MJFAS
Malaysian Journal of Fundamental and Applied Sciences

RESEARCH ARTICLE

Simulation Study of the Test on Covariance Estimator for Outlier Detection in Multivariate Data with Mean and Covariance Shifts

Sharifah Sakinah Syed Abd Mutalib[a]*, Siti Zanariah Satari[b], Wan Nur Syahidah Wan Yusoff[b]

[a]Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia; [b]Centre for Mathematical Sciences, Universiti Malaysia Pahang Al-Sultan Abdullah, Lebuh Persiaran Tun Khalil Yaakob, 26300 Gambang, Pahang, Malaysia

Abstract Outlier detection in multivariate data is complex compared to univariate data, which can be done using graphical inspection. Outlier detection is also one of the common issues in multivariate analysis and has been applied to tax fraud detection and industrial food inspection. Outliers' studies are closely related to robust estimators of the sample mean and covariance matrix as these estimators are resistant toward outliers. The Test on Covariance (TOC) is a newly developed robust estimator for multivariate data. Until now, TOC's performance was investigated for two outlier scenarios by shifting the mean and covariance separately. TOC shows good results in both outlier scenarios and is found to be applicable in detecting outliers. In this study, the performance of TOC is investigated further in detecting outliers via simulation study for other outliers' scenarios by shifting the mean and covariance simultaneously. Other robust estimators; Fast Minimum Covariance Determinant (FMCD), Minimum Vector Variance (MVV), Covariance Matrix Equality (CME) and Index Set Equality (ISE) are used as a comparison. Various conditions of sample sizes, $n = 30, 50, 100$, number of variables, $p = 2, 3, 5$ and percentage of outliers, $\varepsilon = 5\%, 15\%$ are considered in the simulation study. The performance of all robust estimators is measured by probability to detect outliers *(pout),* masking error *(pmask) and* swamping error *(pswamp)*. Results present that the TOC can be the best robust estimator, give the same performance as other robust estimators in detecting outliers, and have a low masking error when outliers and inliers are far from each other. Moreover, TOC displays good results in low swamping errors for most cases which means TOC has a low probability of misclassifying inliers as outliers compared to other robust estimators. In conclusion, TOC is an applicable and promising approach for outlier detection in multivariate data and can be incorporated with other multivariate analyses.

Keywords: Test on Covariance, robust estimator, Mahalanobis distance, outliers, multivariate data.

***For correspondence:**
s.sakinah@umt.edu.my

Received: 25 June 2024
Accepted: 28 Feb. 2025

Introduction

Outliers are abnormal data points that deviate significantly from most of the observations in the dataset. While detecting outliers in univariate data is relatively straightforward and can often be done visually [1,2], identifying outliers in multivariate data is more complex. Outlier detection is also one of the common issues in multivariate analysis and has been applied to industrial food inspection [3] and tax fraud detection [4]. One standard method for detecting multivariate outliers is the Mahalanobis Distance (MD), which measures the distance of an observation from the data center while accounting for the data's overall shape [5]. The formula for MD is provided in equation (1),

$$d_i\left(\bar{\mathbf{x}}, \mathbf{S}\right) = \sqrt{\left(\mathbf{x}_i - \bar{\mathbf{x}}\right)' \mathbf{S}^{-1}\left(\mathbf{x}_i - \bar{\mathbf{x}}\right)}, \quad i = 1, 2, ..., n, \tag{1}$$

where $\bar{\mathbf{x}}$ is the sample mean, $\mathbf{S}$ is the covariance matrix and n is the sample size. However, MD's sample mean and covariance matrix have masking and swamping effects when the multivariate data contain outliers. Many studies have proposed using robust estimators of mean and covariance matrices as the robust estimators are resistant to outliers to solve masking and swamping problems [6,7].

Various robust estimators such as S, M, Method of Moments (MM), Minimum Volume Ellipsoid (MVE), Minimum Covariance Determinant (MCD) and Fast MCD (FMCD) estimators have been presented in the previous studies. Among these robust estimators, FMCD has been and still is widely used as shown in Salleh [8] and Mashuri *et al*. [9]. The Fast Minimum Covariance Determinant (FMCD) proposed by Rousseeuw & Van Driessen [10] is commonly used as FMCD has proved computationally efficient. The performance of FMCD uses clustered and shifted outliers and the combination of these outliers. However, FMCD still needs a lot of calculation and is time-consuming as FMCD uses covariance determinants in the last step of the algorithm [11].

Hence, Herwindiati *et al*. [6] took the initiative to propose robust estimators by using vector variance and called the new robust estimator Minimum Vector Variance (MVV). Regarding computation time, MVV is faster than FMCD but still lacking when the dimension or number of variables, p increases [12]. MVV has been tested to detect outliers for mean shift outlier scenarios only.

In 2013, Salleh [8] proposed two new robust estimators called Covariance Matrix Equality (CME) and Index Set Equality (ISE). The CME involves a comparison of two covariance matrices, element by element. Conversely, ISE is only a logical comparison of two index sets: old subset and new subset. ISE has been demonstrated to work excellently in computation time [11]. However, Salleh [8] did not test CME and ISE to detect outliers in multivariate data but applied CME and ISE to monitor process variability. According to Salleh [8], finding a condition for two covariance matrices to be equal can be further examined.

Hence, Abd Mutalib *et al*. [13] proposed a new robust estimator based on the idea of CME and ISE named Test on Covariance (TOC) and the performance of TOC is investigated via a simulation study. Abd Mutalib *et al*. [13] used one outlier scenario named mean shift in their study. Next, Abd Mutalib *et al*. [14] study the performance of TOC in two outlier scenarios: mean shift and covariance shift separately. Both studies found that TOC shows good results and a promising approach to detecting outliers for multivariate data. Abd Mutalib *et al*. [15] conducted a further study to investigate the performance of TOC in five real multivariate datasets from existing literature. Findings showed that TOC is a promising approach to detecting outliers in all datasets.

Therefore, in this study, further investigation is conducted to discover the performance of TOC in outlier scenarios when both the mean and covariance are shifted simultaneously. The performance of TOC will be analyzed and compared with FMCD, MVV, CME and ISE. The rest of the paper is organized as follows. The following section explains the materials and methods used in this study. Details about the simulation study and TOC are discussed in detail in this section. Then, the results and discussion of the simulation study are presented next. The last section presents the conclusion of this study.

Materials and Methods

Figure 1 shows the general procedure of this study's simulation study. Multivariate data is generated first according to the outlier scenario. Next, TOC and existing robust estimators are obtained. The simulation study is done 10,000 times after obtaining the robust estimators. Lastly, the performance of TOC is evaluated and compared with other robust estimators using performance measurement by Sebert *et al*. [16].

Generate the data according to outlier scenario.

↓

Obtain TOC and existing robust estimators (FMCD, MVV, CME and ISE).

↓

Simulation study is done to investigate the performance of the TOC.

↓

The performance of TOC is evaluated using performance measurement by Sebert *et al*. (1998) [16] and compared with FMCD, MVV, CME and ISE.

Figure 1. Simulation study general procedure

Test on Covariance Estimator

The TOC involving the equality test of variance-covariance structure is proposed by Abd Mutalib *et al*. [13]. The equality of two covariance structures is tested by using equation (2) with the hypothesis $H_0 : \Sigma_{old} = \Sigma_{new}$ versus $H_1 : \Sigma_{old} \neq \Sigma_{new}$,

$$u = \upsilon \left[\sum_{i=1}^{p} \left(\lambda_i - \ln \lambda_i \right) - p \right], \tag{2}$$

where $\upsilon = n - 1$ is the degrees of freedom, p is the number of variable and λ_i are the eigenvalue of $\Sigma_{new}\Sigma_{old}^{-1}$. Σ_{old} and Σ_{new} are obtained from H_{old} and H_{new} from the algorithm. The null hypothesis, H_0 is rejected if $u > \chi^2 \left[\alpha, \frac{1}{2} p \left(p+1 \right) \right]$ [17].

