A comparative study of correlation measures for the application of anomaly detection in Data Streams

Table of Contents

Certificate

This is to certify that the thesis titled "**A comparative study of correlation measures for the application of anomaly detection in Data Streams**" being submitted by Sagar Khatri to the Indraprastha Institute of Information Technology, Delhi for the award of Master of Technology, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other University or institute for the award of any degree/ diploma.

Acknowledgements

I take this opportunity to express my profound gratitude and deep regards to my supervisor Dr. Vikram Goyal for giving me the opportunity to work on this project. The door to his office was always open whenever I had a doubt regarding my research or writing. He constantly provided with mentorship and steered me in the right direction Whenever I needed it. I would not have been able to complete my thesis work this smoothly in IIIT Delhi without his handholding and consistent support.

Abstract

We study two measures namely maximal information coefficient (MIC) and distance correlation (dCor) for the application anomaly detection. MIC is based on the concept of information theory and measures the mutual information between two variables. dCor uses distances between observations for its calculation. Both of these measures are considered better than the classical Pearson Correlation coefficient that has been the main measure for many decades. The main reported issue with the Pearson correlation measure is to not guaranteeing independence between variables even in the case of coefficient value being zero. Furthermore, it is very sensitive to outliers and assumes continuous normally distributed data for its working. We experimentally study the efficacy of MIC and dCor measures over energy data for different experimental settings, i.e. windows size, overlapping threshold and anomaly threshold. We observe that MIC consistently outperforms dCor.

List of Figures

Chapter 1

Introduction

Anomaly detection is a process of discovering relationships in data that do not match to normal behavior in data. These non-confirming relationships in data are often represented by terms such as anomalies, outliers, exceptions or aberrations in different application domains. If a anomaly arises in a dataset, this show that relationship in data has changed with time. If relationship in dataset starts to change, this scenario represents dependency between variables in dataset changing. Correlation measures would help us to depict that change in dependency between variables in datasets. This would help us to detect the changing relationship in dataset over a period of time. Changing relationship in dataset shows that an anomaly may or may not be occurring in dataset. Reasons for using these two specific measures are explained further.

Due to the advancement of technology nowadays, an enormous amount of data is being stored. Different data mining techniques have been used increasingly to match the needs of modern times and to analyze relationships among vast numbers of random variables. Linear relationships can be easily analyzed but Non-Linear relationships are not so easily analyzed. Correlation measures used in our study is a way of solving the above problem.

Correlation Measures:

Correlation measures studied were from the maximal information-based nonparametric exploration (MINE) family. Measures which come under

MINE family can be used not only to identify important relationships in data sets but also to characterize them.

MIC (maximal information coefficient) and dCor (distance correlation) are the two measures, used in our study, comes under the MINE family. A standard implementation of MIC and dCor in R-studio were used in our study.

Packages for MIC and dCor:

- MIC - MINE package: This package can be installed in R-studio and can be used to find out MIC value between two random variables. It takes a data frame as input, with the input variable in two different columns. It returns a list of values. The first list in the result consists of a matrix which shows dependency between each variable in the data frame.
- dCor - Energy Package. This package can be installed in R-studio and can be used to find out dCor value between two random variables. It takes two vectors of random variables as input and returns the dCor value of dependency between these two variables directly.

We use these correlation measures along with their implementation (using the R packages) in our experiments.

Data Set:

We use energy data in our experiments. This data consists of the Current and Power readings. This is a time-series data recorded with the interval of 30 seconds each. It was collected over four days continuously. At first look of the data, one could find a lot of change points (i.e. a sudden change in values in the data).

After the data was recorded, it was labeled for anomalies. During the labeling process, one extra column was included in the dataset titles '*Anomaly*'. This column added in the dataset defines whether the current data point (row) is an anomaly or not. The anomalous data point (row) is marked using value 1 in the Anomaly column(which is added at point of

labeling dataset). A non anomalous data point is marked using value 0 in the Anomaly column.

Labeling of the dataset has been done in multiple ways to create multiple datasets. After creating multiple datasets, these are used to run experiments separately. MIC and dCor measures both have been applied on both the datasets separately.

Simulating Streaming Environment:

Simulations for the streaming environment were done by using the data and accessing it in form of a window. Windows in streaming data environment were defined in a couple of ways as defined below:

1. **Time Interval**: In this scenario, a window was defined on the basis of time. A time period was decided and data values received in that time interval contributed to a window. For example, if time interval is 5 minutes then 10 data points(i.e. as we have recorded a data point in 30 seconds interval) in our dataset would contribute to a single window.
2. **The number of Data Points:** In this scenario, a window was defined on the basis of total data points in a window. For example, if the window size is 100 then in the streaming environment as soon as we receive 100 data points it would contribute to a particular window.

The streaming environment is defined on the basis of two attributes in our experiments: a) window size and, b) overlap window size or overlap window percentage. Please refer to **Streaming Environment Setup chapter** for further details on these attributes.

Anomaly Detection:

During the experiments, both MIC and dCor were applied on windows to find the dependency between data in consecutive windows. If the change in dependency values is more than a *change threshold* defined

for an experiment then it would be considered an anomaly. Now the performance of both the measures will be measured on the basis of how many anomalies have been identified correctly. F1 score, Precision, and Recall would be used for performance comparison between both the measures. The process of anomaly detection would be discussed in detail during the report.

Outline:

The rest of this thesis is organized as follows: In Section 2, Correlation measures are explained in detail. In section 3, Dataset is defined in detail and Data labeling process will also be defined in detail. In section 4, Simulating Streaming Environment setup would be explained. In section 5, Anomaly Detection approach would be explained. In section 6, Experiments Configurations and Results would be explained. In section 7, Future work and Conclusion.

Chapter 2

Details Of Correlation Measures

Distance correlation(Dcor):

Distance correlation was introduced by Szekely *et al.* (2007) [1] to identify non-linear relationships between two random variables. Distance correlation is a measure of statistical dependence between two random variables or two random vectors . Distance correlation value ranges between 0-1. Dcor returns value zero if and only if the random variables are statistically independent, contrary to Pearson's correlation, which can be zero for dependent random variables. Dcor returns 1 if and only if variables being analyzed are highly dependent.

The distance correlation is derived from many other quantities that are used in its calculation, those are:

- Distance variance.
- Distance standard deviation.
- Distance covariance.

The name distance correlation derived from the fact that it is based on the concept of energy distances (a statistical distance between probability distributions).

Distance Covariance:

Suppose We have two vectors X,Y of n dimension. Two distance matrices of N x N dimension is calculated using Euclidean norm for both vectors, called A and B respectively. Now distance covariance is the arithmetic average of Products of these two distance matrices.

$$\mathrm{dCov}_n^2(X,Y) := \frac{1}{n^2}\sum_{j=1}^{n}\sum_{k=1}^{n} A_{j,k}\,B_{j,k}.$$

Distance Variance & Standard Deviation:

The distance variance is a special case of distance covariance when the two variables are identical.

$$\mathrm{dVar}_n^2(X) := \mathrm{dCov}_n^2(X,X) = \tfrac{1}{n^2}\sum_{k,\ell} A_{k,\ell}^2,$$

The distance standard deviation is the square root of the distance variance.

The distance correlation of two random variables is obtained by dividing their distance covariance by the product of their distance standard deviations. The distance correlation is

$$\mathrm{dCor}(X,Y) = \frac{\mathrm{dCov}(X,Y)}{\sqrt{\mathrm{dVar}(X)\ \mathrm{dVar}(Y)}},$$

dCor(X,Y) = Distance correlation between two variables X,Y.

dCov(X,Y) = Distance covariance between two variables X,Y.

dVar(X) = Distance variance of X.

dVar(Y) = Distance variance of Y.

Energy package of R-studio can be used to calculate Dcor.

Maximal Information Coefficient:

Instinctively, MIC[2] depends on the idea, that if any sort of connection exists between two arbitrary data vectors, then a grid can be drawn on the scatter plot of the two data vectors with the end goal being, that it segments the information to capture the relationship between the random data vectors.

MIC for two variable data is calculated by exploring maximal grid resolution, depending on the sample size. Now the largest possible MI achievable by any a-by-b grid is applied to the data.

MIC can capture linear or non linear dependence between two variables X and Y. MIC does binning on the scatter plot and applies Mutual Information(MI) on variables. MIC selects the number of bins and it picks the maximum over many possible grids. MIC uses MI internally, it is a measure of dependence between two variables.

Mutual Information(MI):

MI is one of many quantities that measure how much one random variable tells us about another. The MI of two continuous random variables *X* and *Y* can be defined as:

$$I(X;Y) = \sum_{x,y} P_{XY}(x,y) \log \frac{P_{XY}(x,y)}{P_X(x) P_Y(y)}$$

Here, $P_X(x)$ and $P_Y(y)$ are the marginal probabilities of variables x and y respectively.

MI can have only positive values. High MI shows a large increase in dependency between two random variables. Low MI shows a small dependence between two random variables. If MI is zero between two random variables means the variables are independent [3].

Now after selecting multiple bins, these MI values are normalized to ensure a fair comparison between grids of different dimensions and to obtain modified values between 0 and 1. The characteristic matrix *M* is defined as $M=(m_{a,b})$, where $(m_{a,b})$ is the highest normalized MI achieved by an a-by-b grid, and the statistic MIC to be the maximum value in *M*.

Formally, for a grid G, let MI_G denote the MI of the probability distribution induced on the boxes of *G*, where the probability of a box is proportional to the number of data points falling inside the box. The (a,b)-th entry $m_{a,b}$ of the characteristic matrix equals :

$$M_{a,b} = \frac{(MI_G)}{\log \min\{a, b\}},$$

, where the maximum is taken over all a-by-b grids *G*. MIC is the maximum of $m_{a,b}$ over ordered pairs (a,b) such that $ab < B$, where *B* is a function of sample size. Usually, set $B=n^{0.6}$.

Mine Package of R-studio can be used to calculate MIC between two random data vectors.

Chapter 3

Dataset and Labeling Process

Dataset:

Data being used in our experiments is energy data. It consists of two variables related to Energy consumption. Current and Power are the two variables which we have chosen to record for our dataset. Data has been recorded at an interval of 30 seconds, i.e. a data point is inserted in dataset every 30 seconds. We have recorded data over a period of 4 days, so that many number of change points can be captured in a dataset. An instance of dataset recorded is shown below.

Data and Time	Power	Current
2017-02-24 06:00:19 IST	2321.33	3.436488
2017-02-24 06:00:51 IST	2324.054	3.437996
2017-02-24 06:01:22 IST	2324.91	3.436389
2017-02-24 06:01:53 IST	2326.969	3.438613
2017-02-24 06:02:24 IST	2327.761	3.438796
2017-02-24 06:02:55 IST	2327.054	3.438158
2017-02-24 06:03:26 IST	2333.285	3.453826
2017-02-24 06:03:57 IST	2335.995	3.450533
2017-02-24 06:04:29 IST	2331.504	3.447939
2017-02-24 06:05:00 IST	2330.86	3.447602
2017-02-24 06:05:31 IST	2329.883	3.44761
2017-02-24 06:06:02 IST	2329.312	3.447152
2017-02-24 06:06:33 IST	2349.477	3.471178
2017-02-24 06:07:04 IST	2335.327	3.451024
2017-02-24 06:07:36 IST	2328.997	3.450494

Figure 1: Dataset instance before labelling

Labeling Process:

After recording the dataset, labeling of dataset has been done for anomalous data points. Labeling has been done in two ways to make two different datasets.

