



# Mengenali Mahasiswa Kriteria Outlier

*by* Gunawan Ridowati

---

**Submission date:** 25-Jun-2020 11:56AM (UTC+0700)

**Submission ID:** 1349374290

**File name:** Mengenali\_Mahasiswa\_Kriteria\_Outlier.pdf (485.18K)

**Word count:** 3172

**Character count:** 18897

# MENGENALI MAHASISWA KRITERIA *OUTLIER* MENGGUNAKAN *LOCAL OUTLIER FACTOR* DAN *CONNECTIVITY-BASED OUTLIER FACTOR*

Ridowati Gunawan

Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma  
Universitas Sanata Dharma, Paingan Maguwoharjo Depok Sleman Yogyakarta  
rido@usd.ac.id

## Abstrak

Para dosen maupun pembimbing akademik akan sangat terbantu ketika sejak awal dapat mengenali karakteristik akademik dari setiap mahasiswanya. Fokus perhatian para dosen adalah terhadap mahasiswa yang memiliki karakteristik yang tidak biasa, atau dikatak<sup>2</sup> sebagai mahasiswa dengan kriteria *outlier*. Berbagai pendekatan dan algoritma telah banyak dikemukakan. Salah satunya adalah dengan menggunakan pendekatan *density based*. Algoritma *local outlier factor* digunakan untuk mendapatkan derajat perbedaan dari mahasiswa terhadap kumpulannya, sedangkan algoritma *connectivity-based outlier factor* digunakan untuk mendapatkan *outlier* seperti halnya *local outlier factor* terutama yang memiliki kepadatan yang rendah. Pada penelitian ini kedua algoritma ini digunakan untuk mengenali mahasiswa dengan Criteria *outlier* dengan data set mahasiswa angkatan 2010-2013. Atribut yang akan digunakan untuk mengenali mahasiswa Criteria *outlier* ini adalah data hasil psikotes yang dilaksanakan oleh seluruh mahasiswa beserta dengan indeks prestasi mahasiswa semester yang diperoleh oleh mahasiswa. Kedua algoritma ini dapat mengenali mahasiswa kriteria *outlier*. Akan tetapi keduanya menghasilkan beberapa data *outlier* yang berbeda. Algoritma *Connectivity-based Outlier Factor* menghasilkan jumlah data mahasiswa terindikasi *outlier* lebih banyak dibandingkan dengan algoritma *Loco/ Outlier Factor*.

**Kata kunci :** *Outlier, Local Outlier Factor, Connectivity-based Outlier Factor*.

## 1. Pendahuluan

### 1.1 Latar Belakang Masalah

Memprediksi mahasiswa baru yang akan mampu menyelesaikan studi sampai akhir merupakan kendala tersendiri. Proses seleksi mahasiswa baru yang bermacam-macam menyebabkan standar yang ditetapkan berbeda-beda. Untuk itulah maka diperlukan alat tes yang sama yang harus diberlakukan kepada seluruh mahasiswa baru sehingga memudahkan dalam melihat kualitas akademik mahasiswa. Setelah calon mahasiswa diterima, maka dilakukan proses tes lanjutan yang dikenal dengan psikotes. Walaupun tidak membawa dampak apapun (karena telah menjadi mahasiwa), akan tetapi proses ini penting untuk mendapatkan gambaran mahasiswa yang telah diterima. Dari hasil psikotes tidak semua mahasiswa memiliki hasil sesuai dengan yang diharapkan. Perlu segera diketahui mahasiswa mana saja yang memiliki karakteristik akademik yang berbeda dengan kumpulan. Mahasiswa dengan karakteristik yang berbeda inilah yang menjadi perhatian dari pembimbing akademik mapun dari pengurus program studi.

Teknik untuk mengenali karakteristik yang berbeda dikenal dengan nama Deteksi *Outlier*.

