Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Outlier detection in circular regression model using minimum spanning tree method

Outlier detection in circular regression model using minimum spanning tree method

Nur Faraidah Muhammad Di[1], Siti Zanariah Satari[2] and Roslinazairimah Zakaria[3]

[1,2,3] Centre for Mathematical Sciences, Universiti Malaysia Pahang, 26300 Gambang, Pahang Darul Makmur, Malaysia.

[1,2]Corresponding author: nurfaraidah@gmail.com, zanariah@ump.edu.my

Abstract. The existence of outliers in circular-circular regression model can lead to many errors, for example in inferences and parameter estimations. Therefore, this study aims to develop new algorithms that can detect outliers by using minimum spanning tree method. The proposed method is examined via simulation study with different number of sample sizes and level of contaminations. Then, the performance of the proposed method was measured using "success" probability, masking effect, and swamping effect. The results revealed that the proposed method were performed well and able to detect all the outliers planted in various conditions.

1. Introduction

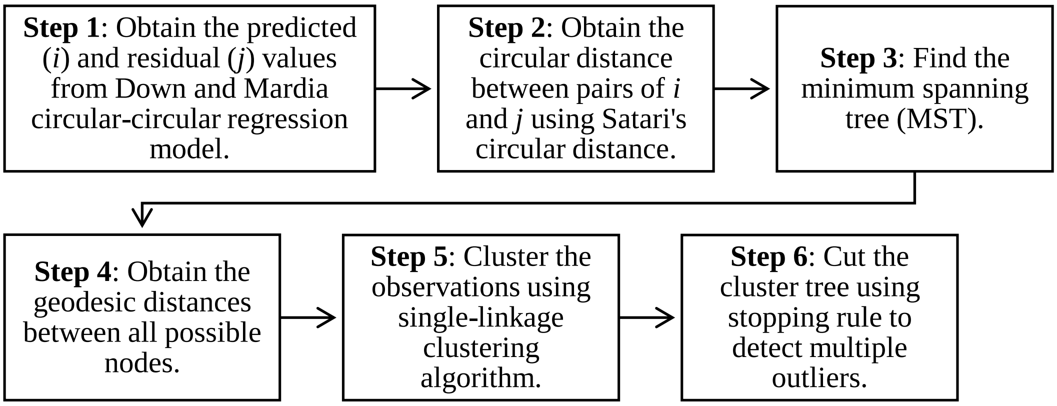
Circular data has been widely used in various areas such as in biology, geology, geography and medical. The existence of outlier in circular data is one of the most challenging tasks due to the high dimensionality of the data. Circular data is data that occurs around circle and measured in degree (0^o, 360^o] or radian $(0, 2\pi]$. The presence of outlier in circular data is measured by using certain circular distance to measure the distance of the observation from the mean direction. Previous studies discovered that cluster-based method in outlier detection produced good results in linear data set [1]. Hence, the motivation of this study is to propose a clustering-based method in detecting outliers in circular data, focusing on Down and Mardia circular-circular regression model [2]. Among the clustering methods, single-linkage is widely used as an outlier detection method since this method is sensitive to the presence of outliers [3].

In addition, [4] indicated that the performance of single-linkage algorithm can be improved by incorporating minimum spanning tree (MST) algorithms into the clustering. MST is defined as the tree connecting all nodes with minimum total weight. A spanning tree is a set of N-1 similarities that links all of the N objects in the data set together in a connected graph without any cycles. Meanwhile, [5] proposed a tree agglomerative hierarchical clustering (TAHC) method for the detection of clusters in MSTs and the result revealed that TAHC method presented better results on the artificial trees compared to the existing method such as Louvain algorithm, which suffers from a resolution limit. Besides that, [6] used MST based on k-partition clustering method and several similar distances, such as Euclidean, Euclidean Minimum Spanning Tree (EMST), Maximum Standard Deviation Reduction (MSDR) and Hierarchical EMST (HEMST) to detect clusters by maximizing the overall standard deviation reduction, without a given k value. Instead of clustering the data, another used of MST with clustering method is to detect outliers as shown by [1] in two-phase clustering process for outlier detection using modified k-mean and single-linkage with Euclidean distance.