Labeling Process for Dataset1:

Now unlabeled dataset is used for generating dataset1. Mean is used as a measure to mark anomalous rows in dataset. Now if any row in the dataset differs more than 5 percent from mean of previous values 10 values than it is considered as an anomaly. An extra column is added in dataset by name "Anomaly". If value differs more than 5 percent from mean of previous 10 values, than 1 is inserted in new column otherwise 0. So 1 represents anomalous row and 0 represents non-anomalous row.

Labeling Process for Dataset2:

Now unlabeled dataset is used for generating dataset2. Mean is used as a measure to mark anomalous rows in dataset. Now if any row in the dataset differs more than 10 percent from mean of previous values 10 values than it is considered as an anomaly. An extra column is added in dataset by name "Anomaly". If value differs more than 5 percent from mean of previous 10 values, than 1 is inserted in new column otherwise 0. So 1 represents anomalous row and 0 represents non-anomalous row.

Date and Time	Power	Current	Anomaly
2017-02-24 06:00:51 IST	2324.053711	3.437996149	0
2017-02-24 06:01:22 IST	2324.909912	3.436389208	0
2017-02-24 06:01:53 IST	2326.96875	3.438613415	0
2017-02-24 06:02:24 IST	2327.760986	3.438795567	0
2017-02-24 06:02:55 IST	2327.054443	3.438158035	0
2017-02-24 06:03:26 IST	2333.285156	3.453825951	0
2017-02-24 06:03:57 IST	2335.995361	3.450532913	0
2017-02-24 06:04:29 IST	2331.50415	3.447939396	0
2017-02-24 06:05:00 IST	2330.860107	3.447601557	0
2017-02-24 06:05:31 IST	2329.882813	3.447610378	0
2017-02-24 06:06:02 IST	2329.311768	3.447152376	0
2017-02-24 06:06:33 IST	2349.476563	3.471177816	0
2017-02-24 06:07:04 IST	2335.327393	3.451024055	0
2017-02-24 06:07:36 IST	2328.99707	3.450494051	0
2017-02-24 06:08:07 IST	2330.777344	3.454977512	0
2017-02-24 06:08:38 IST	2327.127686	3.452737331	0
2017-02-24 06:09:09 IST	2327.231689	3.452677727	0
2017-02-24 06:09:40 IST	2325.802979	3.451361895	0
2017-02-24 06:10:11 IST	2322.821533	3.450975418	0
2017-02-24 06:10:42 IST	1787.740601	2.661090136	1
2017-02-24 06:11:13 IST	1784.213623	2.659864426	1
2017-02-24 06:11:44 IST	1783.092041	2.659712076	1
2017-02-24 06:12:15 IST	1552.394043	2.319708347	1
2017-02-24 06:12:46 IST	1550.162598	2.320430279	1

Figure 2: Dataset instance after labelling

Chapter 4

Streaming Environment Setup

Streaming environment is simulated by using the concept of windows and overlap between consecutive windows. Data available to us for experiments is accessed on window basis.

Now understand the concept of windows better, these two terms need to be explained in detail :

- Window Size.
- Overlap Window Size or Overlap Percent.

Window Size:

Windows in our streaming environment simulation are defined on the basis of number of data points it consists. Windows are not defined on the basis of time interval in our experiments. Window Size attribute defines the number of data point to be accessed for each window. For ex : if the window size is 10 than data points which would be accessed first are shown in the figure.

	Date and Time	Power	Current	Anomaly
1	2017-02-24 06:00:51 IST	2324.053711	3.437996149	0
2	2017-02-24 06:01:22 IST	2324.909912	3.436389208	0
3	2017-02-24 06:01:53 IST	2326.96875	3.438613415	0
4	2017-02-24 06:02:24 IST	2327.760986	3.438795567	0
5	2017-02-24 06:02:55 IST	2327.054443	3.438158035	0
6	2017-02-24 06:03:26 IST	2333.285156	3.453825951	0
7	2017-02-24 06:03:57 IST	2335.995361	3.450532913	0
8	2017-02-24 06:04:29 IST	2331.50415	3.447939396	0
9	2017-02-24 06:05:00 IST	2330.860107	3.447601557	0
10	2017-02-24 06:05:31 IST	2329.882813	3.447610378	0
11	2017-02-24 06:06:02 IST	2329.311768	3.447152376	0
12	2017-02-24 06:06:33 IST	2349.476563	3.471177816	0
13	2017-02-24 06:07:04 IST	2335.327393	3.451024055	0
14	2017-02-24 06:07:36 IST	2328.99707	3.450494051	0
15	2017-02-24 06:08:07 IST	2330.777344	3.454977512	0
16	2017-02-24 06:08:38 IST	2327.127686	3.452737331	0
17	2017-02-24 06:09:09 IST	2327.231689	3.452677727	0
18	2017-02-24 06:09:40 IST	2325.802979	3.451361895	0
19	2017-02-24 06:10:11 IST	2322.821533	3.450975418	0
20	2017-02-24 06:10:42 IST	1787.740601	2.661090136	1

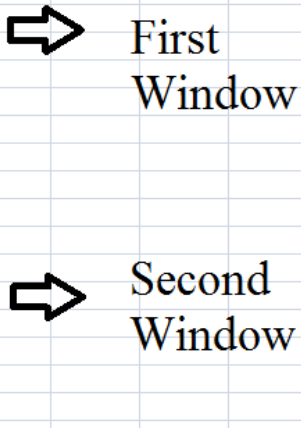


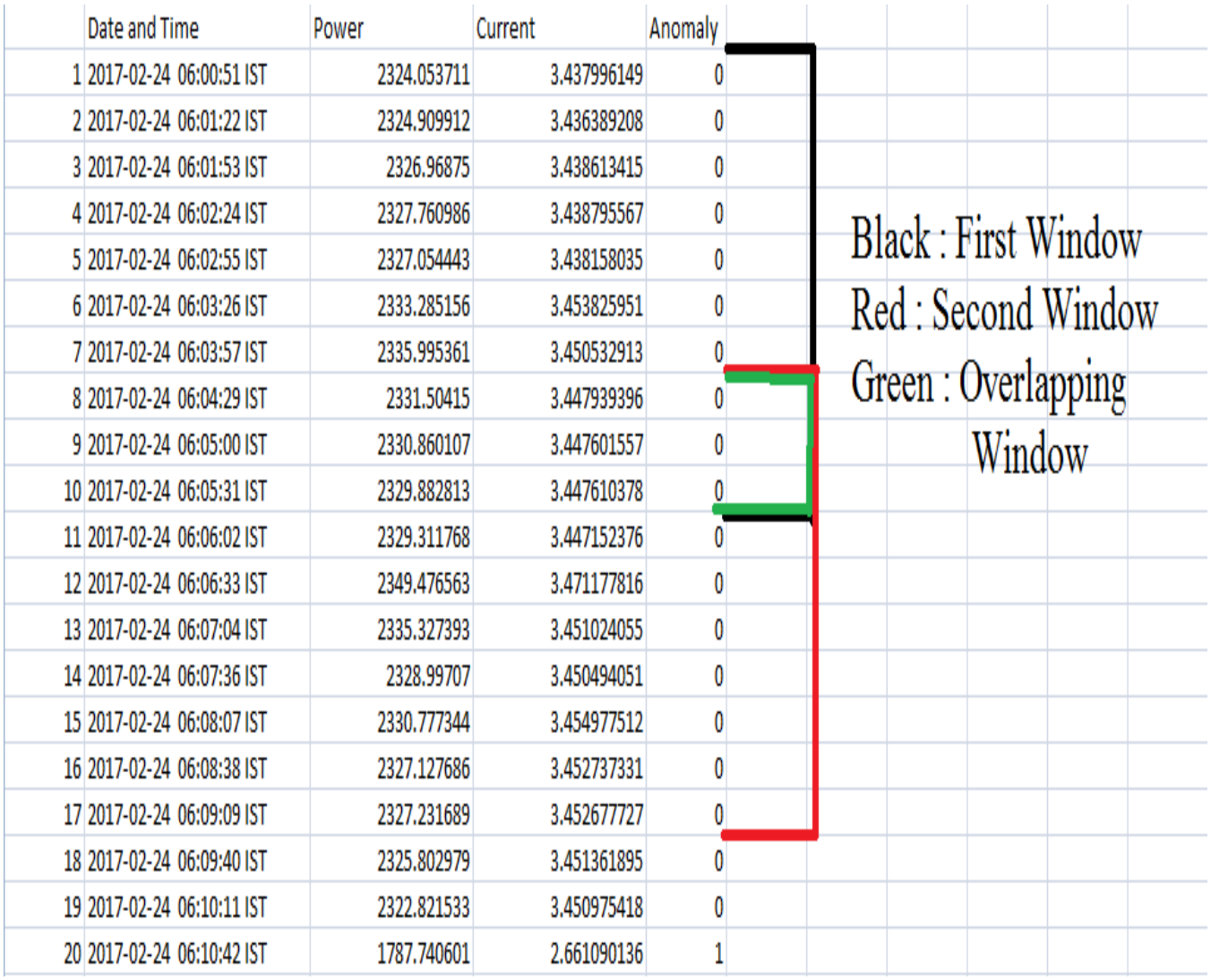
Figure 3: Environment set-up with window size 10

Overlap Window Size or Overlap Percent:

Overlap Window Size attribute in our simulation environment setup tells us about the common data points between two consecutive windows. Overlap Percent tells us how much percent of window is common between consecutive windows. For ex : Window size = 10, overlap percent = 30 % than overlap window size = 3. Figure below depicts the above example.

Steps for Simulating Streaming Environment:

- Firstly we fix attributes such as Window Size and Overlap Percent between windows.
- Secondly we start accessing our dataset in window basis. Window properties are specified by the above attributes.
- Thirdly we loop over the whole data set in terms of windows and overlap windows.