Dalam deteksi *outlier* terdapat 3 pendekatan yang secara umum sering dipakai yaitu: *Clustering Based*, *Distance Based* dan *Density Based*. Dari ketiga pendekatan tersebut muncul beragam algoritma yang digunakan untuk mencari *outlier* dari sekumpulan data. Akan tetapi tidak semua algoritma tersebut cocok digunakan untuk menganalisis sebuah *outlier* dari kelompok data, apalagi jika data tersebut memiliki dimensi yang tinggi atau memiliki atribut yang banyak. Salah satu algoritma yang ada adalah algoritma LOF (*Local Outlier Factor*) algoritma ini digunakan dalam pendekatan *Density Based*, algoritma ini tidak secara eksplisit menyatakan bahwa suatu obyek adalah sebuah *outlier*, namun algoritma memberikan bobot / derajat *outlier* terhadap suatu obyek nilai. Perbaikan dari LOF untuk menangani kepadatan data yang rendah dikembangkan algoritma COF (*Connectivity-based Outlier Factor*). COF termasuk ke dalam pendekatan *density based*, yang dalam menentukan sebuah data *outlier* berdasarkan derajat terhadap obyek yang lain juga.

Salah satu fungsi data mining dalam dunia pendidikan adalah untuk mengembangkan sebuah metode yang dapat menemukan keunikan dari sebuah data yang berasal dari sistem pendidikan tersebut, dan menggunakan metode tersebut untuk

lebih memahami siswa, sehingga dapat mengembangkan sebuah sistem yang sesuai [1].

Raharjo [4] telah berhasil menerapkan algoritma *LOF* dan Ruwidiati [6] telah berhasil menerapkan algoritma *COF*. Keduanya menggunakan data set yang sama yaitu data akademik mahasiswa untuk angkatan 2007-2008, dimana data akademik yang digunakan adalah data dari penerimaan mahasiswa baru baik dari jalur tes maupun dari jalur prestasi dan data indeks prestasi semester untuk setiap mahasiswa. Data mahasiswa *outlier* dari kedua algoritma tersebut telah dapat diperoleh, yang secara intuitif hal tersebut akan sulit diperoleh. Raharjo [4] menyimpulkan bahwa kondisi stabil hasil *outlier* terjadi pada saat minPts (jumlah ketetanggaan) lebih besar dari 7 pada saat mahasiswa dengan jalur tes dan jalur prestasi digabungkan. Sedangkan jika melihat dari jalur tes saja maka ketika minPts antara 5-10 menghasilkan jumlah mahasiswa *outlier* yang cenderung stabil. Nilai *LOF* sangat bergantung pada kepadatan jarak obyek data dalam suatu jangkauan minPts, ketika suatu Cluster memiliki kepadatan yang tinggi maka nilai *LOF* akan cenderung kecil, dan ketika suatu kluster memiliki kepadatan yang rendah maka nilai *LOF* akan cenderung lebih besar. Sedangkan Ruwidiati [6] menyimpulkan bahwa makin tinggi nilai minPts maka derajat *COF* nya akan semakin tinggi artinya kepadatannya akan semakin rendah.

Perbedaan alat tes menjadi masalah tersendiri dalam menghasilkan data *outlier*. Jalur tes yang berdasarkan hasil tes yang dilaksanakan Universitas, sedangkan jalur prestasi berdasarkan nilai raport. Untuk itu perlu dibuat alat ukur yang sama untuk menilai kemampuan akademik setiap mahasiswa.

Memperbaiki penelitian sebelumnya, penelitian yang dilakukan menggunakan data set yang berbeda dengan atribut yang berbeda. Nilai semester yang diperoleh mahasiswa akan dibandingkan dengan nilai psikotest. Terdapat 6 nilai dari nilai psikotest yang akan dijadikan sebagai atribut untuk mencari data *outlier*. Kedua algoritma tersebut akan coba untuk dibandingkan terutama dalam hal kecepatan dalam menghasilkan data *outlier* dan juga melihat data *outlier* yang dihasilkan. Apakah keduanya menghasilkan data *outlier* dengan mahasiswa yang sama atau berbeda dalam setiap iterasinya. Kemampuan dari *COF* yang dapat mendeteksi *outlier* yang memiliki kepadatan ketetanggaan yang rendah akan coba diamati.

Harapannya dengan menggunakan alat test yang sama maka evaluasi pembelajaran dapat lebih optimal. Hasil data *outlier* dapat dimanfaatkan program studi untuk memantau kemampuan akademik setiap mahasiswa. Pada saat proses evaluasi dilakukan oleh program studi setiap akhir semester IV dapat dengan mudah dilakukan apakah seorang mahasiswa akan dikeluarkan (DO) atau tetap dipertahankan. Pemantauan setiap semester terhadap mahasiswa yang terdeteksi sebagai *outlier*

dapat mencegah terjadinya DO sehingga mahasiswa dapat menyelesaikan studinya dengan tepat waktu, disamping itu juga jika *outlier* karena mahasiswa memiliki prestasi yang luar biasa dapat diarahkan untuk lebih positif seperti untuk mengikuti kompetisi atau didorong untuk menyelesaikan studinya lebih cepat.