The TOC algorithm is similar to the FMCD algorithm, with a new procedure introduced for the stopping rule in Step 6. The FMCD algorithm is outlined below.

Step 1: Select an arbitrary subset H_{old} containing h different observations, where h is the smallest integer greater than or equal $\left(n + p + 1 \right)/2$, where p is the number of variables and n is the sample size.

Step 2: Compute the mean vector $\bar{X}_{H_{old}}$ and covariance matrix $S_{H_{old}}$ of all observations belonging to H_{old}.

Step 3: Compute $d_{H_{old}}^2 \left(i \right) = \left(X_i - \bar{X}_{H_{old}} \right)' S_{H_{old}}^{-1} \left(X_i - \bar{X}_{H_{old}} \right)$ for $i = 1, 2, \ldots, n$.

Step 4: Sort $d_{H_{old}}^2 \left(i \right)$ for $i = 1, 2, \ldots, n$ in increasing order $d_{H_{old}}^2 \left(\pi \left(1 \right) \right) \leq d_{H_{old}}^2 \left(\pi \left(2 \right) \right) \leq \ldots \leq d_{H_{old}}^2 \left(\pi \left(n \right) \right)$ where π is a permutation on $i = 1, 2, \ldots, n$.

Step 5: Define $H_{new} = \left\{ X_{\pi(1)}, X_{\pi(2)}, \ldots, X_{\pi(h)} \right\}$ and then calculate $\bar{X}_{H_{new}}$, $S_{H_{new}}$ and $d_{H_{new}}^2 \left(i \right)$ for $i = 1, 2, \ldots, n$.

Step 6FMCD: Stopping Rule. If $\det\left(S_{H_{new}} \right) = 0$, repeat Step 1 – Step 5. Otherwise, if $\det\left(S_{H_{new}} \right) < \det\left(S_{H_{old}} \right)$, let $H_{old} := H_{new}$, $\bar{X}_{H_{old}} := \bar{X}_{H_{new}}$ and $S_{H_{old}} := S_{H_{new}}$. Then go to Step 3. Otherwise, the process is stopped and $\det\left(S_{H_{new}} \right) = \det\left(S_{H_{old}} \right)$ is obtained.

Step 6 for TOC is given as follows,

Step 6TOC (Stopping Rule): If H_0 is rejected, calculate $\bar{X}_{H_{new}}$ and let $H_{old} := H_{new}$, $\bar{X}_{H_{old}} := \bar{X}_{H_{new}}$ and $S_{H_{old}} := S_{H_{new}}$. Then go to Step 3. Otherwise, the process is stopped.

The complete step of the TOC algorithm is shown in Figure 2.

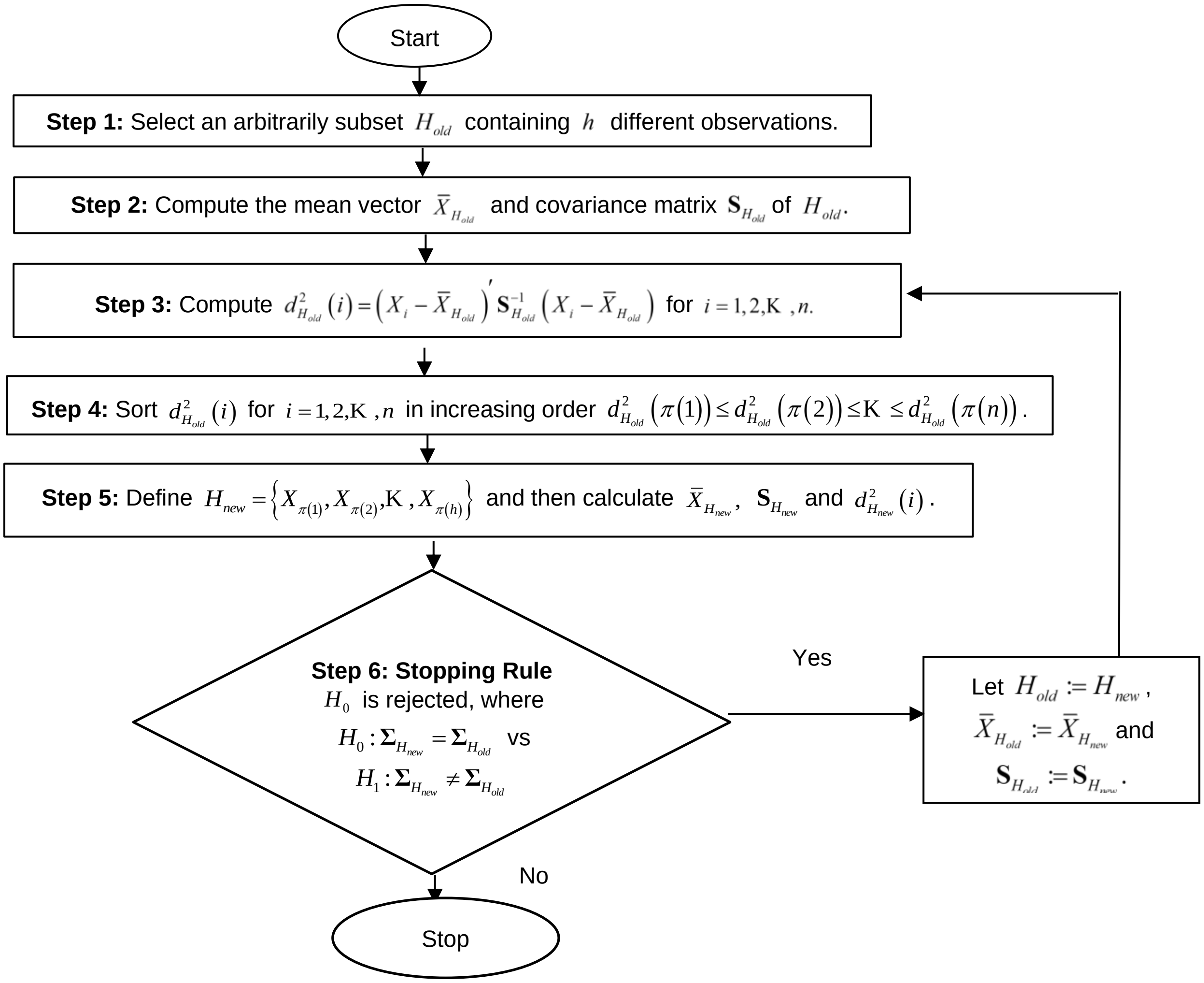

Figure 2. TOC Algorithm

In this study, the performance of TOC in detecting outliers in multivariate data is compared with FMCD, MVV, CME and ISE. The MVV, CME, and ISE also used the FMCD algorithm but differed in Step 6. The following are Step 6 for MVV, CME and ISE.

Step 6MVV: If $Tr\left(S^2_{H_{new}}\right) = 0$, repeat Step 1 – Step 5. Otherwise, if $Tr\left(S^2_{H_{new}}\right) \ne Tr\left(S^2_{H_{old}}\right)$, let $H_{old} := H_{new}$, $\bar{X}_{H_{old}} := \bar{X}_{H_{new}}$ and $S_{H_{old}} := S_{H_{new}}$. Then go to Step 3. Otherwise, the process is stopped and $Tr\left(S^2_{H_{new}}\right) = Tr\left(S^2_{H_{old}}\right)$ is obtained.

Step 6CME: If $\sqrt{Tr\left(S_{H_{new}} - S_{H_{old}}\right)^2} \ne 0, \quad I_{new} \ne I_{old}$

Step 6ISE: If $I_{new} \ne I_{old}$, let $H_{old} := H_{new}$, calculate $\bar{X}_{H_{new}}$ and let $H_{old} := H_{new}$, $\bar{X}_{H_{old}} := \bar{X}_{H_{new}}$ and $S_{H_{old}} := S_{H_{new}}$. Then go to Step 3. Otherwise, the process is stopped.