	Date and Time	Power	Current	Anomaly
1	2017-02-24 06:00:51 IST	2324.053711	3.437996149	0
2	2017-02-24 06:01:22 IST	2324.909912	3.436389208	0
3	2017-02-24 06:01:53 IST	2326.96875	3.438613415	0
4	2017-02-24 06:02:24 IST	2327.760986	3.438795567	0
5	2017-02-24 06:02:55 IST	2327.054443	3.438158035	0
6	2017-02-24 06:03:26 IST	2333.285156	3.453825951	0
7	2017-02-24 06:03:57 IST	2335.995361	3.450532913	0
8	2017-02-24 06:04:29 IST	2331.50415	3.447939396	0
9	2017-02-24 06:05:00 IST	2330.860107	3.447601557	0
10	2017-02-24 06:05:31 IST	2329.882813	3.447610378	0
11	2017-02-24 06:06:02 IST	2329.311768	3.447152376	0
12	2017-02-24 06:06:33 IST	2349.476563	3.471177816	0
13	2017-02-24 06:07:04 IST	2335.327393	3.451024055	0
14	2017-02-24 06:07:36 IST	2328.99707	3.450494051	0
15	2017-02-24 06:08:07 IST	2330.777344	3.454977512	0
16	2017-02-24 06:08:38 IST	2327.127686	3.452737331	0
17	2017-02-24 06:09:09 IST	2327.231689	3.452677727	0
18	2017-02-24 06:09:40 IST	2325.802979	3.451361895	0
19	2017-02-24 06:10:11 IST	2322.821533	3.450975418	0
20	2017-02-24 06:10:42 IST	1787.740601	2.661090136	1

Figure 4: Environment set-up with overlapping window size

While simulating streaming environment, Correlation measures are applied on each window. Each window is analyzed on the basis of their MIC and dCor scores. Now these attributes window size and overlap percent also have effects on the MIC and dCor scores of the windows. If the window size is large enough than it improves the MIC score, as MIC can find out dependency better if sample size is larger. dCor performs better in smaller window size also.

Different configurations of Streaming environments are used to run multiple experiments on both the dataset. Configuration of streaming

environment will also have certain impact on the performance of both MIC and dcor in anomaly detection process.

Anomaly detection using this streaming environment setup would be explained in next section of this report.

Chapter 5

Anomaly Detection Process

Anomaly detection process depends on three attributes :

- Window Size(WS).
- Overlap percent(OP).
- Change Threshold(CH).

Window size and overlap percent have been explained in detail in Streaming environment setup. Change Threshold is main attribute used to define whether a window consists of anomalous rows or not.

Change Threshold (CH):

This attribute is used to define the maximum amount of change allowed in MIC or dCor scores of consecutive windows. If the MIC/dCor scores of two consecutive windows changes more than the threshold defined, than that window would be considered to have some anomalous rows in it. For ex : if CH=1 %, than if scores between two windows vary more than 1%, it would be considered anomaly will be checked for anomalous rows in that window. If anomalous rows are present than it would be marked as TP(True Positive) otherwise FP(False Positive).

IF the value doesn't change more than the specified threshold, than window would be analyzed for anomalous row. If anomalous rows exist in the window than it would be marked as FN(False Negative) otherwise it would be marked as TN(True Negative).

Idea Behind Using Correlation Measures For Anomaly Detection:

Correlation measures can give us an idea about how much two variables are dependent on each other. When that relationship between two variable changes, correlation score is bound to change. This means if a change point occurs in any window, correlation score of that window is going to change compared to previous window.

Now overlap between window also helps us some time, consider if two windows are not completely same but are 30 % same. If relationship changes in the rest 70 % of consecutive window, than correlation score of this window is going change a lot compared to previous window. This will help us to determine that the consecutive window consists of anomalous rows.

Now if Overlap percent is quite high it means that both the windows are almost same, this won't help us as correlation score of both the windows won't change much. So high overlap percent would reduce the performance of both the correlation measure in detecting anomaly. This pattern has been observed in our experiment.

Window Size also effects the performance of MIC measure while finding dependency between variables in data. In our experiments it has been observed that increasing window size up to a point increases MIC performance. MIC performs well over a large sample size instead of a smaller sample size is shown by A. Adam Ding and Yi L,2013 [4].

Chapter 6

Experiments and Results

Experiments are conducted in different streaming environment configuration. Details are given below :

- Window Sizes is taken from 100 to 900.
- Overlap Percent is taken from 10 to 90.
- Change-Threshold percent is taken from 0.5 to 2.
- Graphs have been plotted with F1 scores on Y-axis and Window Sizes on x-axis.
- F1 score of both MIC and dCor are plotted together for comparison purpose. MIC is shown by Blue dots and dCor by Red dots in graphs.
- F1 score for both MIC and dCor are calculated on every window size from 100 to 900 for each overlap percent from 10-90 and for Change threshold value from 0.5 to 2 percent.
- Results are shown by combining three graphs in to one graph. Each graph corresponds to a particular overlap percent value under a particular Threshold change value.

Experiments On DataSet 1:

For Change Threshold = 0.5% and Overlap Percent = 10% - 30%

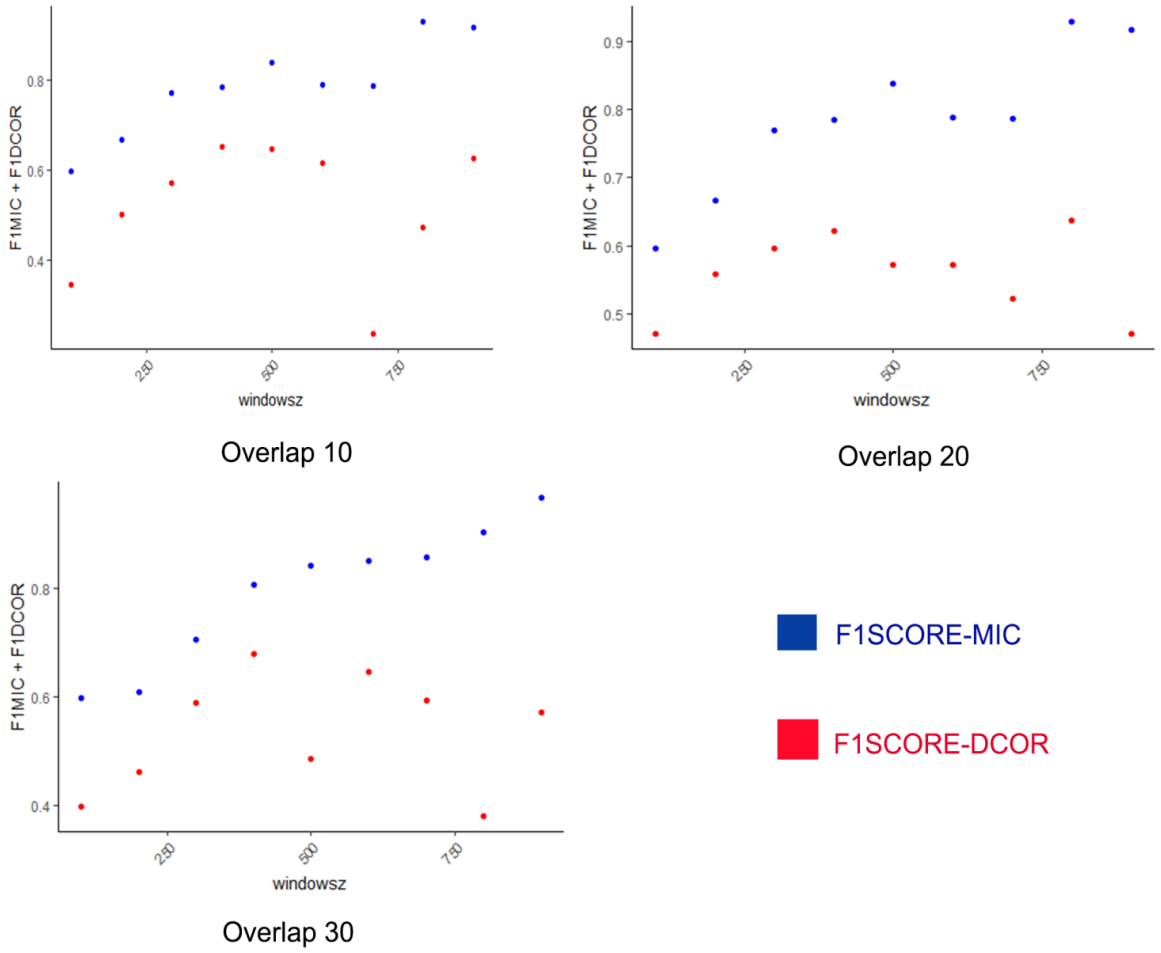


Figure 5: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 10 to 30 and Change Threshold being fixed to 0.5 %. In the above graphs it is easily visible that MIC is performing better than dCor.

Experiments On DataSet 1:

For Change Threshold = 0.5% and Overlap Percent = 40% - 60%

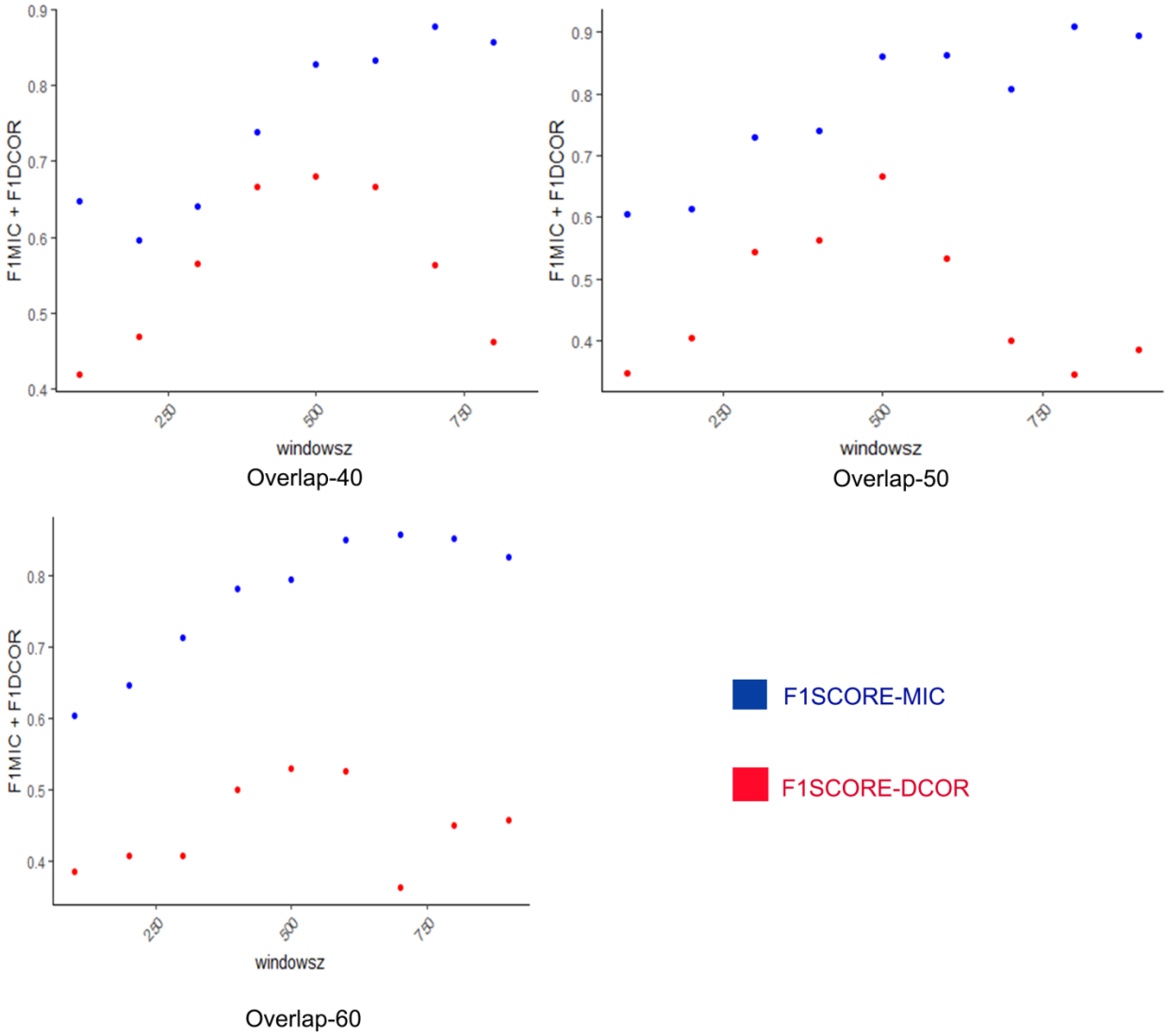


Figure 6: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 40 to 60 and Change Threshold being fixed to 0.5 %. In the above graphs it is easily visible that MIC is performing better than dCor. It can be spotted due to the increase in overlap percent value max F1 score of the

measures has decreased. dCor is significantly more affected by this. Lets observe in next figure, the effects of increasing overlap percent further.