## 1.1 Tujuan

Tujuan penelitian ini adalah menerapkan algoritma *Local Outlier Factor* dan *Connectivity-based Outlier Factor* ke dalam sebuah sistem untuk mendeteksi *outlier* dari data akademik mahasiswa yang berasal dari data hasil psikotes setelah mahasiswa diterima dan data prestasi akademik setiap semester. Tujuan yang kedua adalah untuk mencari perbedaan dari kedua algoritma tersebut.

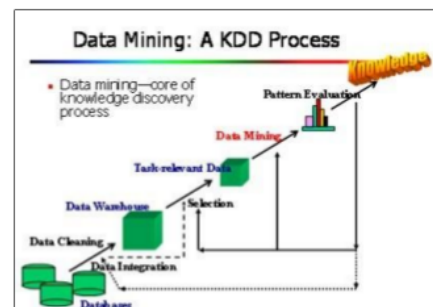
## 1.2 Rumusan Masalah

1. Bagaimana algoritma *Local Outlier Factor* dan *Connectivity-based Outlier Factor* dapat mendeteksi mahasiswa dengan kriteria *outlier* berdasarkan nilai test potensi akademik dan nilai akademik semester ?

2. Apakah perbedaan hasil antara algoritma *Local Outlier Factor* dan *Connectivity-based Outlier Factor* dalam mendeteksi mahasiswa dengan kriteria *outlier* ?

## 1.4 Metodologi Penelitian

Metodologi yang digunakan untuk menyelesaikan permasalahan tersebut mengikuti tahapan umum yang diilustrasikan oleh Jiawei Han dan Kamber [3] dalam proses *Knowledge Discovery in Databases (KDD)*. Proses KDD dapat dilihat pada Gambar 1.



Gambar 1. Proses Knowledge Discovery in Databases

Proses dimulai dari proses pembersihan data, proses integrasi dari data hasil psikotest dan data prestasi akademik, proses seleksi dan proses pencarian *outlier* yang akan diletakan pada proses

penambangan data. Hasil dari *outlier* akan dianalisis dikenali polanya sehingga dapat dimanfaatkan oleh pengambil keputusan.

## 2. Dasar Teori

### 2.1 Outlier

Secara lebih sederhana *outlier* adalah data yang terlalu berbeda atau tidak konsisten dengan satu set data. *Outlier* dapat disebabkan oleh kesalahan eksekusi atau juga oleh pengaturan program secara *default* [2].

### 2.2 Metode Density-Based

Metode *density-based* tidak secara eksplisit mengklasifikasikan sebuah obyek adalah *outlier* atau bukan, akan tetapi lebih kepada pemberian nilai kepada obyek sebagai derajat kekuatan obyek tersebut. Ukuran derajat kekuatan ini adalah *local outlier factor (LOF)*. Pendekatan untuk pencarian outlier ini hanya membutuhkan sebuah parameter yaitu *MinPts*. *Minpts* adalah jumlah tetangga terdekat yang digunakan untuk mendefinisikan *Local Neighborhood* suatu obyek.

#### 2.2.1 Algoritma Local Outlier Factor (LOF)

Secara sederhana algoritma *LOF* dideskripsikan sebagai berikut [2]:

1. Menghitung *k-distance* dari obyek *p*.  
Tujuan dari perhitungan *k-distance* ini adalah untuk menentukan tetangga dari *p*, secara sederhana *k-distance* dari sebuah obyek *p*, adalah jarak maksimal dari *P* terhadap tetangga terdekatnya
2. Menemukan *k-distance neighborhood* dari *p*.  
*k-distance neighborhood* suatu objek *p* dinotasikan *N<sub>k-distance</sub>(p)*, atau *N<sub>k</sub>(p)* dimana berisi setiap objek dengan jarak tidak lebih besar dari *kdistance (p)*.
3. Menghitung *reachability distance* dari obyek, *Reachability distance* dari suatu obyek *p* terhadap obyek *o* adalah *distance(p, o)* atau *kdistance(o)*, dengan membandingkan keduanya dicari nilai yang maksimum sehingga, *reach-dist<sub>i</sub>(p, o) = max{kdistance(o), distance(p, o)}*.
4. Menghitung kepadatan lokal dari obyek *p*.  
Kepadatan lokal / *local reachability density* dari *p* di definisikan seperti pada persamaan

$$lrd_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} reach-dist_{MinPts}(p, o)}{|N_{MinPts}(p)|} \quad \dots(1)$$

