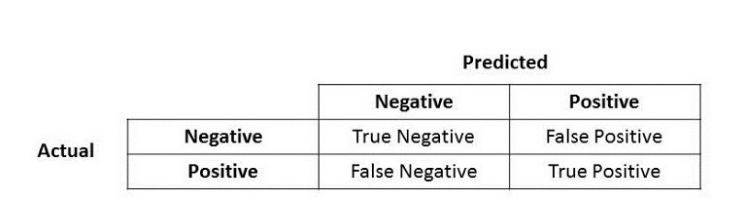
**The results of the confusion matrix for all flags along with their precision, recall, f1 score and accuracy:**



**Note:** f1, f2, …, f6 represent flag1, flag2, …, flag6.

**Accuracy**: It is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.





**Precision** talks about how precise/accurate your model is out of those predicted positive, how many of them are actual positive.

Precision is a good measure to determine, when the costs of False Positive is high. For instance, email spam detection. In email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam).

**Recall** calculates how many of the Actual Positives our model capture through labeling it as Positive (True Positive). Applying the same understanding, we know that Recall shall be the model metric we use to select our best model when there is a high cost associated with False Negative.

For instance, in fraud detection. If a fraudulent transaction (Actual Positive) is predicted as non-fraudulent (Predicted Negative), the consequence can be very bad for the bank.