Experiments On DataSet 1:

For Change Threshold = 0.5% and Overlap Percent = 70% - 90%

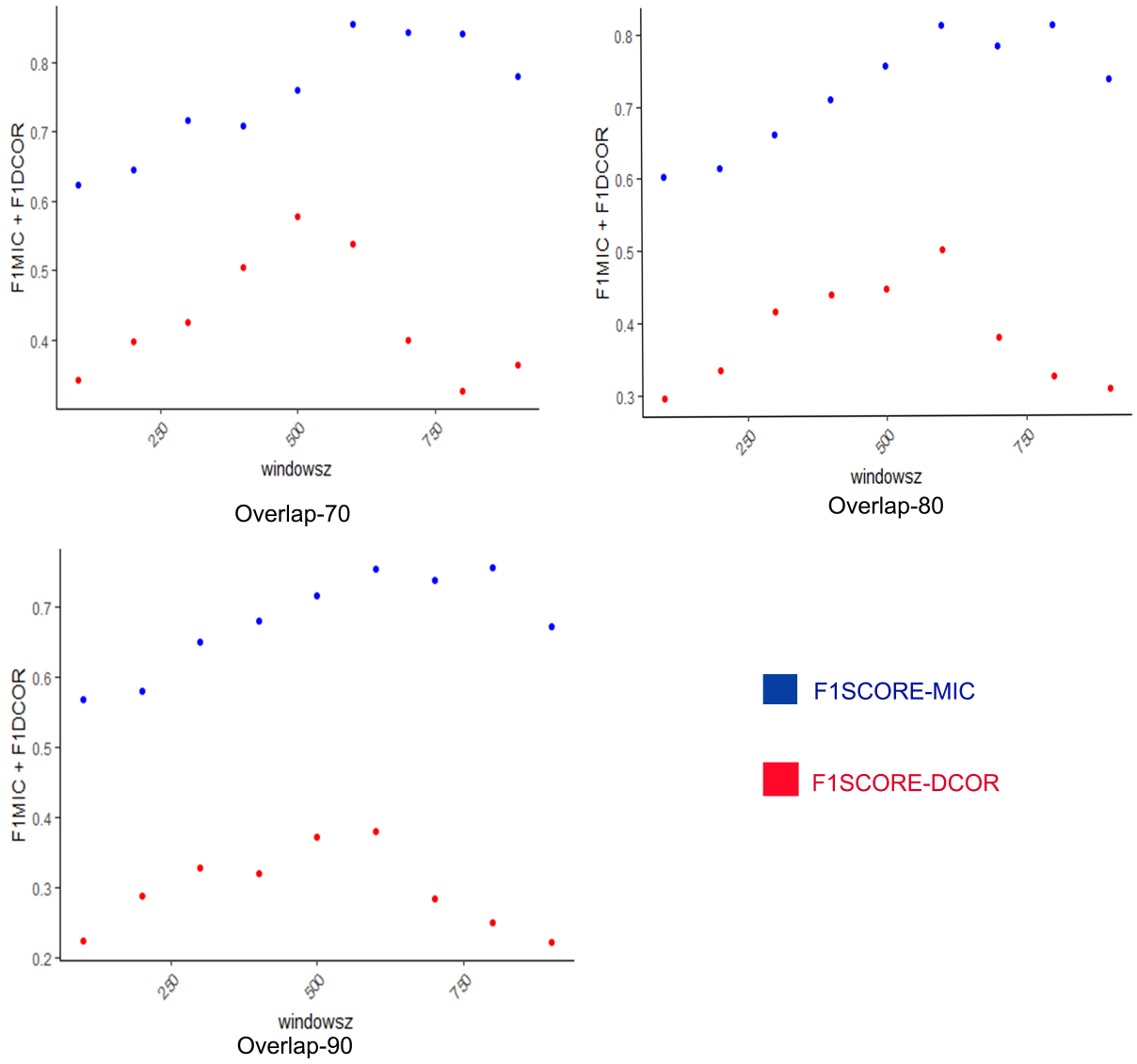


Figure 7: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 70 to 90 and Change Threshold being fixed to 0.5 %. These graph clearly

show the prominent effect of the increased overlap percent on the performance of both the measures. dCor is still more effected by it. But MIC has also been affected.

Experiments On DataSet 2:

For Change Threshold = 1% and Overlap Percent = 10% - 30%.

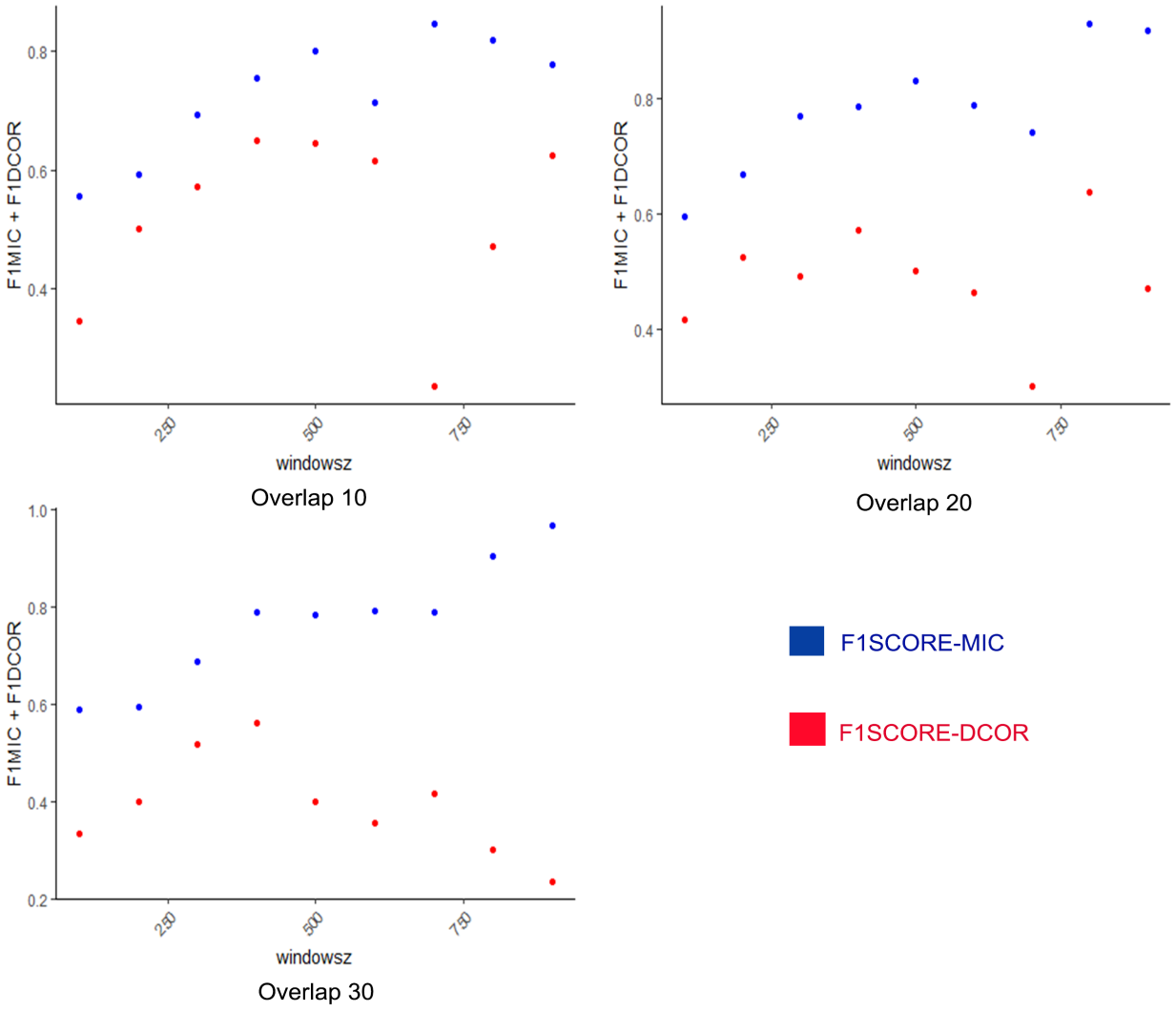


Figure 8: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 10 to 30 and Change Threshold being fixed to 1 %. In the above graphs it is easily visible that MIC is performing better than dCor. Effect of increase in overlap percent from 10 to 30 has increased performance of MIC a bit. For detailed analysis of the effect of overlap percent, plots have been generated with MIC values under different overlap percent in a single graph and same has been done for dCor.

Experiments On DataSet 1:

For Change Threshold = 1% and Overlap Percent = 40% - 60%.