5. Menghitung dan membuat peringkat LOF dari setiap obyek.

*Local outlier factor* dari *P* didefinisikan seperti pada persamaan (2):

$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} \frac{lrd_{MinPts}(o)}{lrd_{MinPts}(p)}}{|N_{MinPts}(p)|} \quad \dots(2)$$

dalam algoritma LOF sebuah obyek dikatakan sebagai outlier apabila memiliki nilai LOF yang tinggi atau menjauhi 1, sedangkan obyek yang memiliki nilai LOF rendah atau mendekati 1 maka obyek tersebut tidak dapat dikategorikan sebagai outlier. Nilai LOF yang tinggi mengindikasikan bahwa obyek tersebut memiliki kepadatan yang rendah terhadap lingkungannya sehingga berpotensi menjadi sebuah outlier.

### 2.2.2 Algoritma Connectivity-based Outlier Factor (COF)

Algoritma untuk menghitung nilai COF sebagai berikut [7]:

1. Mencari *Nearest Neighbor (k-JVN)* untuk setiap record data *p* pada *k nearest neighbor (k-JVN)*;
2. Mencari *set based nearest path (SBN-path)* atau *s. SBN-path* dari record data *p* pada set *N<sub>k</sub>(p)* adalah urutan record dengan jarak terdekat masing-masing *p, s = {p1, p2, ..., pr}* sehingga untuk semua  $1 \leq r-1$ , *pt+1* adalah tetangga terdekat dari (*p1, ..., pr*) pada (*pt+1, ..., pr*).
3. Mencari *set based nearest trail (SBN-vail)* atau *tr. SBN-trail* adalah urutan *edge* terhadap *s* atau *SBN-path* di mana setiap tepi menghubungkan dua tetangga terdekat berturut-turut dari jalur *SBN-path*. Dapat dinotasikan *SBN-trail = (e(1), ..., e(r-1) j*.
4. Menghitung *Cost Description*. *Cost Description* adalah jarak dari masing-masing *edge* pada *SBN-trail*.
5. Menghitung *average chaining distance (ac-dist)* dari *pt* ke *N<sub>k</sub> - {p}*, dinotasikan dengan *ac-distN<sub>k</sub>(p)Up@1* dan didefinisikan seperti pada persamaan (3):

$$ac-dist_G(p_1) = \sum_{i=1}^{r-1} \frac{2(r-i)}{r(r-1)} \cdot dist(e_i) \quad \dots(3)$$

dimana *dist<sub>e</sub>* dinotasikan jarak antara node yang terdiri dari tepi. *Average chaining distance* dari *pt* ke *N<sub>k</sub> - {p}* adalah jumlah bobot dari *cost description sequence* dari *SBN trail* untuk beberapa *SBN path* dari *pt*, dan

juga dapat dilihat sebagai rata-rata jarak bobot dalam *cost description sequence* SBN-trail.

- Hitung *connectivity-based outlier factor* (COF) pada record data  $p$  sehubungan dengan  $k$ -neighborhood nya menggunakan rumus persamaan (4):

$$COF_k(p) = \frac{|N_k(p)| \cdot ac-dist_{N_k(p)}(p)}{\sum_{o \in N_k(p)} ac-dist_{N_k(o)}(o)} \quad \dots(4)$$

COF dihitung sebagai rasio *average chaining distance* dari data record  $p$  untuk  $N_k(p)$  dan *average chaining distance* pada  $k$ -distunre neighbor mereka sendiri.

### 3. Pengujian

#### 3.1 Data Pengujian

Data yang akan digunakan dalam pengujian adalah data hasil psikotes dan nilai indeks prestasi semester (IPS) semester 1-4. Karena untuk angkatan 2013 belum memiliki nilai IPS semester 4 maka akan diuji pada semester 1 saja. Proses pembaharuan dilakukan terhadap gudang data telah dilakukan oleh Rosa dkk [6] dengan menambahkan data Basil psikotes.