Simulation Study

A simulation study is performed by generating multivariate data from the following mixture *p*-variate normal distributions [5,6,18–20] and is given as in equation (3).

$$(1-\varepsilon)N_p\left(\vec{\mu}_0,\Sigma_0\right)+\varepsilon N_p\left(\lambda\vec{\mu}_1,\delta\Sigma_1\right), \tag{3}$$

where $\Sigma_0=\Sigma_1=I_p$, $\vec{\mu}_0=(0\ 0\ldots 0)'$ and $\vec{\mu}_1=(1\ 1\ldots 1)'$ is of dimension p. Inliers are generated from $N_p\left(\vec{\mu}_0,\Sigma_0\right)$ and outliers are generated from $N_p\left(\lambda\vec{\mu}_1,\delta\Sigma_1\right)$. The separation between outliers and inliers is determined by the values of λ and δ where both values are mean shift and covariance shift values, respectively. In this study, both λ and δ are shifted simultaneously. The outlier scenario in this study is different from the study done by Abd Mutalib *et al*. [13], Abd Mutalib *et al*. [14], and Abd Mutalib *et al*. [15], where mean (λ) or covariance (δ) are shifted separately.

The simulation study has been conducted for different values of the sample sizes, $n=30,50,100$ and different number of variables, $p=2,3,5$. The percentage of outliers is set as $\varepsilon=5\%,15\%$, the distance of outliers by the shifting mean is $\lambda=2,4$, and the distance of outliers by shifting the covariance is $\delta=2,10$.

Performance Measurement

The steps to identify outliers are given below. Robust mean and covariance matrix obtained from FMCD, MVV, CME, ISE and TOC will replace $\bar{\mathbf{x}}$ and $\mathbf{S}$ in Step 1.

Step 1: Compute the distance $d^2(\mathbf{x}_i)=\sqrt{(\mathbf{x}_i-\bar{\mathbf{x}})^t\mathbf{S}^{-1}(\mathbf{x}_i-\bar{\mathbf{x}})}$ for $i=1,2,\ldots n$, where $\bar{\mathbf{x}}$ and $\mathbf{S}$ are the robust mean and covariance matrix of FMCD, MVV, CME, ISE and TOC.

Step 2: Use the cut-off value $\sqrt{\chi^2_{p,0.975}}$ to detect outliers. If $d(\mathbf{x}_i)>\sqrt{\chi^2_{p,0.975}}$, $\mathbf{x}_i$ is an outlier.

The simulation study was done using the R statistical package. The simulation is run repeatedly 10000 times. The performance of the five robust estimators is measured by using three different measurements, which are the probability of detecting outliers (*pout*), masking error (*pmask*) and swamping error (*pswamp*) given as follows [16,21]. Equations (4) – (6) are the formulas for the performance measures:

$$pout=\frac{\text{"}success\text{"}}{s}, \tag{4}$$

$$pmask=\frac{\text{"}failure\text{"}}{(out)(s)}, \tag{5}$$

$$pswamp=\frac{\text{"}false\text{"}}{(n-out)(s)}, \tag{6}$$

where "*success*" is the number of data sets that the robust estimators successfully identified all the outliers, "*failure*" is the number of outliers in all data sets that are detected as inliers and "*false*" is the number of inliers in all data sets that falsely detected as outliers. Meanwhile, s is the total number of simulations, out is the number of outliers and n is the sample size. The *pout*, *pmask* and *pswamp* values will be between 0 and 1. The best robust estimator will show the highest *pout* value when the value approaches 1 and the lowest value of *pmask* and *pswamp* when the value approaches 0 [22].

Results and Discussion

Results of the simulation study are presented in Table 1 to Table 3 and illustrated in Figure 3 to Figure 5. Table 1 shows results for the probability of detecting outliers (*pout*) by all robust estimators. The best results are bold to indicate the best robust estimator's performance. From Table 1, the *pout* values of each δ for all robust estimators increase when values of λ increases for any fixed values of n, p and ε. It shows that all robust estimators have better performance detecting outliers when the values of λ increase. From Table 1, the *pout* values for all robust estimators are 1.0000 when $p=5,\varepsilon=5\%$ for all values of n. These results indicate that all robust estimators successfully detected all outliers for these cases. Most of the *pout* values are more than 0.9000 when $\lambda = 4, \varepsilon = 5\%$ for any fixed values of δ, n and p. As we can see from Table 1, the *pout* values for all robust estimators are decreasing as the percentage of outliers, ε increasing except for one case when $n = 30, p = 2, \delta = 2, \lambda = 2$. This might be because as the number of outliers increases, it becomes harder for all robust estimators to detect outliers. From Table 1, TOC show quite good results and becomes the best estimator in some cases, for example, when $n = 30, p = 3, \delta = 2, \lambda = 4$ and $\varepsilon = 15\%$. Alternatively, the results for *pout* can be represented in Figure 3. For illustrative purposes, only the graph for $n = 100$ and $p = 5$ were chosen. From Figure 3, all robust estimators approach 1 as the values of λ increasing and as we can see in this case, *pout* values approach 1 faster for $\delta = 10$ than $\delta = 2$.

Table 1. The performance measure using "success" probability (*pout*) of different robust estimators

n	p	δ	λ	$\varepsilon = 5\%$					$\varepsilon = 15\%$				
				FMCD	MVV	CME	ISE	TOC	FMCD	MVV	CME	ISE	TOC
30	2	2	2	**0.6636**	0.6252	**0.6636**	0.6252	**0.6636**	0.4809	**0.4919**	**0.4919**	0.4809	**0.4919**
			4	0.9745	0.9793	0.9793	0.9793	**0.9846**	0.9933	**0.9940**	**0.9940**	**0.9940**	0.9918
		10	2	**0.7600**	0.7585	0.7535	0.7500	0.7467	**0.5233**	**0.5233**	**0.5233**	**0.5233**	0.4932
			4	**0.9029**	**0.9029**	**0.9029**	**0.9029**	**0.9029**	**0.6975**	**0.6975**	0.6832	**0.6975**	**0.6975**
	3	2	2	**0.8682**	**0.8682**	**0.8682**	**0.8682**	0.7776	0.2224	**0.2298**	0.2224	0.2224	0.2224
			4	**0.9999**	**0.9999**	**0.9999**	**0.9999**	0.9998	0.9851	0.9851	0.9829	0.9829	**0.9852**
		10	2	**0.9051**	0.897	0.892	0.8962	0.9009	**0.7193**	**0.7193**	0.7084	**0.7193**	0.7045
			4	**0.9704**	**0.9704**	**0.9704**	0.9702	0.9699	**0.8900**	**0.8900**	**0.8900**	0.8895	0.8895
	5	2	2	0.9582	0.9588	**0.9603**	0.9506	0.9592	0.8870	0.8870	0.8650	**0.9096**	0.8870
			4	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	0.9990	**1.0000**
		10	2	0.9818	0.9803	0.9824	**0.9829**	0.9822	0.9504	0.9508	0.9440	**0.9542**	**0.9542**
			4	0.9984	0.9983	0.9986	**0.9987**	0.9984	**0.9940**	0.9936	0.9936	0.9936	**0.9940**
50	2	2	2	0.6408	0.6286	**0.6600**	**0.6600**	**0.6600**	**0.1625**	0.1378	0.1378	**0.1625**	0.1378
			4	**0.9951**	**0.9951**	**0.9951**	0.9936	0.9906	0.9822	0.9822	0.9822	**0.9844**	0.9822
		10	2	**0.6632**	0.6621	**0.6632**	0.6621	**0.6632**	**0.2620**	**0.2620**	**0.2620**	**0.2620**	**0.2620**
			4	**0.8382**	**0.8382**	0.8239	0.8295	0.8239	**0.6736**	**0.6736**	**0.6736**	**0.6736**	0.6437
	3	2	2	**0.8076**	**0.8076**	**0.8076**	0.6765	0.7495	0.3538	**0.4212**	0.3538	0.3538	0.3538
			4	0.9967	0.9967	0.9964	0.9964	**0.9972**	**0.9961**	0.9960	0.9960	0.9960	0.9933
		10	2	0.8250	**0.8300**	**0.8300**	0.8250	**0.8300**	0.5998	0.5998	**0.6030**	0.5998	0.5877
			4	0.9554	**0.9558**	0.9554	0.9554	**0.9558**	0.8756	**0.8781**	**0.8781**	**0.8781**	**0.8781**
	5	2	2	0.9399	**0.9460**	0.9410	**0.9460**	0.9245	0.5916	**0.8020**	0.5913	0.2384	0.5916
			4	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**0.9999**	**0.9999**	**0.9999**	**0.9999**	**0.9999**
		10	2	0.9590	**0.9624**	0.9580	0.9593	0.9607	0.8746	**0.8771**	0.8683	0.8739	0.8726
			4	0.9974	0.9980	0.9975	**0.9981**	0.9977	**0.9877**	0.9866	0.9865	0.9872	0.9866
100	2	2	2	0.3431	**0.3641**	0.3431	0.3431	0.3181	**0.0484**	**0.0484**	**0.0484**	**0.0484**	0.0203
			4	**0.9864**	**0.9864**	**0.9864**	**0.9864**	0.9795	**0.9528**	**0.9528**	**0.9528**	**0.9528**	0.9473
		10	2	0.5675	0.5675	**0.5705**	**0.5705**	0.5663	0.1049	**0.1067**	0.1049	**0.1067**	0.0881
			4	**0.8035**	**0.8035**	**0.8035**	**0.8035**	0.7948	**0.4433**	0.4375	0.4375	0.4375	0.4185
	3	2	2	**0.5550**	**0.5550**	**0.5550**	**0.5550**	0.5436	**0.0176**	**0.0176**	**0.0176**	**0.0176**	0.0161
			4	**0.9991**	**0.9991**	**0.9991**	**0.9991**	0.9987	**0.9895**	**0.9895**	**0.9895**	**0.9895**	**0.9895**
		10	2	**0.7645**	0.7630	**0.7645**	0.7643	0.7582	**0.3112**	**0.3112**	**0.3112**	**0.3112**	**0.3112**
			4	**0.9381**	0.9376	**0.9381**	0.9365	0.9310	0.8093	0.8093	**0.8094**	0.8093	0.8093
	5	2	2	0.8326	**0.8436**	0.8326	0.8326	0.7972	**0.4321**	**0.4321**	**0.4321**	0.4286	0.3333
			4	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**
		10	2	0.9317	0.9326	**0.9329**	0.9326	0.9317	0.7987	0.7984	**0.8053**	0.8046	0.7864
			4	**0.9944**	0.9942	**0.9944**	**0.9944**	**0.9944**	0.9784	0.9784	0.9784	**0.9793**	0.9784