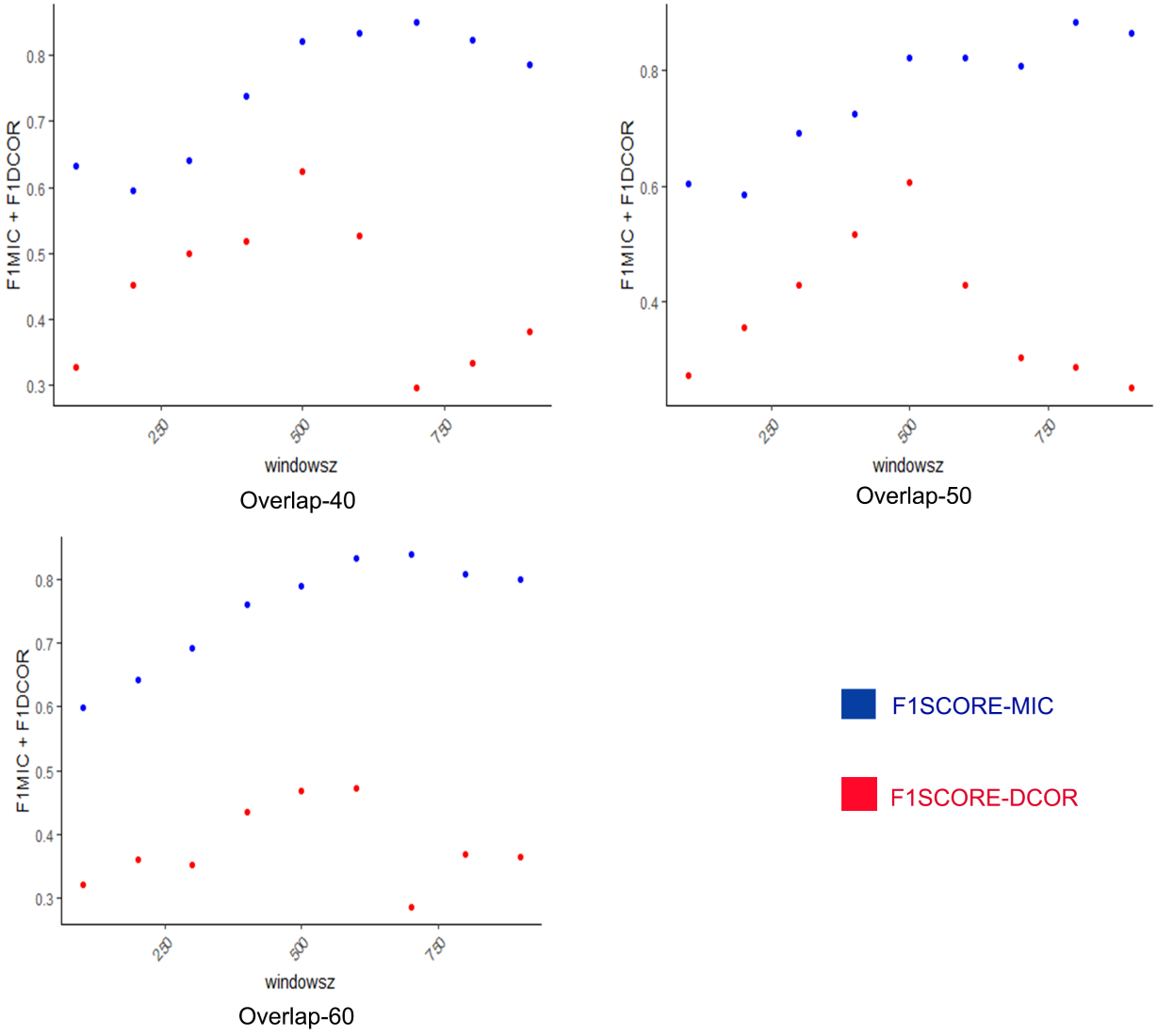


Figure 9: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 40 to 60 and Change Threshold being fixed to 1 %. Similar pattern is being followed as in the case of change threshold 0.5.

Experiments On DataSet 1:

For Change Threshold = 1% and Overlap Percent = 70% - 90%.

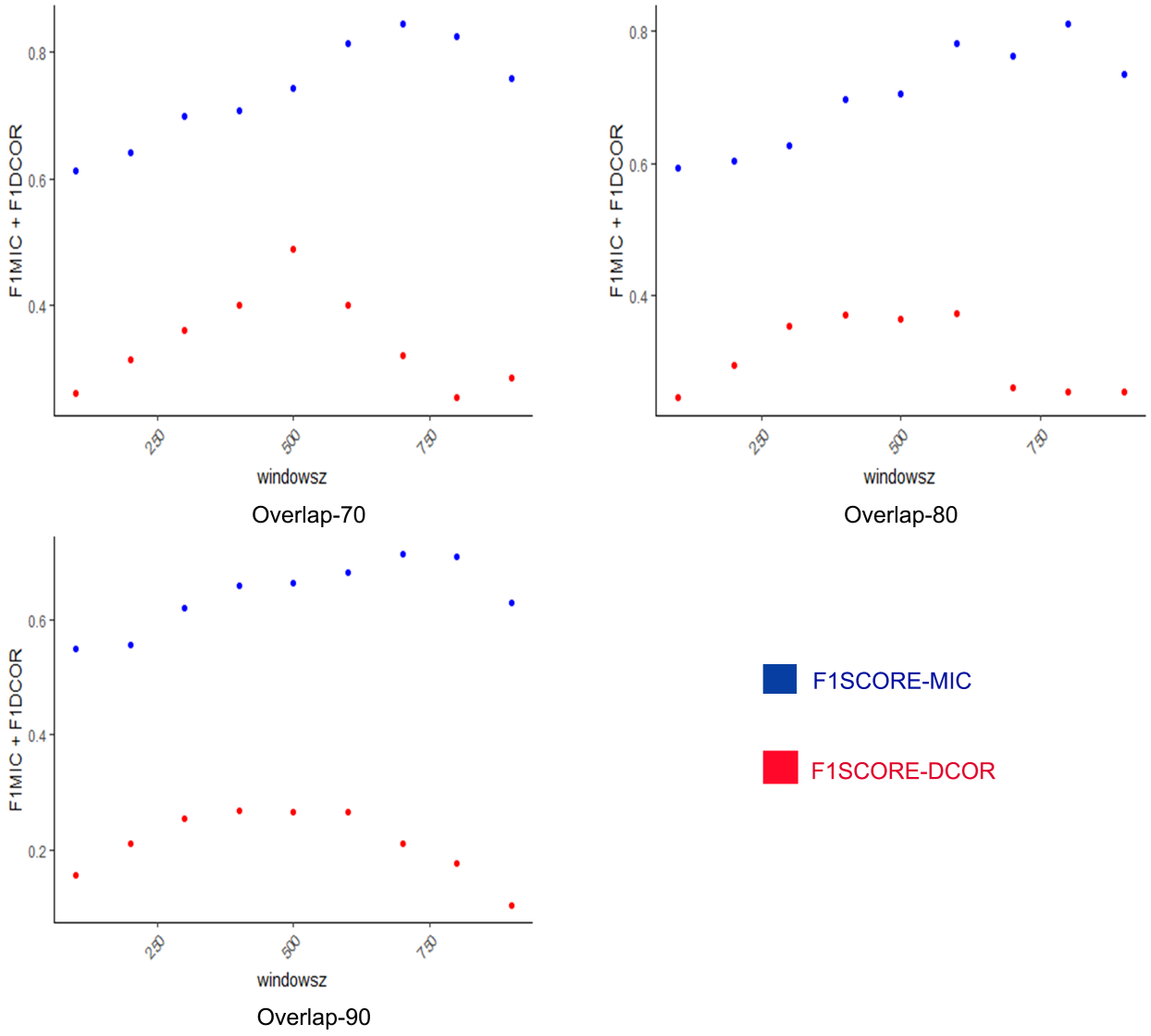


Figure 10: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 70 to 90 and Change Threshold being fixed to 1 %. Similar pattern is being followed as in the case of change threshold 0.5.

Experiments On DataSet 1:

For Change Threshold = 2% and Overlap Percent = 10% - 30%.

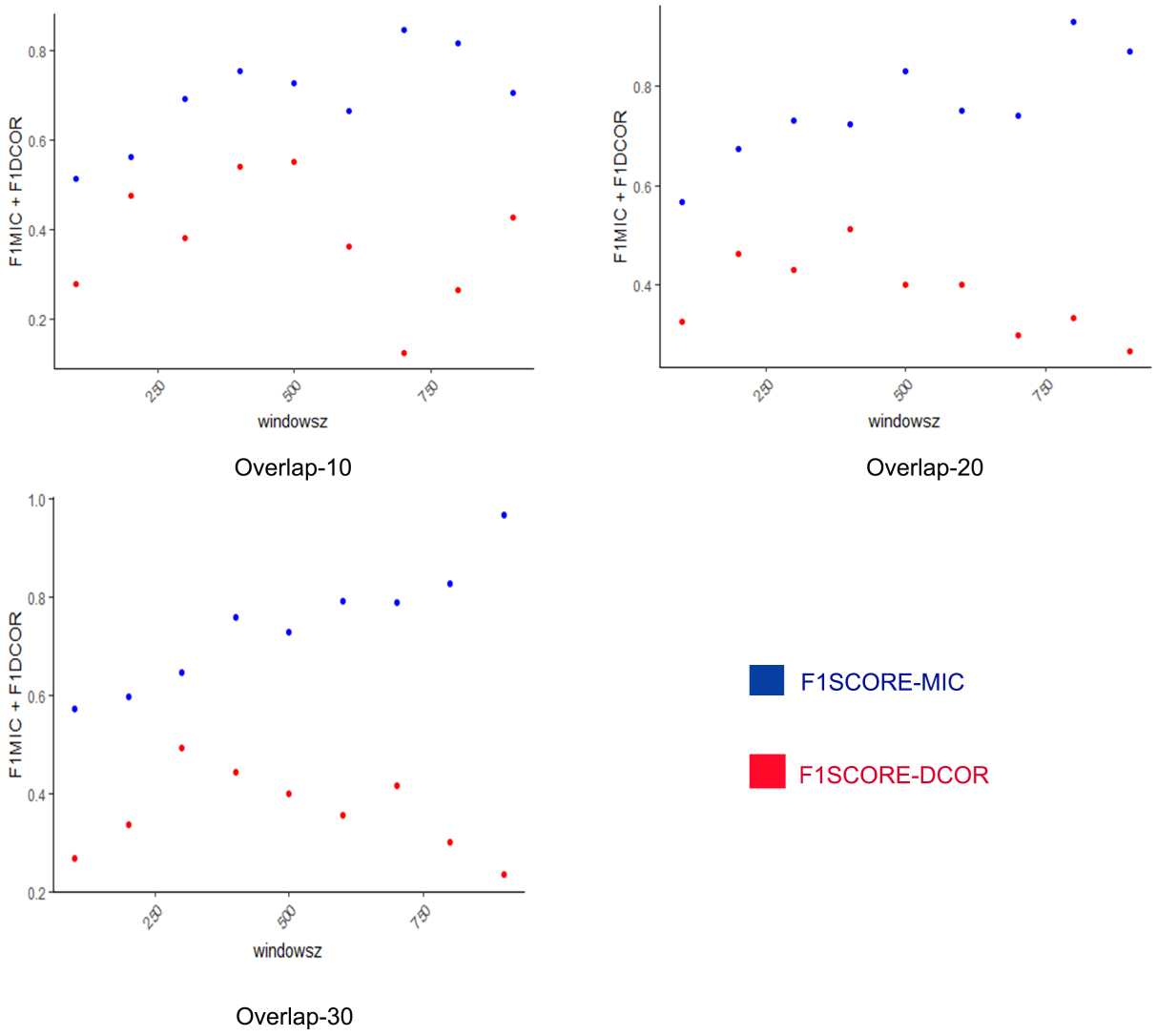


Figure 11: Experiments On DataSet 1

Above figure shows us three graphs for each overlap percent from 10 to 30 and Change Threshold being fixed to 2 %. Till now MIC is performing better under every configuration. Here it can be clearly observed as the overlap percent goes from 10 to 30 MIC performs a bit better.

Experiments On DataSet 1:

For Change Threshold = 2% and Overlap Percent = 40% - 60%.