Psikotes yang dilakukan terdiri dari 6 mata tes yaitu (a). **Tanggapan Ruang (TR)** yang digunakan untuk mengukur kemampuan berpikir terhadap pemahaman terhadap susunan (struktur) benda-benda tiga dimensi beserta bagian-bagiannya.; (b). **Berpikir Matematis A (BMA)**, yang mengukur kemampuan berpikir terhadap pola-pola relasi dari sederetan bilangan.; (c). **Berpikir Matematis B (BMB)**, yang mengukur kemampuan berpikir induktif-deduktif khususnya dalam menerapkan prinsip-prinsip kuantitatif untuk menyelesaikan persoalan-persoalan yang membutuhkan perhitungan matematis (*mathematical reasoning*).; (d). **Berpikir Verbal Analogi (BVA)**, tes ini mengukur kemampuan berpikir verbal, khususnya dalam memahami relasi-relasi antara pengertian-pengertian yang dinyatakan secara verbal.; (e). **Berpikir Verbal Klasifikasi (BVB)** yang mengukur kemampuan berpikir logis (penalaran), khususnya dalam mengadakan klasifikasi obyek-obyek atau konsep-konsep. Kemampuan ini amat diperlukan dalam setiap bidang ilmu dan dalam komunikasi sehari-hari. Tes ini bersama tes berpikir Verbal Analogi mengukur kemampuan berpikir verbal atau mengukur kemampuan penalaran.; (f). **Vokabuler (VOK)** yang digunakan untuk mengukur luasnya pengetahuan bahasa, khususnya penguasaan arti kata-kata.

Psikotes juga mengukur tingkat kesesuaian minat mahasiswa dengan program studi. Untuk hasil tes ini akan digunakan sebagai bahan konfirmasi dari data hasil outlier yang dihasilkan.

Data lain yang digunakan adalah data dari sistem penerimaan mahasiswa baru yang berupa

nilai tes dari jalur tes dan nilai raport dari jalur prestasi. Data ini juga akan digunakan untuk melakukan proses konfirmasi terhadap data outlier yang dihasilkan.

Jumlah data dari setiap angkatan adalah dapat dilihat pada tabel 1:

**Tabel 1.** Tabel jumlah data untuk setiap angkatan

| Angkatan | Jumlah |
|----------|--------|
| 2010     | 83     |
| 2011     | 74     |
| 2012     | 104    |
| 2013     | 116    |

#### 3.2 Hasil Pengujian Deteksi Outlier

Pengujian terhadap data dilakukan untuk menentukan apakah muncul data outlier. Beberapa pengujian yang dilakukan adalah :

- Pengujian terhadap data mahasiswa angkatan 2010 menggunakan algoritma *LOF* dan *COF* dengan minPts=5,10,20 dan batas outlier =1.4. Data yang digunakan adalah data nilai 6 mata tes psikotes dan ips1. Hasil untuk *LOF* dapat dilihat pada tabel 2. Sedangkan untuk hasil dari algoritma *COF* dapat dilihat pada tabel 3.

**Tabel 2.** Hasil Pengujian Deteksi Outlier Menggunakan Al oritma *LOF*, minPts=5,10,20.

| MinPts=5 |          | MinPts=10 |          | MinPts=20 |           |
|----------|----------|-----------|----------|-----------|-----------|
| MHS      | Lof      | MHS       | Lof      | MHS       | Lof       |
|          | 1,594291 |           | 1,532778 |           | 1,6799761 |
| 10076    | 1,594238 | 10002     | 1,530035 | 10002     | 1,4388558 |
| 10085    | 1,571111 | 10076     | 1,526075 | 10069     | 1,4106650 |
| 10002    | 1,514930 |           | 1,517622 |           |           |
| 10044    | 1,424367 | 10085     | 1,425006 |           |           |
| 10065    | 1,420746 |           |          |           |           |