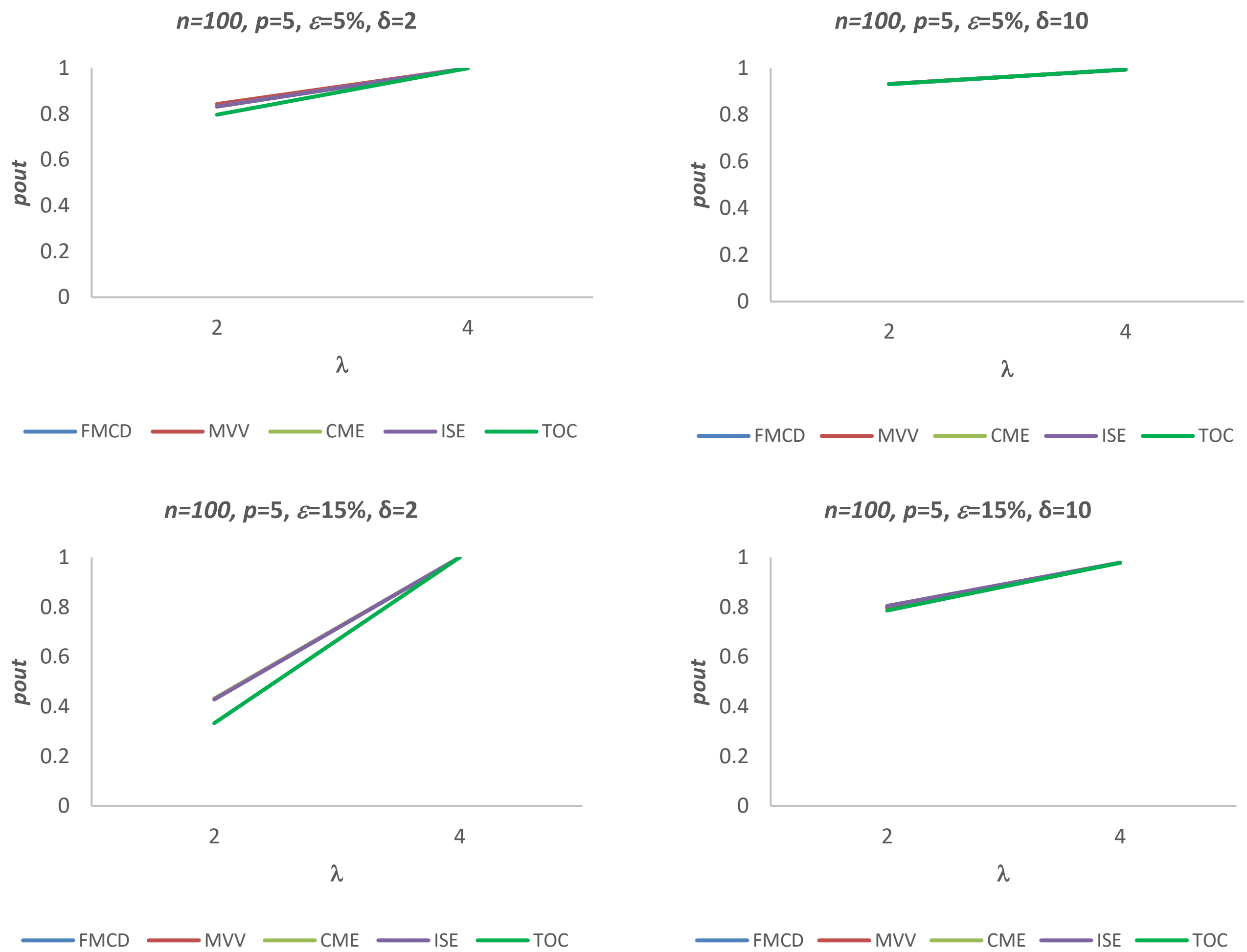

Figure 3. Plot of "success" probability (*pout*) versus distance of outliers and inliers (λ)

Table 2 shows the results for the probability of misclassifying outliers as inliers (*pmask*). The *pmask* values of each δ for all robust estimators decrease when the values of λ increases for any fixed values of n, p and ε. From Table 2, the *pmask* values for all robust estimators are 0.0000 when $p = 5$ for any fixed values of n and ε. These results indicate that all robust estimators do not misclassify outliers as inliers for these cases. Most of the *pmask* values from Table 2 are less than 0.1000, which indicates all robust estimators show good performance in not misclassifying outliers as inliers in most cases. The highest *pmask* value is 0.2596 and is recorded by FMCD, CME, ISE and TOC when $n = 30, p = 3, \delta = 2, \lambda = 2$ and $\varepsilon = 15\%$. The bold number is to show the best results for *pmask*. As seen from Table 2, TOC shows good results and becomes the best estimator in some cases, such as when $n = 50, p = 3, \delta = 2, \lambda = 4$ and $\varepsilon = 5\%$. Figure 4 shows a graphical representation for *pmask* values. We choose graph for $n = 100$ and $p = 5$ for illustrative purposes. From Figure 4, all robust estimators approach 0 as the values of λ increasing. This shows that all robust estimators have low probability of misclassifying outliers as inliers when the distance of outliers increases by shifting the mean. From Figure 4 we also can see that *pmask* values approach 0 faster as the values of λ increasing for $\delta = 10$ than $\delta = 2$. All robust estimators have lower masking errors as the values of λ and δ increasing.