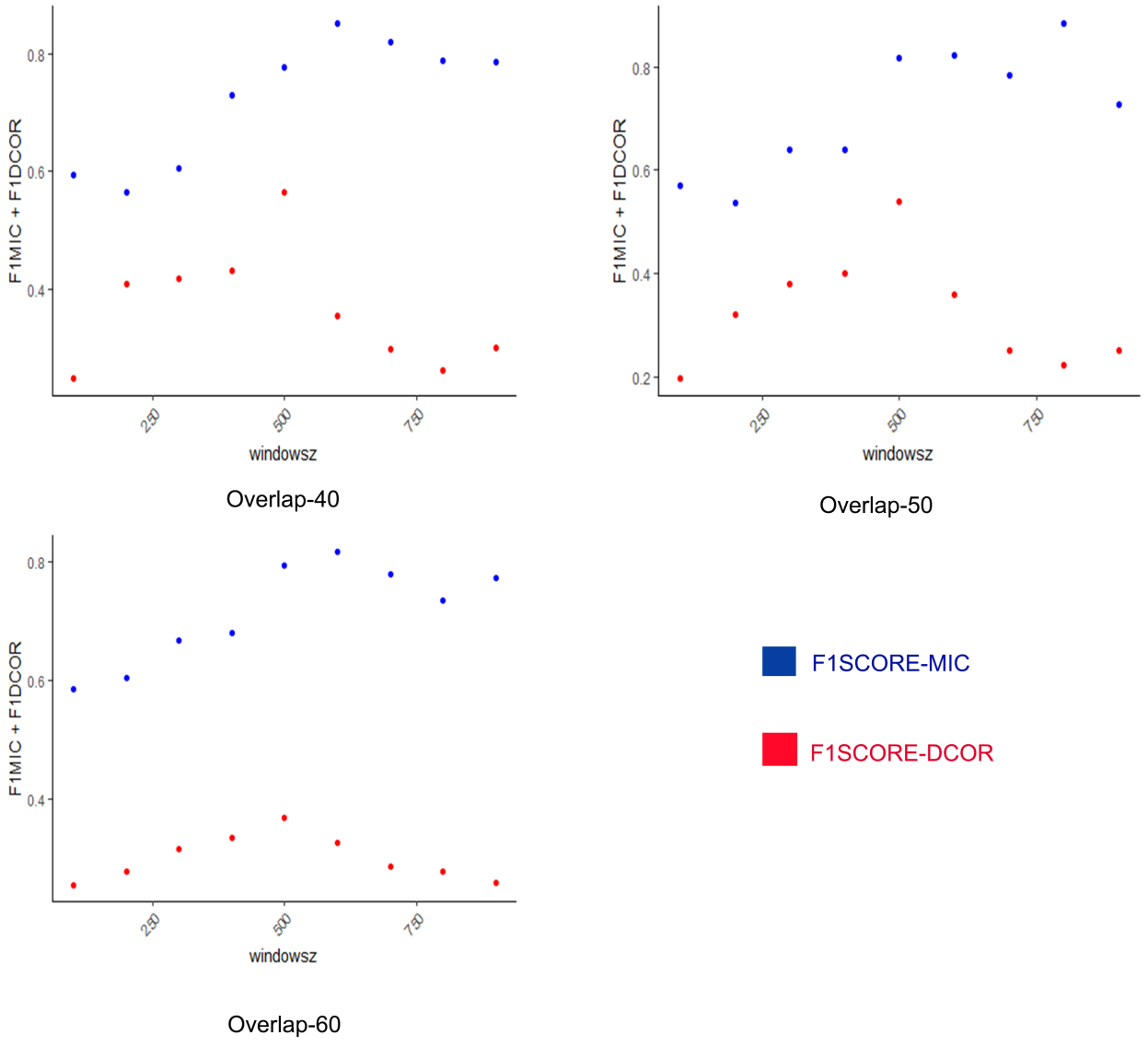


Figure 12: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 40 to 60 and Change Threshold being fixed to 2 %. Till now MIC is performing better under every configuration. Similar pattern to Change threshold 0.5% and 1% is being followed, as performance starts to decrease as overlap percent increases over 40 or 50 percent.

Experiments On DataSet 1:

For Change Threshold = 2% and Overlap Percent = 70% - 90%.

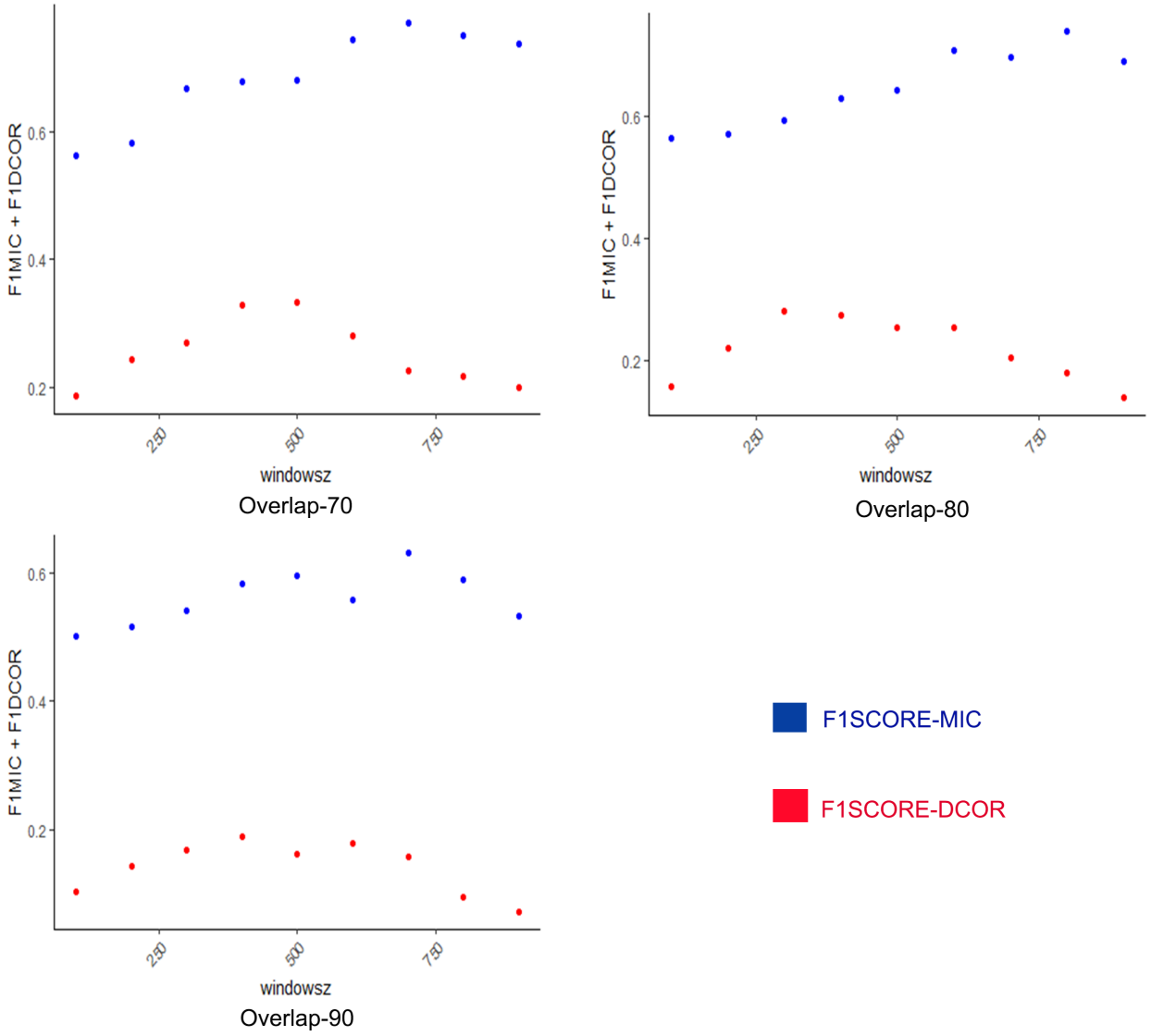


Figure 13: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 70 to 90 and Change Threshold being fixed to 2 %. Till now MIC is performing better under every configuration. Similar pattern to Change threshold 0.5% and 1% is being followed. Change Threshold doesn't have any prominent effect till now. When Change threshold value was kept higher for ex 10% or 20%, performance of dCor was very low and it also had an adverse effect on the performance of MIC.

Experiments On DataSet 2:

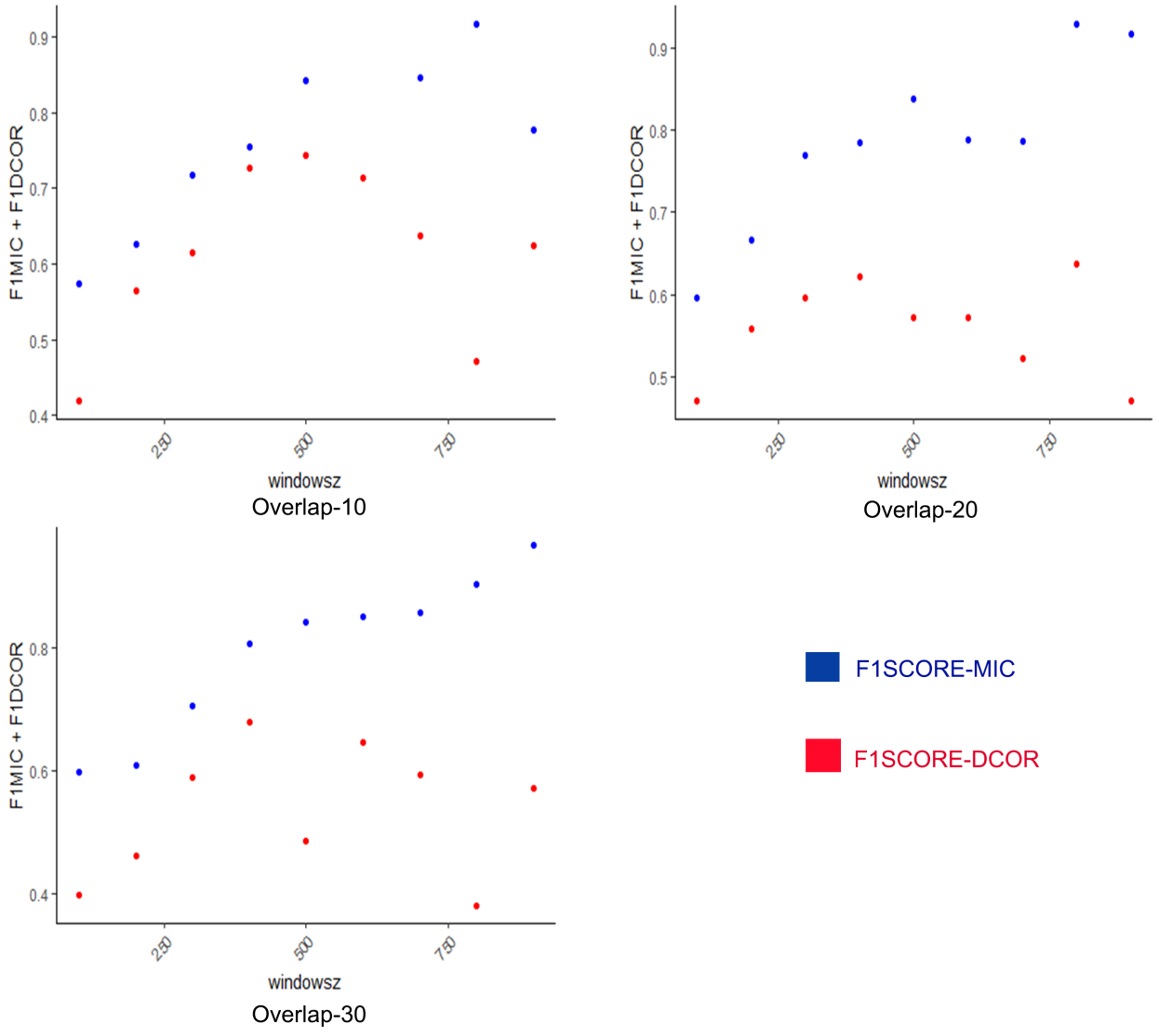


Figure 14: Experiments On DataSet 2

For Change Threshold = 0.5% and Overlap Percent = 10% - 30%.

