DAFTAR PUSTAKA

Bobak. 1999. *Maternal Nursing Care Plans*. Mosby Company.

Croux, C., Filzmoser, P., dan Joossens, K. 2008. Classification Efficiencies for Robust Linier Discriminant Analysis. *Statistica Sinicia*, hal. 581– 599.

Depkes RI. 1998. *Profil Kesehatan Indonesia 1999*. Jakarta.

Hair, J.F., Anderson, R.E., Thamtham, R.L., dan Black, W.C. 1998. *Multivariate Data Analysis. Fifth Edition*. Prentice Hall International, Inc., New Jersey.

Hardle, W dan Simar, L. 2003. *Applied Multivariate Statistical Analysis*. Method & Data Technologies.

Hayati, M. 2009. *Analisis Diskriminan Pada Faktor-Faktor Yang Mempengaruhi Gizi Buruk Balita di Jawa Timur*. Jurusan Statistika. Skripsi tidak diterbitkan. Surabaya. FMIPA ITS.

Haykin, S. 1999. *Neural A Chomprehensive Foundation*. Prentice Hall, Inc., New Jersey.

He, H. dan Fung, W.F. 2000. High Breakdown Estimation for Multiple Populations with Applications to Discriminant Analysis. *Jurnal of Multivariate Statistics*, hal. 151 – 162.

Johnson, R.A. dan Wichern, D.W. 2002. *Applied Multivariate Statistical Analysis. Fifth Edition*. Prentice Hall, Inc., United States of America.

Kardjiati, S.M. 1987. *Determinan Bayi Berat Lahir Rendah dalam Kaitannya dengan Kesehatan Bayi Baru Lahir dan Keamanan Persalinan: laporan Lokakarya Peningkatan Keamanan Persalinan dan Kesehatan Bayi Baru Lahir*. Perkumpulan Perinatologi Indonesia. Depkes RI.

Manuaba, I.B. 1998. *Ilmu Kebidanan, Penyakit Kandungan, dan Keluarga Berencana*. Penerbit buku kedokteran EGD, Jakarta.

Matjik, A.A. dan Sumertajaya, I.M. 2011. *Sidik Peubah Ganda Dengan Menggunakan SAS.* Edisi Pertama. IPB PRESS, Bogor.

Noeryanti, 2001. Detecting Multiple Outliers in Multivariate Samples with S-estimation Method. *Jurnal Teknologi Industri*, Vol. V, hal. 217 – 226.

Rencher, A.C. 2002. *Methods of Multivariate Analysis*. Second Edition. John Wiley & Sons, Inc., Canada.

Rousseuw, P.J., Lopuhaa, H.P., dan Campbell, N.A. 1998. On The Calculation of A Robust S-Estimator of A Covariance Matrix. *Journal Statistics in Medicine*, 17, hal. 2685 – 2695.

Rousseuw, P.J. dan Zomeren, V.B.C. 1990. Unmasking Multivariate Outlier and Leverage Points. *Journal of American Statistical Association*, hal. 633 – 661.

Saraswati, E. 1998. *Resiko Ibu Hamil Kurang Energi Kronis (KEK) dan Anemia untuk melahirkan Bayi dengan Berat Badan Lahir Rendah (BBLR)*. Penelitian Gizi dan Makanan jilid 21.

Sharma, S. 1996. *Applied Multivariate Technique*. John Wiley & Sons, Inc., New York.

Timm, N.H. 1975. *Multivariate Analysis with Application in Education and Psychology*. Wadsworth Publishing Co, Inc., Belmont, California.

Lampiran 1. Berat Bayi Baru Lahir Bulan Januari 2012 sampai November 2012 di Puskesmas Manyar

No.	Berat Bayi Baru Lahir (gram)	Kelompok (Y)
1	3400	1
2	3000	1
3	3300	1
4	3000	1
5	3100	1
6	3800	1
7	2900	1
8	2900	1
9	3000	1
10	3600	1
11	2900	1
12	2900	1
13	3000	1
14	2900	1
15	3000	1
16	3100	1
17	3100	1
⋮	⋮	⋮
298	1700	2
299	2200	2
300	2100	2
301	2500	2
302	1700	2
303	2400	2
304	2300	2
305	1500	2
306	1500	2
307	1600	2
308	2300	2