Table 2. The performance measure using masking error (*pmask*) of different robust estimators

n	p	δ	λ	$\varepsilon=5\%$ FMCD	MVV	CME	ISE	TOC	$\varepsilon=15\%$ FMCD	MVV	CME	ISE	TOC
30	2	2	2	**0.1868**	0.2104	**0.1868**	0.2104	**0.1868**	0.1358	**0.1322**	**0.1322**	0.1358	**0.1322**
			4	0.0128	0.0104	0.0104	0.0104	**0.0078**	0.0013	**0.0012**	**0.0012**	**0.0012**	0.0016
		10	2	**0.1280**	0.1292	0.1317	0.1337	0.1358	**0.1218**	**0.1218**	**0.1218**	**0.1218**	0.1340
			4	**0.0496**	**0.0496**	**0.0496**	**0.0496**	**0.0496**	**0.0689**	**0.0689**	0.0731	**0.0689**	**0.0689**
	3	2	2	**0.0679**	**0.0679**	**0.0679**	**0.0679**	0.1181	0.2596	**0.2542**	0.2596	0.2596	0.2596
			4	**0.0000**	**0.0000**	**0.0000**	**0.0000**	0.0001	**0.0030**	**0.0030**	0.0034	0.0034	**0.0030**
		10	2	**0.0489**	0.0536	0.0534	0.0534	0.0511	**0.0646**	**0.0646**	0.0650	**0.0646**	0.0691
			4	**0.0149**	**0.0149**	**0.0149**	0.0150	0.0152	**0.0230**	**0.0230**	**0.0230**	0.0231	0.0231
	5	2	2	0.0211	0.0209	**0.0200**	0.0251	0.0205	0.0239	0.0239	0.0289	**0.0189**	0.0239
			4	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		10	2	0.0092	0.0100	0.0089	**0.0086**	0.0090	0.0102	0.0101	0.0115	**0.0094**	**0.0094**
			4	0.0008	0.0008	0.0007	**0.0006**	0.0008	**0.0012**	0.0013	0.0013	0.0013	**0.0012**
50	2	2	2	0.1376	0.1437	**0.1291**	**0.1291**	**0.1291**	**0.2046**	0.2200	0.2200	**0.2046**	0.2200
			4	**0.0016**	**0.0016**	**0.0016**	0.0021	0.0032	0.0022	0.0022	0.0022	**0.0020**	0.0022
		10	2	**0.1273**	0.1279	**0.1273**	0.1279	**0.1273**	**0.1530**	**0.1530**	**0.1530**	**0.1530**	**0.1530**
			4	**0.0572**	**0.0572**	0.0624	0.0606	0.0624	**0.0479**	**0.0479**	**0.0479**	**0.0479**	0.0536
	3	2	2	**0.0689**	**0.0689**	**0.0689**	0.1222	0.0921	0.1218	**0.1024**	0.1218	0.1218	0.1218
			4	0.0011	0.0011	0.0012	0.0012	**0.0009**	0.0005	0.0005	0.0005	0.0005	**0.0001**
		10	2	0.0622	**0.0602**	**0.0602**	0.0622	**0.0602**	0.0620	0.0620	**0.0614**	0.0620	0.0642
			4	0.0150	**0.0149**	0.0150	0.0150	**0.0149**	0.0166	**0.0163**	**0.0163**	**0.0163**	**0.0163**
	5	2	2	0.0204	**0.0182**	0.0200	**0.0182**	0.0258	0.0633	**0.0273**	0.0633	0.1634	0.0633
			4	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		10	2	0.0140	**0.0127**	0.0143	0.0138	0.0134	0.0166	**0.0161**	0.0175	0.0166	0.0168
			4	0.0009	0.0007	0.0008	**0.0006**	0.0008	**0.0015**	0.0017	0.0017	0.0016	0.0017
100	2	2	2	0.1899	**0.1808**	0.1899	0.1899	0.2016	**0.1793**	**0.1793**	**0.1793**	**0.1793**	0.2263
			4	**0.0028**	**0.0028**	**0.0028**	**0.0028**	0.0042	**0.0032**	**0.0032**	**0.0032**	**0.0032**	0.0036
		10	2	0.1060	0.1060	**0.1050**	**0.1050**	0.1064	0.1432	**0.1420**	0.1432	**0.1420**	0.1523
			4	**0.0425**	**0.0425**	**0.0425**	**0.0425**	0.0447	**0.0528**	0.0537	0.0537	0.0537	0.0563
	3	2	2	**0.1114**	**0.1114**	**0.1114**	**0.1114**	0.1148	**0.2356**	**0.2356**	**0.2356**	**0.2356**	0.2408
			4	**0.0002**	**0.0002**	**0.0002**	**0.0002**	0.0003	**0.0007**	**0.0007**	**0.0007**	**0.0007**	**0.0007**
		10	2	0.0527	0.0529	0.0527	**0.0526**	0.0542	**0.0750**	**0.0750**	**0.0750**	**0.0750**	**0.0750**
			4	**0.0127**	0.0128	**0.0127**	0.0131	0.0142	0.0140	0.0140	**0.0139**	0.0140	0.0140
	5	2	2	0.0359	**0.0333**	0.0359	0.0359	0.0443	**0.0540**	**0.0540**	**0.0540**	0.0545	0.0698
			4	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		10	2	0.0143	**0.0141**	**0.0141**	**0.0141**	0.0143	0.0148	0.0148	**0.0143**	**0.0143**	0.0159
			4	**0.0011**	0.0012	**0.0011**	**0.0011**	**0.0011**	0.0015	0.0015	0.0015	**0.0014**	0.0015