In this study, a single-linkage MST namely S-MST that is based on Satari's circular distance introduced in [7] is proposed. This study aims to develop a modified clustering algorithm to detect multiple outliers in circular-circular regression model using single-linkage MST method. The proposed method is the extension of the clustering algorithm proposed by Satari [7, 8], namely S-SL that used single-linkage method and Euclidean distance to detect multiple outliers in circular-circular regression model.

2. Proposed Method

To develop S-MST, six stages are employed in order to detect multiple outlets in circular-circular regression model. Figure 1 displays the stages in the development of the S-MST method.



Step 1: Obtain the predicted (*i*) and residual (*j*) values from Down and Mardia circular-circular regression model.

Step 2: Obtain the circular distance between pairs of *i* and *j* using Satari's circular distance.

Step 3: Find the minimum spanning tree (MST).

Step 4: Obtain the geodesic distances between all possible nodes.

Step 5: Cluster the observations using single-linkage clustering algorithm.

Step 6: Cut the cluster tree using stopping rule to detect multiple outliers.

Figure 1. Stages in the development of the proposed method

2.1 Stage 1: Down and Mardia (DM) Model

The proposed method is applied on Down and Mardia (DM) [2] circular-circular regression model. The DM model is given by:

$$v = \beta + 2\tan^{-1}\left\{\omega \tan\frac{1}{2}(u-\alpha)\right\}, \tag{1}$$

where *u* and *v* are fixed independent angle and the dependent random angle respectively. The values of α and β are the angular location parameters and ω is the slope parameter with a closed interval of [-1,1]. The probability density function (pdf) and the angular error (*e*) are given in the equation (2) and (3), respectively

$$f(v) = \frac{1}{2\pi I_0(\kappa)}\exp\{\kappa\cos(v-u)\}, \tag{2}$$

$$e = v - \mu(u : \alpha, \beta, \omega). \tag{3}$$

The angular error (*e*) follows von Mises distribution with the mean direction of 0 $(\mu = 0)$ and nonnegative error concentration parameter, $(\kappa \geq 0)$.

2.2 Stage 2: Circular Distance

There are two reasonable measures of 'circular distance' defined by [9]. One of the circular distance is defined as the distance between observations that take the smaller of the two arc lengths between the points along the circumference,

$$d_{ij} = \pi - | \pi - | \theta_{ik} - \theta_{jk} || \tag{4}$$

where, d_{ij} is the distance between observation *i* and *j*, θ_{ik} is the value of the *k*th variable for the *i*th observation and θ_{jk} is the value of the *k*th variable for the *j*th observation.

In this study, we used Satari's distance [7] that derived from circular distance in equation 4 and City-block distance as a similarity measure. The Satari's distance is displayed as

$$d_{ij(Satari)} = \sum_{k=1}^{p} \left(\pi - \left| \pi - \left| \theta_{ik} - \theta_{jk} \right| \right| \right), \tag{5}$$

where *p* is the number of variables, θ_{ik} is the *k*th variable of *i*th observation and θ_{jk} is the *k*th variable of *j*th observation. By using Satari's distance, the distance matrix is calculated between all possible pairs of cluster, and then be used to calculate the minimum spanning tree.

2.3 Stage 3: Minimum Spanning Tree (MST)

In this study, we employed Kruskal's algorithm to compute MST from the similarity distance matrix. The Kruskal's algorithm is a greedy algorithm that finds the smallest weight edge that does not produce cycle in the MST. The weight here indicates the distance calculated in stage 2. To compute MST by using Kruskal's algorithm, the following steps are implied.

i. Sort all the edges in increasing order of their weight.
ii. Choose the smallest edge without cycle. If a cycle is formed, then discard.
iii. Repeat step (ii) until there are (*V*-1) edges in the spanning tree (*V* is the number of vertices).

2.4 Stage 4: Geodesic distance

The next step is to calculate the geodesic distance (matrix C) between all the possible pairs of nodes in a graph based on the MST calculated in stage 3 as an input for the clustering algorithm.

2.5 Stage 5: Single-linkage clustering algorithm

From the matrix C in stage 4, we clustered the new distance with single-linkage clustering algorithm defined as the smallest distance between two points in each cluster. This process will produce cluster tree, where the branches in the tree represent clusters.