**Tabel 3.** Hasil Pengujian Deteksi Outlier Menggunakan Algoritma *COF*, minPts=5,10,20.

| MinPts=5 |          | MinPts=10 |          | MinPts=20 |          |
|----------|----------|-----------|----------|-----------|----------|
| MHS      | COF      | MHS       | COF      | MHS       | COF      |
|          | 1,734402 |           |          |           | 1,649307 |
| 10002    | 1,572352 | 10076     | 1,657441 | 10113     | 1,634255 |
| 10065    | 1,535325 |           |          | 10065     | 1,623459 |
| 10085    | 1,480782 | 10074     | 1,561627 | 10085     | 1,583859 |
| 10031    | 1,465295 | 10065     | 1,537553 | 10069     | 1,535261 |
| 10036    | 1,453033 | 10085     | 1,502336 | 10002     |          |
| 10078    | 1,438422 | 10069     | 1,490034 | 10044     | 1,464662 |

|       |            |       |          |    |          |
|-------|------------|-------|----------|----|----------|
| 10074 | 1,432376   | 10036 | 1,449191 | 00 | 1,406745 |
| 10069 | 1,417398   |       |          |    |          |
| 10003 | 1,409372   |       |          |    |          |
| 10089 | 1,40876714 |       |          |    |          |

2. Pengujian terhadap data mahasiswa angkatan 2010 menggunakan algoritma *LOF* dan *COF* dengan minPts=20 dan batas outlier =1.4 untuk data nilai tes psikotes dan ips1 ; nilai psikotes dengan ips2 dan nilai psikotes dengan ips3. Hasil untuk *LOF* dapat dilihat pada tabel 4. 8 dangan untuk hasil dari algoritma *COF* dapat dilihat pada tabel 5.

4. L  
'abpts Han''batan'utuanrDI 4 dari 'pu'k tes daF IPS1 s.d. IPS4.

| MinPts=20 |            |       |         |       |         |
|-----------|------------|-------|---------|-------|---------|
| IPS 1     |            | IPS 2 |         | IPS 3 |         |
| MHS       | Lof        | MHS   | Lot     | MHS   | Lot     |
|           | 1,67997    | 10060 | 2,11532 |       | 1,68861 |
| 10003     | 1,43585    |       | 1,78404 | 10003 | 1,45748 |
| 10002     | 1,43885    | 10003 | 1,78404 | 10085 | 1,48748 |
| 10069     | 1,40876714 | 10002 | 1,53063 | 10002 | 1,47039 |
|           |            | 10009 | 1,45139 | 10076 | 1,42976 |
|           |            | 10113 | 1,40081 |       |         |

**Tabel S.** Hasil Pengujian Deteksi Outlier COF minPts=20 dan batas outlier=1.4 dari psikotes dan IPS1 s.d. IPS4.

| MinPts=20 |         |       |         |       |         |
|-----------|---------|-------|---------|-------|---------|
| IPS1      |         | IPS 2 |         | IPS 3 |         |
| MHS       | COF     | MHS   | COF     | MHS   | COF     |
|           | 1,54833 | 10060 | 2,32819 |       | 1,79337 |
|           | 1,64930 |       | 1,9037b |       | 1,70425 |
|           | 1,63425 |       | 1,68195 |       | 1,68653 |
| 10065     | 1,62345 |       | 1,61768 | 10074 | 1,62331 |
|           | 1,58385 |       | 1,60600 | 10065 | 1,55821 |
|           | 1,53526 | 10065 | 1,59745 |       | 1,54159 |
|           | 1,53270 |       | 1,57599 |       | 1,48663 |
|           | 1,46466 |       | 1,51558 |       | 1,47951 |
| 10074     | 1,40674 |       | 1,51055 | 10031 | 1,46692 |
|           |         | 10001 | 1,45613 |       |         |
|           |         | 10031 | 1,42472 |       |         |
|           |         | 10036 | 1,40502 |       |         |
|           |         | 10070 | 1,40004 |       |         |

3. Pengujian terhadap waktu eksekusi program untuk algoritma *LOF* dan *COF*. Data yang digunakan adalah data seluruh angkatan dengan menggunakan 6 nilai mata tes dan ips1

9  
dengan minpts=20. Hasil dapat dilihat pada tabel 6 untuk algoritma *LOF* dan tabel 7 untuk algoritma *COF*.