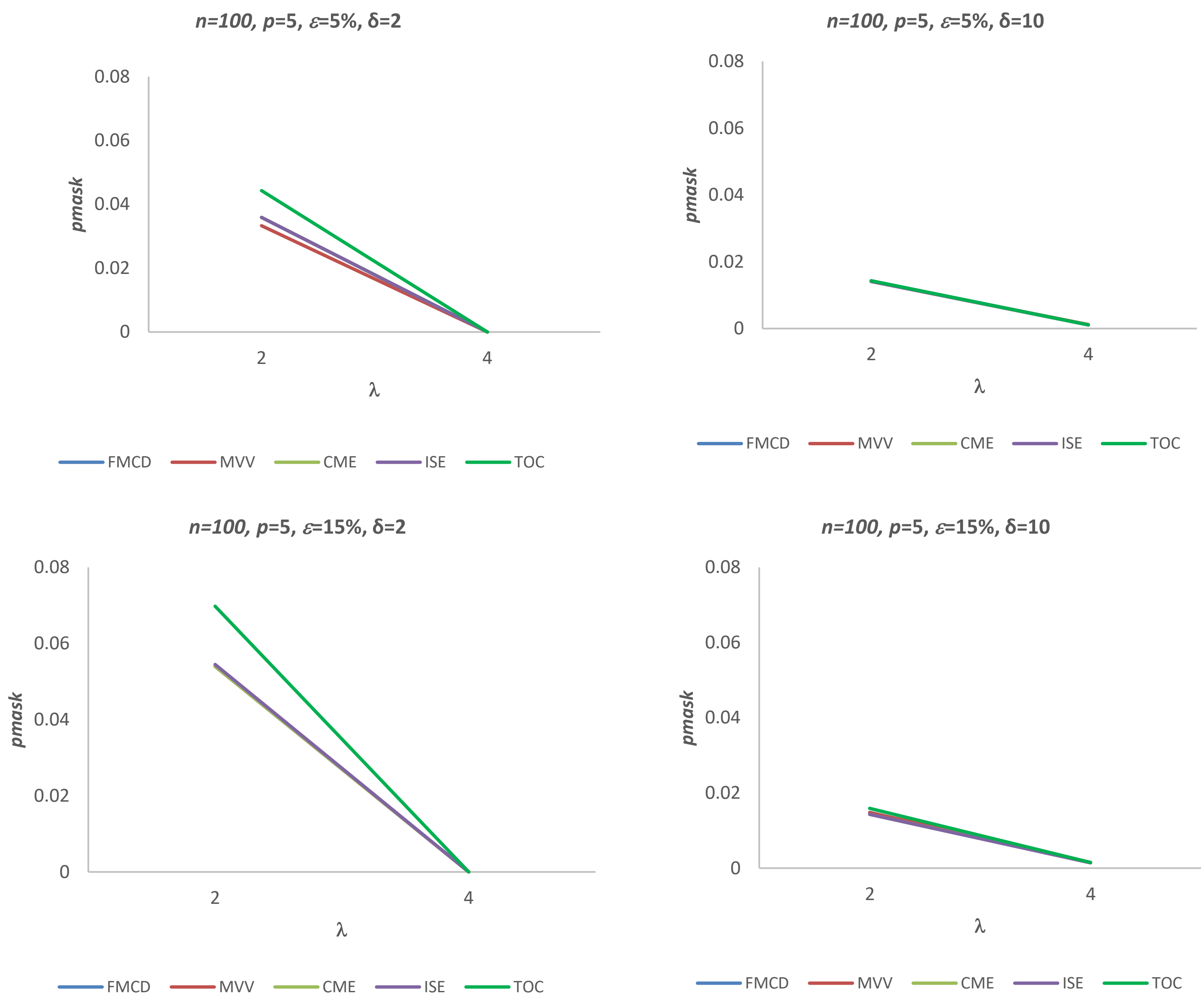

Figure 4. Plot of masking error (*pmask*) versus distance of outliers and inliers (λ)

Table 3 shows results for the probability of misclassifying inliers as outliers (*pswamp*). For *pswamp* values, there is no robust estimator obtained *pswamp* value 0.0000 for n and *p*, which indicated that all robust estimators misclassify inliers as outliers for all cases. The *pswamp* values also show a similar pattern as *pmask* values. The *pswamp* values of each δ for all robust estimators decrease when values of λ increases for any fixed values of n, p and ε. Overall, *pswamp* values for all robust estimators less than 0.3100 indicate that all robust estimators have a low probability of misclassifying inliers as outliers. Figure 5 shows all robust estimators approaching 0 as the values of λ increasing. This indicates that all robust estimators have a low probability of misclassifying inliers as outliers when the distance of outliers increases by shifting the mean.

Table 3. The performance measure using swamping error (*pswamp*) of different robust estimators

n	p	δ	λ	$\varepsilon = 5\%$					$\varepsilon = 15\%$				
				FMCD	MVV	CME	ISE	TOC	FMCD	MVV	CME	ISE	TOC
30	2	2	2	0.1464	**0.1211**	0.1464	**0.1211**	0.1464	**0.1817**	0.1866	0.1866	**0.1817**	0.1866
			4	0.1075	**0.0915**	**0.0915**	**0.0915**	0.094	0.1295	0.1262	0.1262	0.1262	**0.0903**
		10	2	0.1755	0.219	0.1766	0.1839	**0.1672**	0.1713	0.1713	0.1713	0.1713	**0.1320**
			4	**0.1592**	**0.1592**	**0.1592**	**0.1592**	**0.1592**	0.0796	0.0796	**0.0703**	0.0796	0.0796
	3	2	2	0.2253	0.2253	0.2253	0.2253	**0.1267**	**0.0911**	0.1017	**0.0911**	**0.0911**	**0.0911**
			4	0.1521	0.1521	0.1521	0.1521	**0.114**	0.0529	0.0529	0.0592	0.0592	**0.0469**
		10	2	0.1791	0.1741	0.1790	0.1790	**0.1672**	0.1620	0.1620	0.1833	0.1620	**0.1176**
			4	0.1059	0.1059	0.1059	0.1390	**0.0922**	0.1179	0.1179	0.1179	**0.0859**	**0.0859**
	5	2	2	0.2705	0.3025	0.2524	**0.2132**	0.2356	0.1936	0.1936	**0.1919**	0.2029	0.1936
			4	0.2040	0.2282	0.2282	**0.1985**	0.2133	**0.1574**	0.1785	0.1883	0.1804	0.1883
		10	2	0.3098	0.2779	0.3060	0.2720	**0.2406**	0.2266	0.2347	**0.2234**	0.2363	0.2363
			4	0.3039	0.2574	0.2584	0.2431	**0.2275**	0.1990	**0.1803**	**0.1803**	**0.1803**	0.1990
50	2	2	2	0.2463	0.2511	**0.2402**	**0.2402**	**0.2402**	**0.1401**	0.1582	0.1582	**0.1401**	0.1582
			4	0.1756	0.1756	0.1756	0.1490	**0.1419**	**0.1222**	**0.1222**	**0.1222**	0.1297	**0.1222**
		10	2	**0.1582**	0.1685	**0.1582**	0.1685	**0.1582**	**0.1046**	**0.1046**	**0.1046**	**0.1046**	**0.1046**
			4	**0.1104**	**0.1104**	0.1249	0.1315	0.1249	0.0990	0.0990	0.0990	0.0990	**0.0954**
	3	2	2	0.2054	0.2054	0.2054	0.2043	**0.1743**	0.1900	**0.1849**	0.1900	0.1900	0.1900
			4	0.1375	0.1375	0.1367	0.1367	**0.1172**	0.1433	0.1412	0.1412	0.1412	**0.1398**
		10	2	**0.1634**	0.1716	0.1716	**0.1634**	0.1716	0.1311	0.1311	0.1245	0.1311	**0.1197**
			4	**0.1553**	0.1627	**0.1553**	**0.1553**	0.1627	**0.0982**	0.1008	0.1008	0.1008	0.1008
	5	2	2	0.2433	0.2456	0.2370	0.2456	**0.2185**	0.1354	**0.1346**	0.1354	0.1653	0.1354
			4	0.2252	0.2393	0.2252	0.2252	**0.2135**	**0.0925**	0.0998	0.0998	0.0950	**0.0925**
		10	2	**0.1754**	0.1972	0.1772	0.1786	0.1813	0.1156	0.1155	**0.1142**	0.1281	0.1145
			4	0.1731	0.1655	0.1732	0.1731	**0.1651**	0.1030	0.0991	0.1117	**0.0942**	0.0991
100	2	2	2	0.1672	0.1677	0.1672	0.1672	**0.1544**	0.1293	0.1293	0.1293	0.1293	**0.1162**
			4	0.1175	0.1175	0.1175	0.1175	**0.1123**	0.0972	0.0972	0.0972	0.0972	**0.0930**
		10	2	0.1980	0.1980	0.1917	0.1917	**0.1886**	0.1272	0.1305	0.1272	0.1305	**0.1080**
			4	0.1693	0.1693	0.1693	0.1693	**0.1649**	0.0834	0.0860	0.0860	0.0860	**0.0795**
	3	2	2	0.1144	0.1144	0.1144	0.1144	**0.1101**	**0.0787**	**0.0787**	**0.0787**	**0.0787**	0.0795
			4	0.0956	0.0956	0.0956	0.0956	**0.0947**	**0.0693**	**0.0693**	**0.0693**	**0.0693**	0.0695
		10	2	0.1420	0.1391	0.1420	0.1402	**0.1363**	**0.1300**	**0.1300**	**0.1300**	**0.1300**	**0.1300**
			4	0.1112	**0.1081**	0.1112	0.1094	0.1101	**0.0973**	**0.0973**	0.0985	**0.0973**	**0.0973**
	5	2	2	0.1370	0.1380	0.1370	0.1370	**0.1359**	0.1041	0.1041	0.1041	0.0998	**0.0968**
			4	0.1270	**0.1261**	0.1270	0.1266	0.1270	0.0886	0.0900	0.0900	0.0900	**0.0832**
		10	2	**0.1286**	0.1300	**0.1286**	0.1300	**0.1286**	0.1145	0.1172	0.1243	0.1205	**0.1022**
			4	**0.1207**	0.1223	**0.1207**	**0.1207**	**0.1207**	**0.0861**	**0.0861**	**0.0861**	0.0963	**0.0861**