2.6 Stage 6: Cut the tree

To cut the tree, we used stopping rule proposed by [7],

$$\bar{h} + 2.06 s_h \tag{6}$$

where $\bar{h}$ is the average heights of the cluster tree for all *N* – 1 clusters. At significant level of 0.05, the circular mean direction of cluster height, $\bar{h}$ is situated within $\pm 2.06 s_h$ and s_h is the circular standard deviation of cluster height.

3. Simulation Study

The simulation study uses two sample sizes (*n* = 30 and *n* = 100) for an independent circular variable (*u*) and circular error (*e*) from von Mises distribution to illustrate the efficiency of the proposed method for small and large sample sizes. The values of *u* are assumed fixed and generated from $VM(\pi/2, 2)$. The values of *e* are generated from $VM(0, \kappa)$ with the concentration parameter is set to be κ = 5 and κ = 10. Then, based on the generated random samples *u* and *e*, the values of the response variable (*v*) were calculated using Down and Mardia circular-circular regression model in [2] with fixed values of α = 1.5, β = 1.5 and ω = 0.5.

Three outliers were planted in each of data set with six contamination level, λ in the range of $0 \le \lambda \le 1$. The power performance of proposed method in detecting the outliers is measured using "success" probability (*pout*), masking error (*pmask*) and swamping error (*pswamp*) and carried out by simultaneous simulation using the same data set. The *pout* value is defined as the probability that all the planted outliers are successfully detected. Whereas, the *pmask* is the probability that the planted outliers are falsely detected as inliers and the *pswamp* is described as the probability of clean observations detected as outliers. The proposed method is compared with the S-SL clustering algorithm proposed by [7] to see the difference in terms of power performance in the detection of multiple outlier. A method is considered as good if the value of *pout* is approaching one, and the *pmask* and *pswamp* values are approaching zero.

4. Results and findings

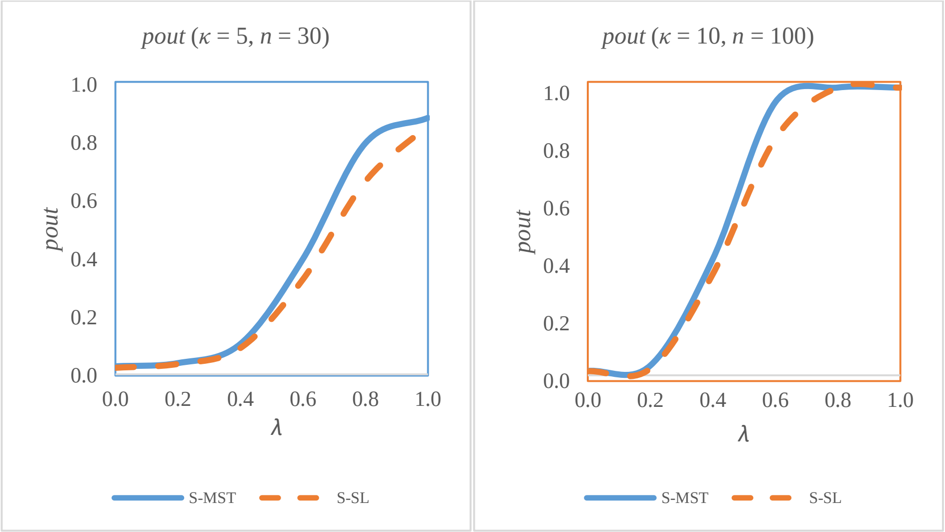
Table 1 presents the performance of the proposed method for various sample sizes (*n*), kappa value (κ) and level of contamination (λ) compared with S-SL clustering algorithm. In regards of the level of contamination, we can see that the values of *pout* increase significantly with an increase in λ.