Keterangan:
1 = Kelompok Berat Bayi Lahir Normal (berat bayi > 2500 gram)
2 = Kelompok Berat Bayi Lahir Rendah (berat bayi ≤ 2500 gram)

Lampiran 2. Peubah-Peubah yang Mewakili Keadaan Ibu Saat Hamil

No.	X_1	X_2	X_3	X_4	X_5	X_6
1	28	0	40	23	10.8	0
2	20	0	40	24	11	0
3	20	0	38	23.5	11	0
4	21	0	69	26	11.1	0
5	24	0	48	24	11	0
6	26	0	55	25	11	0
7	24	0	50	24	11	0
8	24	0	51	24.5	11	0
9	31	1	54	26	11	8
10	27	2	60	28	12	2
11	22	0	45	23.5	11.5	0
12	30	1	50	25	11	5
13	29	0	38.5	23	10.5	0
14	22	1	58	28	11	18
15	25	0	50	25	11.8	0
16	33	1	48	25	12	7
17	31	2	55	25	12	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮
298	37	3	51	28	11	8
299	37	5	61	27	10.8	4
300	28	2	50	22.5	10.8	4
301	19	0	52	27	11.2	0
302	38	2	46	23	11	7
303	30	0	49.5	27	11	0
304	24	0	45.5	21	11.8	0
305	35	0	59	27	11	0
306	37	5	61	27	10.8	4
307	19	0	50	26	11	0
308	27	1	47	24.7	11.1	8

Lampiran 2. (Lanjutan)

Keterangan:

X_1 : Usia ibu (tahun)

X_2 : Jumlah anak (anak)

X_3 : Berat badan ibu prahamil (kg)

X_4 : Ukuran LILA ibu hamil (cm)

X_5 : Kadar Hb ibu hamil (gr/dl)

X_6 : Jarak kelahiran (tahun)

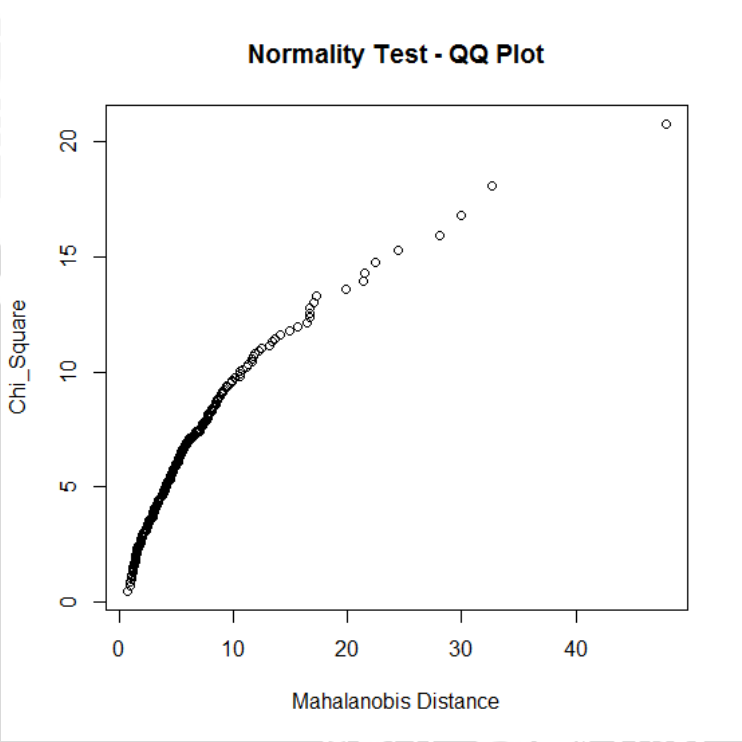
Lampiran 3. *Training set* (TS)