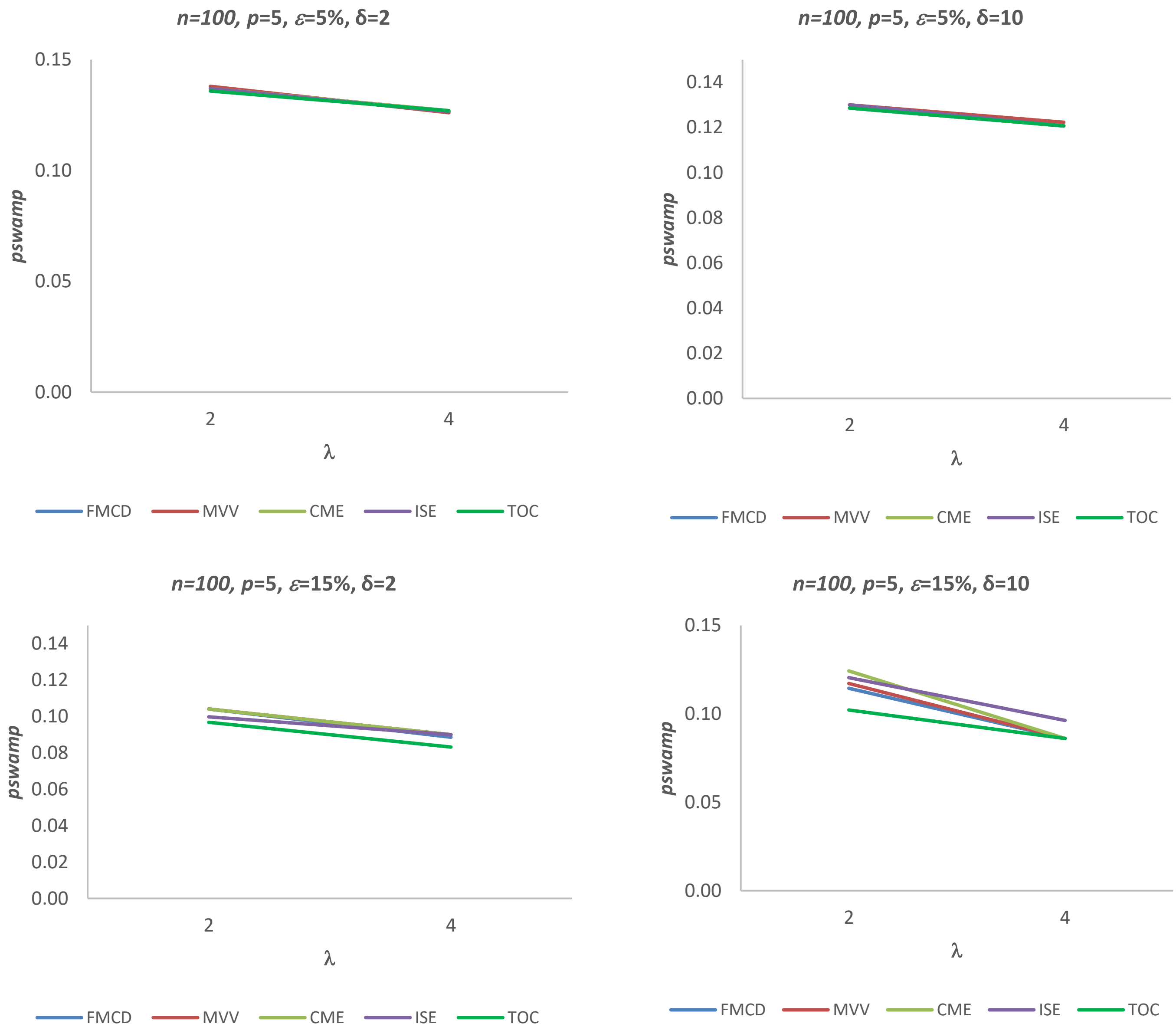

Figure 5. Plot of swamping error (*pswamp*) versus distance of outliers and inliers (λ)

Table 4 shows the *pout*, *pmask* and *pswamp* summary for the best robust estimator in each case. The bod entry is to show the best robust estimator. For *pout* values, it can be seen that TOC can be the best estimator or have the same performance as other robust estimators to detect outliers for most cases except when $n = 100, p = 2$. Out of 36 cases, TOC shows the best result or has a similar performance to other robust estimators in 8 cases when $\lambda = 4, \varepsilon = 5\%, 15\%$ and 4 cases when $\lambda = 2, \varepsilon = 5\%, 15\%$. This result indicates that TOC has a high probability of successfully detecting outliers when the distance between outliers and inliers by shifting the mean is large. TOC shows the best result or has similar performance to other estimators for *pmask* values in 8 cases when $\lambda = 4, \varepsilon = 5\%$ and 4 cases when $\lambda = 2, \varepsilon = 5\%$, while 9 cases when $\lambda = 4, \varepsilon = 15\%$ and 4 cases when $\lambda = 2, \varepsilon = 15\%$. This result also shows that TOC has a low probability of misclassifying outliers as inliers when the distance between outliers and inliers by shifting the mean is large. From Table 4 for *pswamp*, it can be seen that TOC is the best estimator to not misclassify inliers as outliers in most cases regardless of the values of λ and δ. TOC shows the lowest probability of swamping error compared to other robust estimators.

Table 4. Summary of the best robust estimators

n	p	δ	λ	*pout*		*pmask*		*pswamp*	
				$\varepsilon = 5\%$	$\varepsilon = 15\%$	$\varepsilon = 5\%$	$\varepsilon = 15\%$	$\varepsilon = 5\%$	$\varepsilon = 15\%$
30	2	2	2	FMCD, CME, **TOC**	MVV, CME, **TOC**	FMCD, CME, **TOC**	MVV, CME, **TOC**	MVV, ISE	FMCD, ISE
			4	**TOC**	MVV, CME, ISE	**TOC**	MVV, CME, ISE	MVV, CME, ISE	**TOC**
		10	2	FMCD	FMCD, MVV, CME, ISE	FMCD	FMCD, MVV, CME, ISE	**TOC**	**TOC**
			4	**ALL**	FMCD, MVV, ISE, **TOC**	**ALL**	FMCD, MVV, ISE, **TOC**	**ALL**	CME
	3	2	2	FMCD, MVV, CME, ISE	MVV	FMCD, MVV, CME, ISE	MVV	**TOC**	FMCD, CME, ISE, **TOC**
			4	FMCD, MVV, CME, ISE	**TOC**	FMCD, MVV, CME, ISE	FMCD, MVV, **TOC**	**TOC**	**TOC**
		10	2	FMCD	FMCD, MVV, ISE	FMCD	FMCD, MVV, ISE	**TOC**	**TOC**
			4	FMCD, MVV, CME	FMCD, MVV, CME	FMCD, MVV, CME	FMCD, MVV, CME	**TOC**	ISE, **TOC**
	5	2	2	CME	ISE	CME	ISE	ISE	CME
			4	**ALL**	FMCD, MVV, CME, **TOC**	**ALL**	**ALL**	ISE	FMCD
		10	2	ISE	ISE, **TOC**	ISE	ISE, **TOC**	**TOC**	CME
			4	ISE	FMCD, **TOC**	ISE	FMCD, **TOC**	**TOC**	MVV, CME, ISE
50	2	2	2	CME, ISE, **TOC**	FMCD, ISE	CME, ISE, **TOC**	FMCD, ISE	CME, ISE, **TOC**	FMCD, ISE
			4	FMCD, MVV, CME	ISE	FMCD, MVV, CME	ISE	**TOC**	FMCD, MVV, CME, **TOC**
		10	2	FMCD, CME, **TOC**	**ALL**	FMCD,CME, **TOC**	**ALL**	FMCD, CME, **TOC**	**ALL**
			4	FMCD, MVV	FMCD, MVV, CME, ISE	FMCD, MVV	FMCD, MVV, CME, ISE	FMCD, MVV	**TOC**
	3	2	2	FMCD, MVV, CME	MVV	FMCD, MVV, CME	MVV	**TOC**	MVV
			4	**TOC**	FMCD	**TOC**	**TOC**	**TOC**	**TOC**
		10	2	MVV, CME, **TOC**	CME	MVV, CME, **TOC**	CME	FMCD, ISE	**TOC**
			4	MVV, **TOC**	MVV, CME, ISE, **TOC**	MVV, **TOC**	MVV, CME, ISE, **TOC**	FMCD, CME, ISE	FMCD
	5	2	2	MVV, ISE	MVV	MVV, ISE	MVV	**TOC**	MVV
			4	**ALL**	**ALL**	**ALL**	**ALL**	**TOC**	FMCD, **TOC**
		10	2	MVV	MVV	MVV	MVV	FMCD	CME
			4	ISE	FMCD	ISE	FMCD	**TOC**	ISE
100	2	2	2	MVV	FMCD, MVV, CME, ISE	MVV	FMCD, MVV, CME ,ISE	**TOC**	**TOC**
			4	FMCD, MVV, CME, ISE	FMCD, MVV, CME, ISE	FMCD, MVV, CME, ISE	FMCD, MVV, CME ,ISE	**TOC**	**TOC**
		10	2	CME, ISE	MVV, ISE	CME, ISE	MVV, ISE	**TOC**	MVV, ISE
			4	FMCD, MVV, CME, ISE	FMCD	FMCD, MVV, CME, ISE	FMCD	**TOC**	FMCD
	3	2	2	FMCD, MVV, CME, ISE	FMCD, MVV, CME, ISE	FMCD, MVV, CME, ISE	FMCD, MVV, CME, ISE	**TOC**	FMCD, MVV, CME, ISE
			4	FMCD, MVV, CME, ISE	**ALL**	FMCD, MVV, CME, ISE	**ALL**	**TOC**	FMCD, MVV, CME, ISE
		10	2	FMCD, CME	**ALL**	ISE	**ALL**	**TOC**	**ALL**
			4	FMCD, CME	CME	FMCD, CME	CME	MVV	FMCD, MVV, ISE, **TOC**
	5	2	2	MVV	FMCD, MVV, CME	MVV	FMCD, MVV, CME	**TOC**	**TOC**
			4	**ALL**	**ALL**	**ALL**	**ALL**	MVV	**TOC**
		10	2	CME	CME	MVV, CME, ISE	CME, ISE	FMCD, CME, **TOC**	**TOC**
			4	FMCD, CME, ISE, **TOC**	ISE	FMCD, CME, ISE, **TOC**	ISE	FMCD,CME, ISE, **TOC**	FMCD, MVV, CME, **TOC**