Table 1. The *pout*, *pmask*, and *pswamp* values of proposed method

Performance measures			*pout*		*pmask*		*pswamp*	
Algorithm	λ	n	$\kappa = 5$	$\kappa = 10$	$\kappa = 5$	$\kappa = 10$	$\kappa = 5$	$\kappa = 10$
S-MST	0.0	30	0.027	0.041	0.901	0.903	0.092	0.100
		100	0.007	0.015	0.947	0.927	0.064	0.068
	0.2	30	0.038	0.056	0.850	0.826	0.087	0.094
		100	0.012	0.034	0.871	0.804	0.060	0.067
	0.4	30	0.104	0.268	0.712	0.562	0.087	0.083
		100	0.098	0.404	0.615	0.349	0.058	0.057
	0.6	30	0.396	0.755	0.416	0.173	0.085	0.078
		100	0.533	0.949	0.233	0.020	0.049	0.052
	0.8	30	0.794	0.942	0.128	0.029	0.080	0.076
		100	0.926	1.000	0.027	0.000	0.048	0.046
	1.0	30	0.881	0.963	0.058	0.013	0.075	0.075
		100	0.980	1.000	0.005	0.000	0.047	0.043
S-SL	0.0	30	0.022	0.031	0.953	0.965	0.112	0.109
		100	0.036	0.044	0.908	0.936	0.109	0.096
	0.2	30	0.035	0.065	0.888	0.904	0.105	0.095
		100	0.042	0.069	0.821	0.874	0.094	0.093
	0.4	30	0.090	0.285	0.755	0.411	0.095	0.091
		100	0.112	0.352	0.699	0.355	0.068	0.086
	0.6	30	0.326	0.696	0.519	0.035	0.089	0.087
		100	0.488	0.722	0.449	0.022	0.053	0.067
	0.8	30	0.663	0.921	0.182	0.031	0.086	0.064
		100	0.711	1.000	0.122	0.000	0.042	0.059
	1.0	30	0.855	0.954	0.062	0.018	0.086	0.057
		100	0.932	1.000	0.021	0.000	0.036	0.055

The values of *pout* are approaching one with higher level of λ. In contrast, with an increase in the level of λ, the values of *pmask* decreases significantly approaches to zero. Similarly, at high level of contamination, the values of *pswamp* are also decreased to zero.

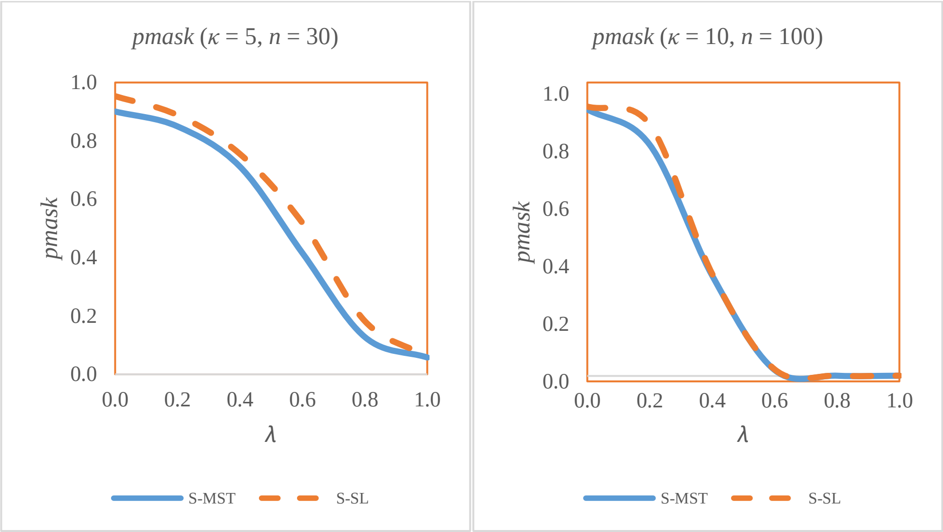
Figure 2 - 4 display the plot of *pout*, *pmask*, and *pswamp* versus level of contamination (λ) of proposed algorithms where the concentration parameter and sample sizes are taken from the smallest and largest values of κ and *n*, ($\kappa = 5$ and $n = 30$) and ($\kappa = 10$ and $n = 100$). The S-MST method is compared with S-SL clustering algorithm. From figure 2, the S-MST method is on par with S-SL clustering algorithm when contamination level is 0.4. However, when $\lambda \geq 0.6$, S-MST method is fastest approaching one. Besides, we can see that with high value of κ and *n*, the *pout* values approach to one as low as $\lambda = 0.8$. In conclusion, S-MST is the best algorithm compared to S-SL clustering algorithm which can detect outliers better with higher value of λ ($\lambda \geq 0.4$).