**Tabel 6.** Waktu Proses *LOF* dengan berbagai ukuran data dan MinPts

| Angkatan<br>(jumlah) | Minpts (detik) |       |       | Rata •Rata<br>(detik) |
|----------------------|----------------|-------|-------|-----------------------|
|                      | 5              | 10    | 30    |                       |
| 2010 (53)            | 0,26           | 0,38  | 0,578 | 0,406                 |
| 2011 (74)            | 0,2            | 0,24  | 0,378 | 0,272                 |
| 2012 (104)           | 0,35           | 0,434 | 0,6   | 0,4613                |
| 2013 (116)           | 0,61           | 0,54  | 0,69  | 0,6133                |

**Tabel 7.** Waktu Proses *COF* dengan berbagai ukuran data dan MinPts

| Angkatan<br>(jumlah) | MinPts (detik) |       |      | Rata•rata<br>(detik) |
|----------------------|----------------|-------|------|----------------------|
|                      | 5              | 10    | 20   |                      |
| 2010(83)             | 0,46           | 0,48  | 0,66 | 0,533                |
| 2011 (74)            | 0,2            | 0,48  | 0,59 | 0,423                |
| 2012(104)            | 0,44           | 0,616 | 0,54 | 0,532                |
| 2013(116)            | 0,626          | 0,651 | 1,38 | 0,862                |

### 3.3 Analisis Hasil Pengujian

Analisis dilakukan terhadap setiap hasil pengujian yang telah dijelaskan pada 3.2

Dari pengujian pertama, dengan menggunakan data angkatan 2010 untuk nilai minPts yang berbeda diperoleh hasil :

- Dari algoritma *LOF*, terdapat 2 orang mahasiswa yaitu 10002 dan 10003 yang selalu menjadi mahasiswa yang terindikasi *outlier*.
- Dari algoritma *COF*, terdapat 4 orang mahasiswa yang selalu menjadi outlier yaitu mahasiswa 10002, 10003, 10113 dan 10076.
- Berdasarkan tabel 2, dengan menggunakan algoritma *LOF* maka makin tinggi nilai *LOF* maka jumlah mahasiswa yang terindikasi *outlier* makin sedikit, derajat *LOF* untuk mahasiswa yang paling *outlier* makin tinggi. Yang berarti bahwa kepadatan datanya semakin rendah.
- Berdasarkan tabel 2, dengan menggunakan *COF* makin tinggi minPts maka semakin sedikit mahasiswa yang terindikasi *outlier*. Derajat *COF* semakin tinggi pula. Sehingga dapat disimpulkan kepadatannya makin berkurang.
- Berdasarkan tabel 2 dan 3, algoritma *COF* lebih banyak mendapatkan data mahasiwa yang terindikasi *outlier*.

Walapun yang ditampilkan hanyalah data untuk angkatan 2010, akan tetapi data untuk angkatan 2011 sd 2013 memiliki karakteristik yang sama. Yaitu makin tinggi nilai minPts

untuk LOF dan COF maka jumlah data yang terdeteksi sebagai *outlier* semakin sedikit. Jumlah data yang terdeteksi *outlier* lebih banyak untuk yang menggunakan COF.

2. Dari pengujian dengan memperhatikan nilai mata uji psikotes dengan ips1, ips2, ips3 untuk masing-masing algoritma diperoleh hasil sebagai berikut :
  - a. Untuk algoritma LOF, karena yang dibandingkan adalah dengan data akademik setiap semester, maka fokus adalah pada mahasiswa yang selalu menjadi *outlier*, yaitu mahasiswa 10002 dan 10003.
  - b. Untuk algoritma COF, 12 mahasiswa yang selalu menjadi *outlier* pada semester 1 sampai dengan semester 3 adalah mahasiswa 10003, 10076, 10065, 10085, 10069, 10002 dan 10044.
  - c. Jumlah mahasiswa yang terdeteksi *outlier* dengan menggunakan COF jumlahnya lebih banyak dari yang menggunakan LOF. Dua orang mahasiswa selalu ada di algoritma COF dan LOF adalah mahasiswa 10002 dan 10003.
3. Pengujian dilihat dari waktu eksekusi perangkat lunak. Hasil yang diperoleh adalah :
  - a. Untuk algoritma LOF, makin tinggi nilai minPts maka waktu eksekusi program makin lama, sedangkan makin banyak jumlah datanya juga akan semakin lama waktu eksekusinya.
  - b. Untuk algoritma COF, maka makin tinggi nilai minPts maka akan semakin lama waktu eksekusinya, demikian juga dengan jumlah data yang semakin banyak maka akan semakin lama waktu eksekusinya.
  - c. Rata-rata waktu eksekusi COF lebih lambat dibandingkan dengan algoritma LOF.