Conclusions

This study investigates the performance of TOC to detect outliers for multivariate data. The performance of TOC is compared with other robust estimators, FMCD, MVV, CME and ISE, via a simulation study. This study uses one outlier scenario by simultaneously shifting the mean and covariance with various conditions, including the number of variables, sample size and percentage of outliers. From the simulation study, TOC shows good results and a promising approach as a robust estimator to detect outliers. It has a low probability of misclassifying outliers as inliers (masking error) in multivariate data ranges from 0.000 to 0.0689. TOC especially shows good results when the distance between outliers and inliers by shifting the mean is far. TOC is the best robust estimator for swamping error for most cases where the *pswamp* ranges from 0.0469 to 0.2363, whether the mean and covariance shift values are high or low. This range is the lowest *pswamp* values compared to other robust estimators. In conclusion, TOC is an applicable and promising approach for outlier detection in multivariate data and can be incorporated with other multivariate analyses [23]. TOC can also be applied to financial or health data as long as the data is continuous and low dimensions.

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

Acknowledgement

The authors thank Universiti Malaysia Terengganu for their financial support.

References

[1] Hadi, A. S., Rahmatullah Imon, A. H. M. M., & Werner, M. (2009). Detection of outliers. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1), 57–70.

[2] Su, X., & Tsai, C.-L. (2011). Outlier detection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3), 261–268.

[3] De Ketelaere, B., Hubert, M., Raymaekers, J., Rousseeuw, P. J., & Vranckx, I. (2020). Real-time outlier detection for large datasets by RT-DetMCD. *Chemometrics and Intelligent Laboratory Systems*, 199, Article 103957.

[4] Savić, M., Atanasijević, J., Jakovetić, D., & Krejić, N. (2022). Tax evasion risk management using a hybrid unsupervised outlier detection method. *Expert Systems with Applications*, 193, Article 116409.

[5] Cabana, E., Lillo, R. E., & Laniado, H. (2021). Multivariate outlier detection based on a robust Mahalanobis distance with shrinkage estimators. *Statistical Papers*, 62(4), 1583–1609.

[6] Herwindiati, D. E., Djauhari, M. A., & Mashuri, M. (2007). Robust multivariate outlier labeling. *Communications in Statistics—Simulation and Computation*, 36(6), 1287–1294.

[7] Rousseeuw, P. J. (1985). Multivariate estimation with high breakdown point. *Mathematical Statistics and Applications*, 8, 283–297.

[8] Salleh, R. M. (2013). *A robust estimation method of location and scale with application in monitoring process variability* (Doctoral dissertation). Universiti Teknologi Malaysia.

[9] Mashuri, M., Ahsan, M., Lee, M. H., & Dwi, D. P. (2021). PCA-based Hotelling's T² chart with fast minimum covariance determinant (FMCD) estimator and kernel density estimation (KDE) for network intrusion detection. *Computers & Industrial Engineering*, 158, Article 107447.

[10] Rousseeuw, P. J., & Van Driessen, K. (1999). A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3), 212–223.

[11] Lim, H. A., & Midi, H. (2016). Diagnostic robust generalized potential based on index set equality (DRGP(ISE)) for the identification of high leverage points in linear model. *Computational Statistics*, 31(3), 859–877.

[12] Salleh, R. M., & Djauhari, M. A. (2011). Robust Hotelling's T² control charting in spike production process. *International Seminar on the Application of Science & Mathematics 2011 (ISASM 2011)*, 1–8.

[13] Abd Mutalib, S. S. S., Satari, S. Z., & Wan Yusoff, W. N. S. (2019). A new robust estimator to detect outliers for multivariate data. *Journal of Physics: Conference Series*, 1366(1), Article 012104.

[14] Abd Mutalib, S. S. S., Satari, S. Z., & Wan Yusoff, W. N. S. (2021). Comparison of robust estimators' performance for detecting outliers in multivariate data. *Journal of Statistical Modeling and Analytics*, 3(2), 36–64.

[15] Abd Mutalib, S. S. S., Satari, S. Z., & Wan Yusoff, W. N. S. (2021). Comparison of robust estimators for detecting outliers in multivariate datasets. *Journal of Physics: Conference Series*, 1988(1).

[16] Sebert, D. M., Montgomery, D. C., & Rollier, D. A. (1998). A clustering algorithm for identifying multiple outliers in linear regression. *Computational Statistics & Data Analysis*, 27(4), 461–484.

[17] Rencher, A. C. (2002). *Methods of multivariate analysis* (2nd ed.). Wiley.

[18] Cerioli, A., Riani, M., & Torti, F. (2011). Accurate and powerful multivariate outlier detection. *Proceedings of*

the 58th World Statistical Congress of the International Statistical Institute, 5608–5613.

[19] Filzmoser, P. (2005). Identification of multivariate outliers: A performance study. *Austrian Journal of Statistics*, 34(2), 127–138.

[20] Filzmoser, P., Maronna, R., & Werner, M. (2008). Outlier identification in high dimensions. *Computational Statistics & Data Analysis*, 52(3), 1694–1711.

[21] Zulkipli, N. S., Satari, S. Z., & Wan Yusoff, W. S. (2022). The effect of different similarity distance measures in detecting outliers using single-linkage clustering algorithm for univariate circular biological data. *Pakistan Journal of Statistics and Operation Research*, 18(3), 561–573.

[22] Santos-Pereira, C. M., & Pires, A. M. (2002). Detection of outliers in multivariate data: A method based on clustering and robust estimators. In *Compstat* (pp. 291–296). Physica.

[23] Hubert, M. (2020). Robust multivariate statistical methods. In *Comprehensive Chemometrics* (2nd ed., pp. 107–122). Elsevier.