Above figure shows us three graphs for each overlap percent from 10 to 30 and Change Threshold being fixed to 0.5 %. In Dataset 2 also MIC is performing better than dCor.

Experiments On DataSet 2:

For Change Threshold = 0.5% and Overlap Percent = 40% - 60%.

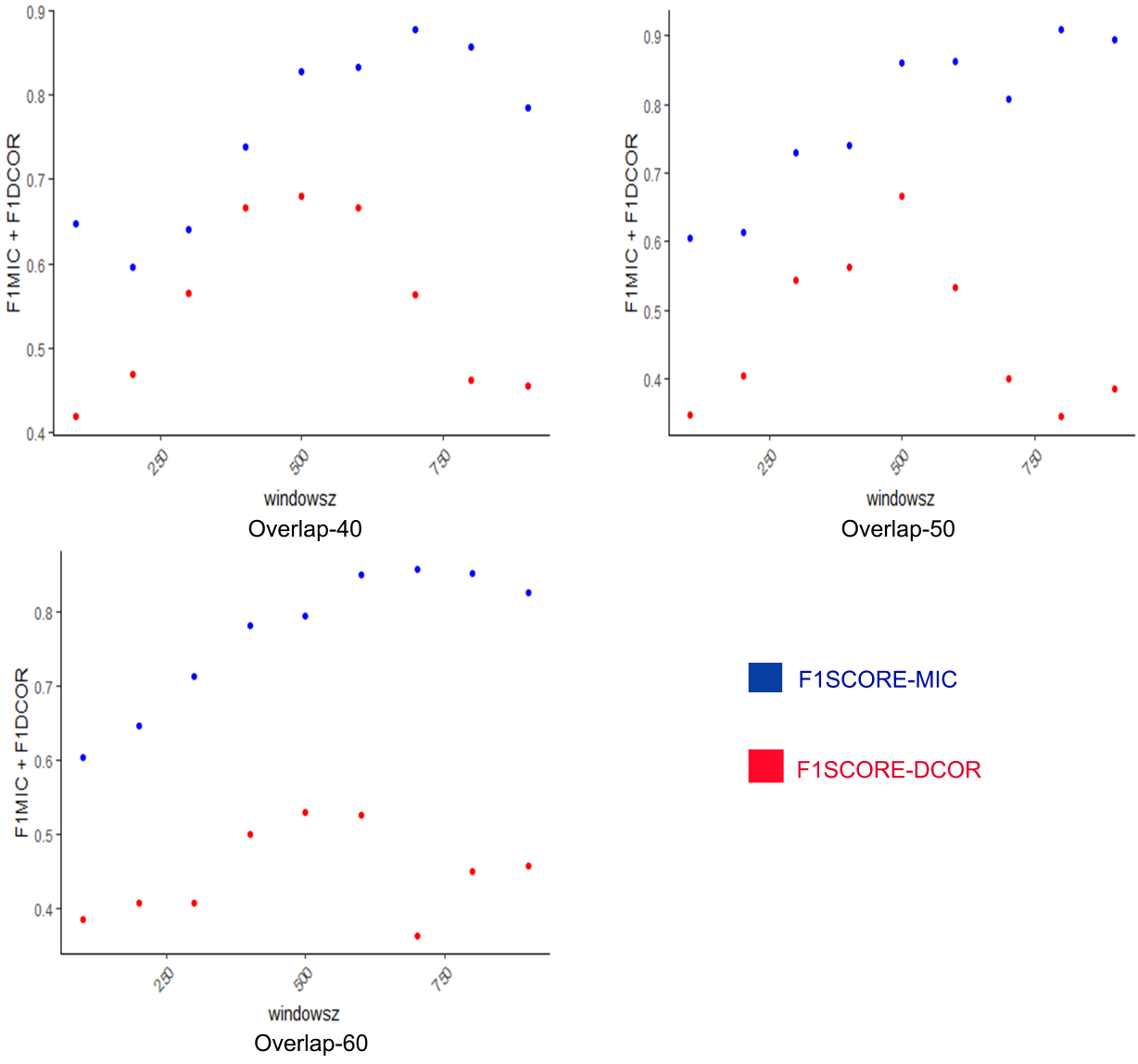


Figure 15: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 40 to 60 and Change Threshold being fixed to 0.5 %. Similar pattern is being followed here, increase in overlap percent has decreased performance of both the measures.

Experiments On DataSet 2:

For Change Threshold = 0.5% and Overlap Percent = 70% - 90%.

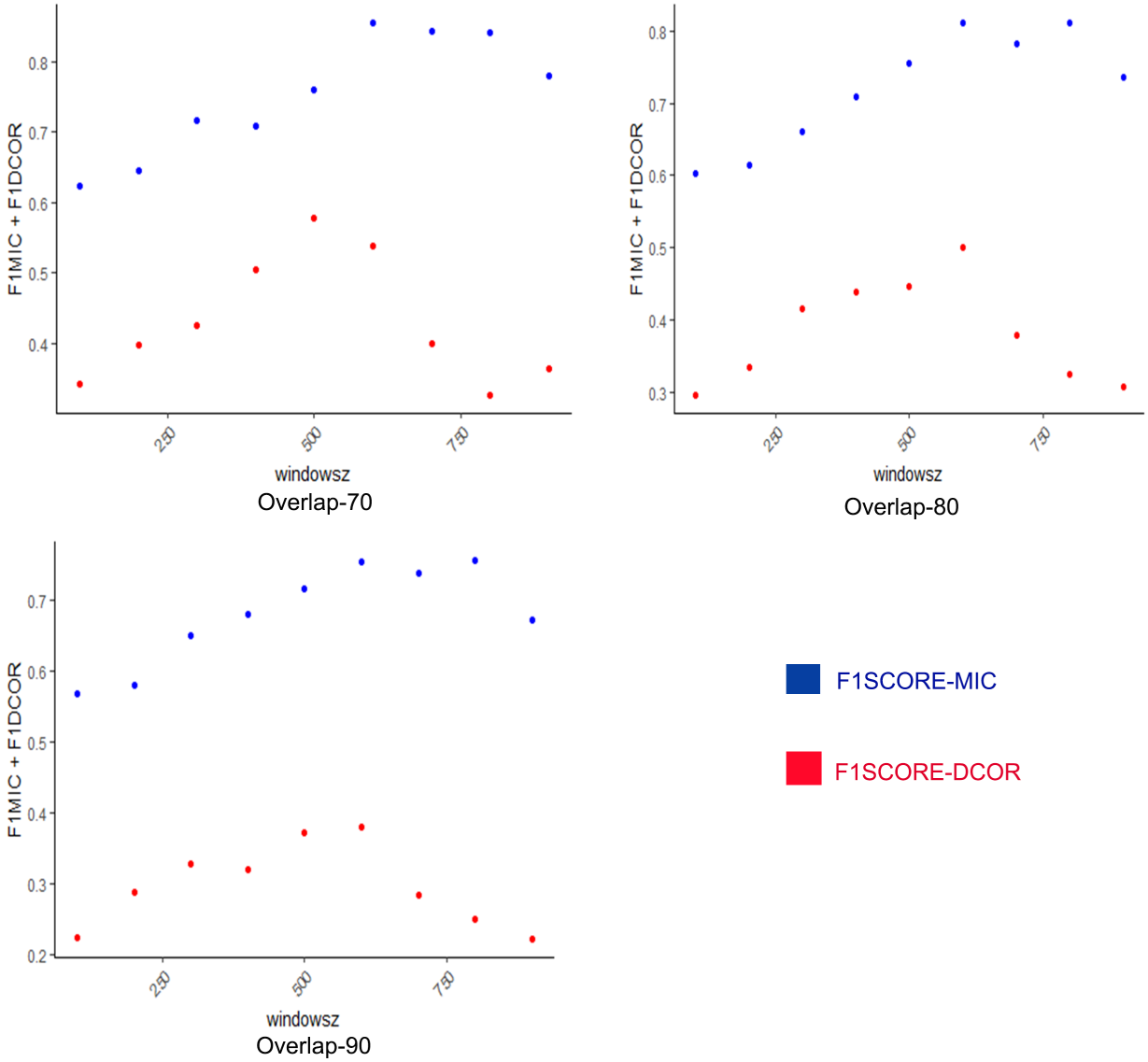


Figure 16: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 70 to 90 and Change Threshold being fixed to 0.5 %. Performance of MIC is much better than dCor in experiments till now.

Experiments On DataSet2:

For Change Threshold = 1% and Overlap Percent = 10% - 30%.