#### 4. Kesimpulan

Dari hasil peneliti<sup>1</sup> ini, mengenali mahasiswa yang terdeteksi *outlier* menggunakan algoritma *Local Outlier Factor* (LOF) dan *Connectivity-based Outlier Factor* (COF) dapat diperoleh beberapa kesimpulan sebagai berikut :

1. Kedua algoritma dapat digunakan untuk mendeteksi mahasiswa *outlier*.
2. Nilai atribut minPts sangat mempengaruhi derajat LOF maupun COF. Makin tinggi nilai minPts maka derajat *outlier* nya semakin tinggi yang berarti bahwa semakin rendah kepadatannya.
3. Dengan algoritma COF data dengan tingkat kepadatan yang rendah dapat terdeteksi yang sebelumnya tidak terdeteksi dengan menggunakan algoritma LOF.
4. Waktu eksekusi LOF lebih cepat dibandingkan dengan waktu eksekusi menggunakan COF.

5. Kedua algoritma tidak secara eksplisit menyatakan sebuah obyek adalah *outlier*, tetapi berdasarkan derajat ooi/iernya.

#### 7. Daftar Pustaka

- [1] Baker, R. S., & Yacef, K., 2009, *The State of Educational Data Mining in 2009: A Review and Future Visions*
- [2] Breunig, M. M., Kriegek, H.-P., Ng, R. T., & Sander, J., 2000, *LOF: Identifying Density-Based Local Outliers*. ACM 2000.
- [3] Han, J., & Kamber, M., 2006, *Data Mining Concept and Techniques*, San Fransisco: Elsevier.
- [4] Harjo, Daniel. T., Gunawan, R., 2014, *Deteksi Outlier Menggunakan Algoritma Local Outlier Factor (Studi Kasus Data Akademik Universitas ABC)*, Prosiding KNSI 2014, Makasar.
- [5] Rosa, P. H., Gunawan, R., & Wijono, S. H. , 2012, *The Development of Academic Data Warehouse as a Basic for Decision Making A Case Study at XYZ University*, International Conference an Enterprise Information Systems and Application.
- [6] Ruwidati, Y.A., 2014, *Deteksi Outlier Menggunakan Algoritma Connectivity-based Outlier Factor (Studi Kasus Data Akademik Mahasiswa TI Universitas Sanata Dharma)*, Yogyakarta.
- [7] Tang, J., Chen, Z., Fu, A., & Cheung, D., 2002, *Enhancing Effectiveness of Outlier Detections for Low Density Pattern*, PAKDD.

# Mengenali Mahasiswa Kriteria Outlier

## ORIGINALITY REPORT

11%

SIMILARITY INDEX

9%

INTERNET SOURCES

3%

PUBLICATIONS

5%

STUDENT PAPERS

## PRIMARY SOURCES

1

[studylib.net](http://studylib.net)

Internet Source

2%

2

[mafiadoc.com](http://mafiadoc.com)

Internet Source

1%

3

[core.ac.uk](http://core.ac.uk)

Internet Source

1%

4

[www.bcs.org](http://www.bcs.org)

Internet Source

1%

5

Submitted to University of Technology, Sydney

Student Paper

1%

6

Submitted to Carnegie Mellon University

Student Paper

1%

7

Submitted to Kaplan International Colleges

Student Paper

1%

8

[fr.scribd.com](http://fr.scribd.com)

Internet Source

1%

9

[doku.pub](http://doku.pub)

Internet Source

1%

|    |                                                                                               |      |
|----|-----------------------------------------------------------------------------------------------|------|
| 10 | <a href="http://www.mitrariset.com">www.mitrariset.com</a><br>Internet Source                 | <1 % |
| 11 | <a href="http://www.computer.org">www.computer.org</a><br>Internet Source                     | <1 % |
| 12 | Submitted to Universitas Airlangga<br>Student Paper                                           | <1 % |
| 13 | <a href="http://www.encyclopedias.biz">www.encyclopedias.biz</a><br>Internet Source           | <1 % |
| 14 | Submitted to UIN Sultan Syarif Kasim Riau<br>Student Paper                                    | <1 % |
| 15 | <a href="http://library.binus.ac.id">library.binus.ac.id</a><br>Internet Source               | <1 % |
| 16 | <a href="http://fitrimustikaa.blogspot.com">fitrimustikaa.blogspot.com</a><br>Internet Source | <1 % |

Exclude quotes      On  
Exclude bibliography      On

Exclude matches      < 5 words