(a) (b)

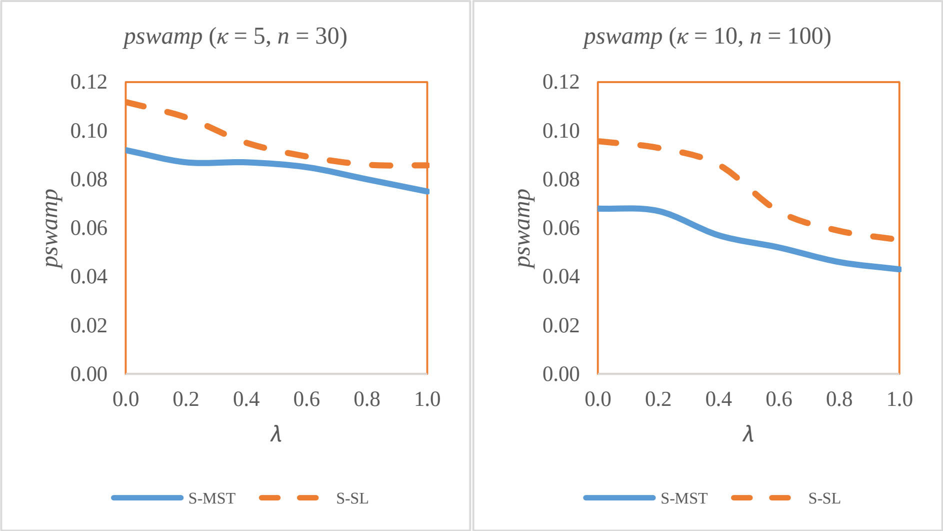
Figure 2. (a) Plot of success probability (*pout*) versus level of contamination with $\kappa = 5$ and $n = 30$. (b) Plot of success probability (*pout*) versus level of contamination with $\kappa = 10$ and $n = 100$.

Figure 3 displays the *pmask* values for all proposed algorithms. Noted that, the value of zero in *pmask* indicated that the method is free from masking error. Both methods approaches zero when $\lambda = 0.8$ with $\kappa = 10$ and $n = 100$. The S-MST outperformed S-SL clustering algorithm especially for high values of κ and *n*. In figure 4, the S-MST method produced small values of *pswamp* as lower as 0.042. The values of *pswamp* is decreasing significantly as the concentration parameter and sample size are increasing. S-MST method is less suffered from swamping error especially with higher number of κ and *n*.



(a) (b)

Figure 3. (a) Plot of masking error (*pmask*) versus level of contamination with $\kappa = 5$ and $n = 30$. (b) Plot of masking error (*pmask*) versus level of contamination with $\kappa = 10$ and $n = 100$.



(a) (b)

Figure 4. (a) Plot of swamping error (*pswamp*) versus level of contamination with $\kappa = 5$ and $n = 30$. (b) Plot of swamping error (*pswamp*) versus level of contamination with $\kappa = 10$ and $n = 100$.

5. Conclusion

In conclusion, based on the values of "success" probability (*pout*), masking error (*pmask*), and swamping error (*pswamp*), it is found that the S-MST performs very well on the simulated random data set with various conditions. Hence, we can conclude that with addition of MST to the single-linkage, it is proven able to improve the method's performance to detect outliers and simultaneously reduce the masking and swamping errors.

Acknowledgement

The Ministry of Higher Education Malaysia and Universiti Malaysia Pahang are acknowledged for the financial support received for this study (*FRGS*, RDU 160117).

References

[1] Jiang M F, Tseng S S and Su C M 2001 *Pattern recognition letters* **22** 691-700

[2] Down T and Mardia K V 2002 *Biometrika* **89**(3) 683-697

[3] Almeida J A S, Barbosa, L M S, Pais A A C C and Formosinho S J 2007 *Chemometrics and Intelligent Laboratory Systems* **87**(2) 208-217

[4] Gower J C and Ross G J S 1969 *Applied Statistics* 54-64

[5] Yu M, Hillebrand A, Tewarie P, Meier J, Dijk B, Mieghem P V and Stam C J 2015 *Chaos* **25** 023107

[6] Grygorash O, Zhou Y and Jorgensen Z 2006 *18th IEEE international conference on tools with artificial intelligence (ICTAI'o6)* 73-81

[7] Satari S Z 2015 Parameter Estimation and Outlier Detection for Some Types of Circular Model *PhD's Thesis* (University of Malaya)

[8] Di N F M and Satari S Z 2017 *AIP Conference Proceedings* 1842

[9] Jammalamadaka S R and SenGupta A 2001 *Topics in circular statistics* World Scientific Publishing 15