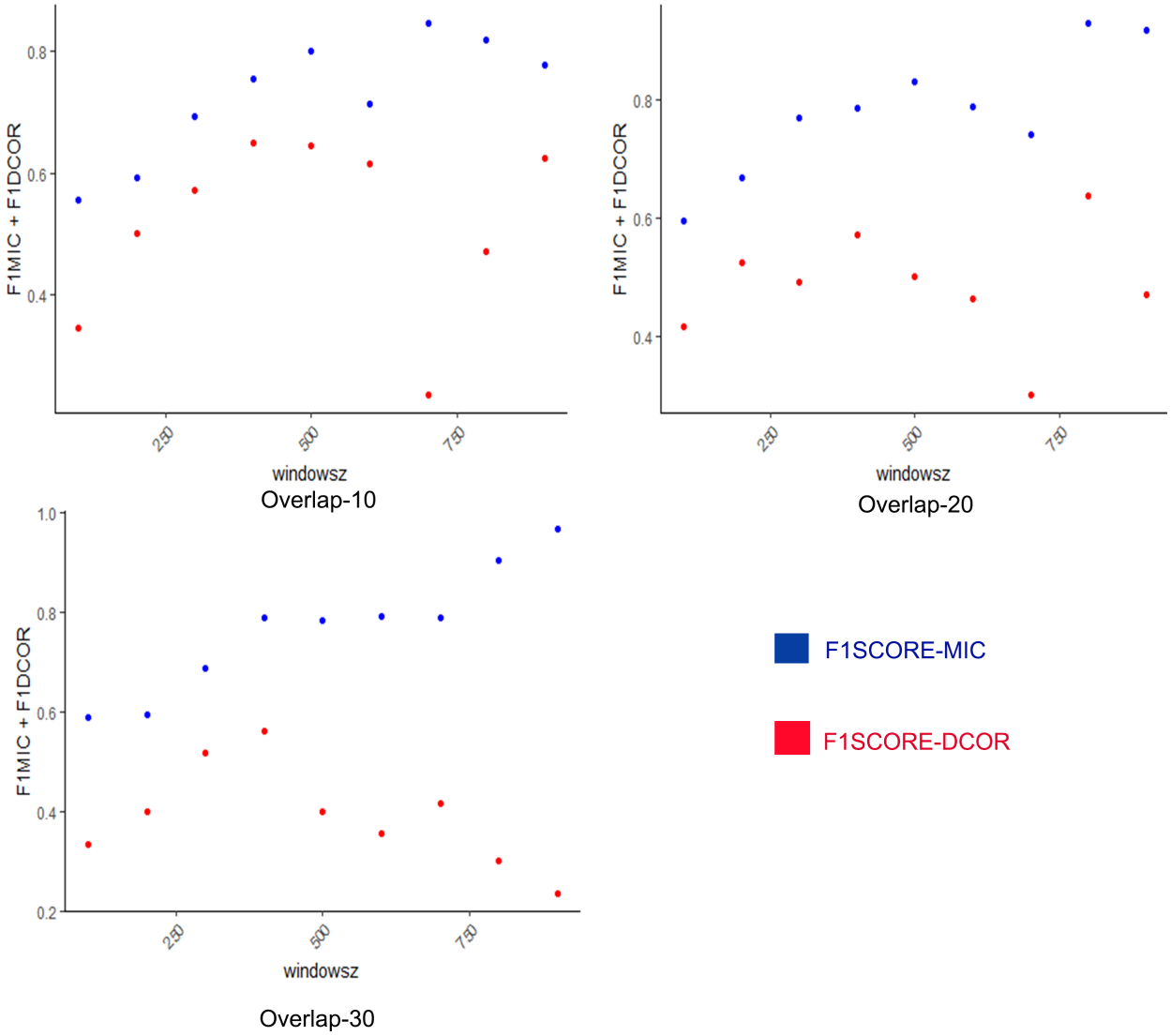
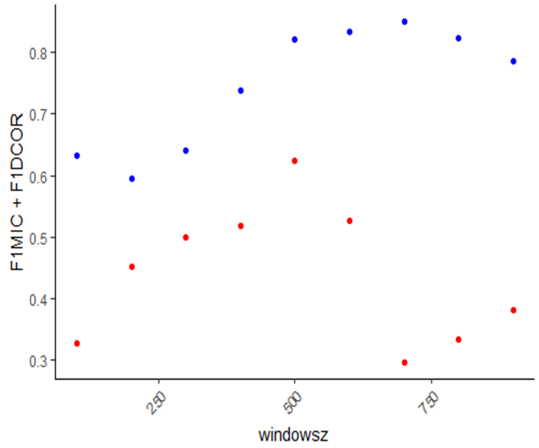


Figure 17: Experiments On DataSet 2

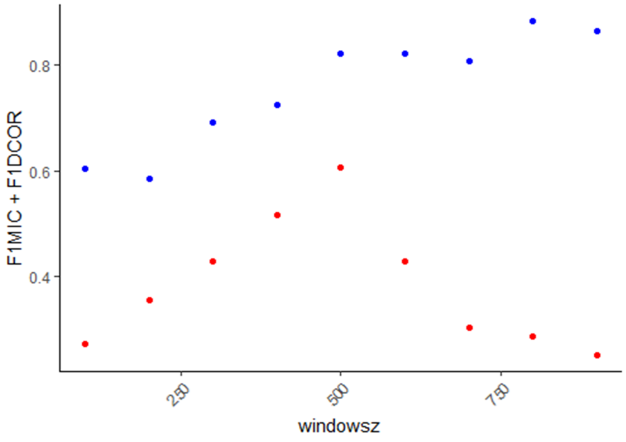
Above figure shows us three graphs for each overlap percent from 10 to 30 and Change Threshold being fixed to 1%. Performance of both the measures is following similar trends, which were seen in previous streaming environment configurations. Here also up to 30% performance of MIC is increasing.

Experiments On DataSet 2:

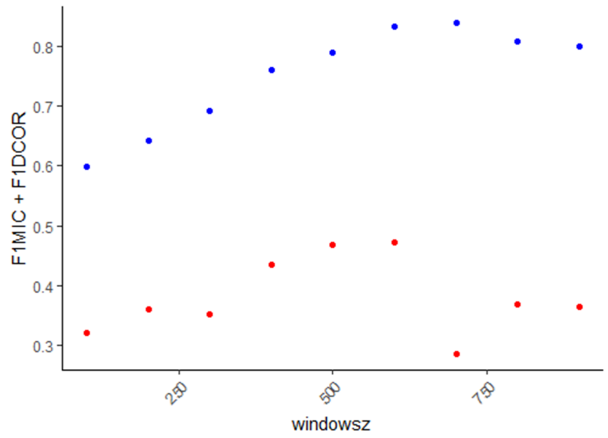
For Change Threshold = 1% and Overlap Percent = 40% - 60%.



Overlap-40



Overlap-50



Overlap-60

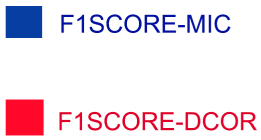


Figure 18: Experiments On DataSet 2

The figure shows us three graphs for each overlap percent from 40 to 60 and Change Threshold being fixed to 1%. Overlap percent increase results in decrease of performance for both the measures.

Experiments On DataSet 2:

For Change Threshold = 1% and Overlap Percent = 70% - 90%.

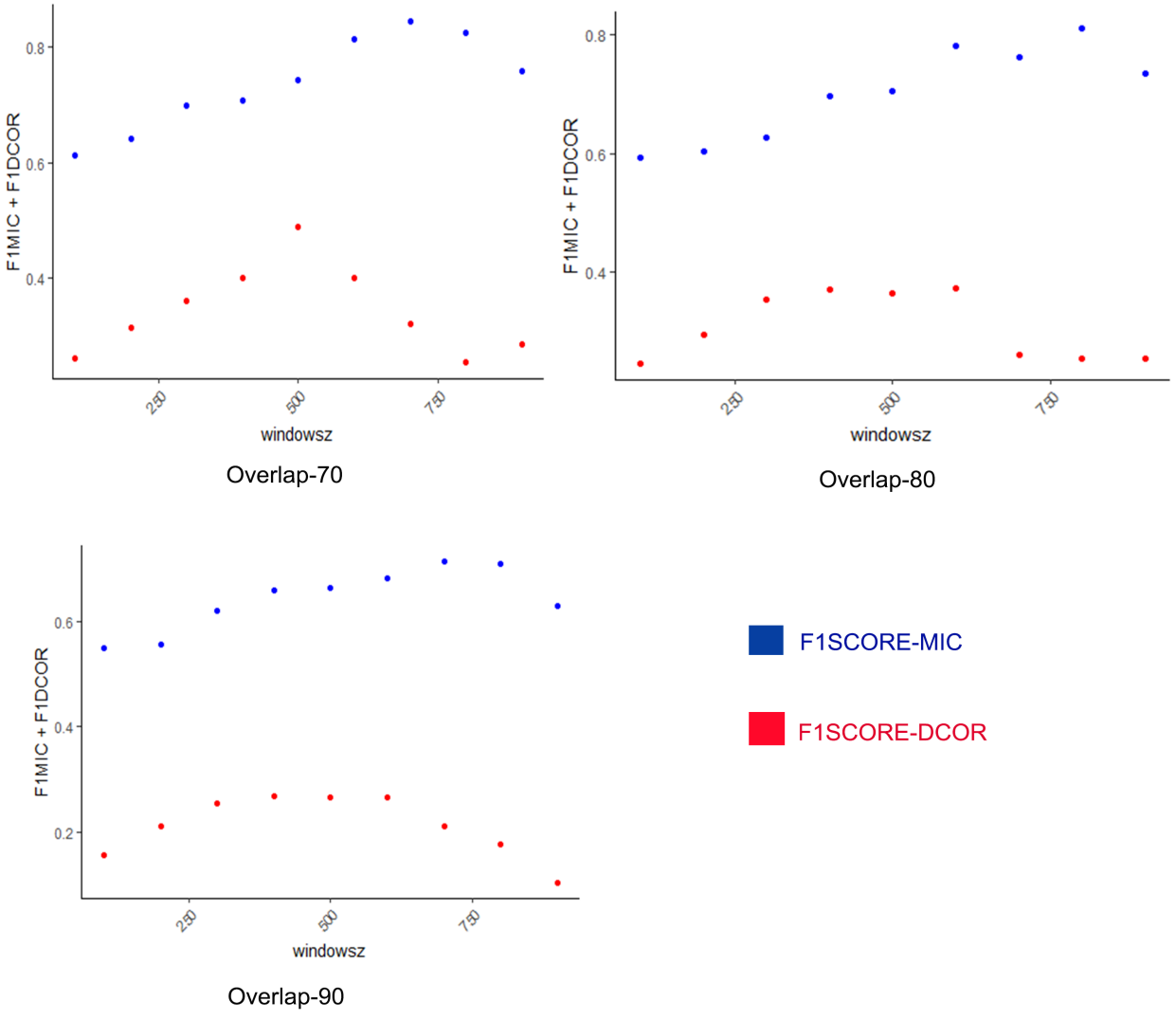


Figure 19: Experiments On DataSet 2

Above figure shows us three graphs for each overlap percent from 70 to 90 and Change Threshold being fixed to 1%. Overlap percent has clearly decreased the performance of both the measures. Similar results were coming in different configurations. This clearly show MIC is better than dCor.

Chapter 7

Conclusion and Future Work

Conclusion

In all the configuration of streaming environment executed, MIC performs way better than dCor. MIC is clearly better in picking up the change in relationship in the data. These measures have not been used on any other different datasets, so we can't say anything about that. But this method for anomaly detection can be applied on streaming environment, if we can develop a way to calculate MIC on streaming data. We can't say that this method can be used to capture any type of anomaly in any type of data. We can say that it can be used on that data, which does not have any contextual information related to it. As MIC and dCor both can be used to find dependency between random data vectors

Future Work

One possibility of future work here could be to develop an algorithm which can calculate MIC on streaming data. Than algorithm developed can be used to analyze the changing relationship between data in streaming environment. If the relationship changes a lot than we can say that an anomaly has occurred in our data.

Problem with this approach would be to calculate MIC incrementally on streaming data.

References:

[1] Szekely G, RizzoM, Bakirov N. Measuring and testing independence by correlation of distances, Ann Stat, 2007, vol. 35(pg. 2769-94).

[2] Reshef DN, Reshef YA, Finucane HK, Detecting novel associations in large data sets,*Science*, 2011, vol. 334
(pg. 1518-24).

[3] Shannon CE, WeaverW, The Mathematical Theory of communication, 1949 Urbana, ILUniversity of Illinois Press.

[4] A. A. Ding and Y. Li, "Copula correlation: An equitable dependence measure and extension of pearson's correlation," arXiv preprint arXiv: 1312.7214, 2013.