No.	X_1	X_2	X_3	X_4	X_5	X_6	Y
1	24	0	48	24	11.6	0	1
2	25	1	37	23.2	11.7	4	1
3	26	1	54	28	11	5	1
4	32	0	43	24	11	0	1
5	34	3	56	25	11.7	4	1
6	31	2	81	30	11	5	1
7	28	1	67.5	31	11	3	1
8	18	0	45	23	11	0	1
9	24	1	50	25	11	4.5	1
10	28	1	34	23	11	4	1
11	24	0	47	21.5	11.8	0	1
12	37	2	42	22.5	11.8	11	1
13	31	2	52	27	11.4	5	1
14	31	1	54	26	11	8	1
15	24	0	50	24	11	0	1
16	34	2	71	33	11	2.5	1
17	27	0	54	26	9.2	0	1
18	20	1	59	29	11	3	1
19	23	1	69	27	11.8	5	1
20	40	6	58	29	11.7	5	1
21	21	0	46	26	11	0	1
22	28	1	47	24.5	11.4	7	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
236	37	5	61	27	10.8	4	2
237	34	1	37	22.5	11	7	2
238	24	0	40	24	10.9	0	2
239	37	5	61	27	10.8	4	2
240	19	0	52	27	11.2	0	2
241	37	3	51	28	11	8	2
242	33	1	48	24	11	1	2
243	35	0	59	27	11	0	2
244	33	3	45	23	10.5	5	2
245	22	0	55	24	11.5	0	2
246	20	3	40	24	12	2	2

Lampiran 4. ***Testing set*** **(VS)**

No.	X_1	X_2	X_3	X_4	X_5	X_6	Y
1	22	0	46	24	10.8	0	1
2	24	0	45	24	11.4	0	1
3	23	0	79	24	12.3	0	1
4	18	0	80	30	11.4	0	1
5	31	1	55	26	12.5	5	1
6	22	0	45	24	22.6	0	1
7	26	2	49	25	10.8	1	1
8	24	1	56	24	11.2	4	1
9	38	3	52	29	10.8	8	1
10	21	0	68	24	11.7	0	1
11	25	1	50	23.5	11	2	1
12	31	1	48	25	11	5	1
13	20	0	51	24	11.8	0	1
14	36	1	56	25.2	11.3	4	1
15	26	2	45	23.5	11.3	1.5	1
16	21	1	47	24	11	2.5	1
17	24	0	50	24.5	11	0	1
18	37	1	56	29	11	5	1
19	35	1	47	24	11	8	1
20	29	1	61	32	11.4	5	1
21	24	0	47	23.5	10.8	0	1
22	21	0	46	22.5	11.2	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
52	32	2	54	25	11	7	1
53	22	1	75	28	11.4	1.2	1
54	24	1	70	27	11.4	4	1
55	26	0	53	25	12	0	1
56	30	2	40	24	11.2	10	1
57	26	0	38	22	11	0	2
58	24	0	31	18	11	0	2
59	24	0	47	24	11	0	2
60	24	0	45.5	21	11.8	0	2
61	25	0	50	25	12	0	2
62	20	0	45	24	11	0	2

Lampiran 5. Hasil Pengujian Asumsi Sebaran Normal Multivariat



Lampiran 6. Hasil Pengujian Asumsi Kehomogenan Matriks Ragam Peragam

Test Results

Box's M		50.302
F	Approx.	2.183
	df1	21
	df2	6.886E3
	Sig.	.001

Tests null hypothesis of equal population covariance matrices.

Lampiran 7. Hasil Pengujian Perbedaan Vektor Rata-Rata

Wilks' Lambda

Test of Function(s)	Wilks' Lambda	Chi-square	df	Sig.
1	.944	13.902	6	.031

Lampiran 8. Hasil Identifikasi Pencilan Pada *Training set*

```
Chi.sq = 14.44938

x- 1 = 1.304054 --> bukan pencilan
x- 2 = 4.875746 --> bukan pencilan
x- 3 = 3.372323 --> bukan pencilan
x- 4 = 8.430354 --> bukan pencilan
x- 5 = 5.097002 --> bukan pencilan
x- 6 = 11.67061 --> bukan pencilan
x- 7 = 5.539238 --> bukan pencilan
x- 8 = 4.024695 --> bukan pencilan
x- 9 = 2.106161 --> bukan pencilan
x- 10 = 5.476774 --> bukan pencilan
x- 11 = 3.964802 --> bukan pencilan
x- 12 = 11.27582 --> bukan pencilan
x- 13 = 2.261156 --> bukan pencilan
x- 14 = 4.760575 --> bukan pencilan
x- 15 = 1.547294 --> bukan pencilan
x- 16 = 12.22877 --> bukan pencilan
x- 17 = 21.46305 --> pencilan
x- 18 = 7.063406 --> bukan pencilan
x- 19 = 8.611391 --> bukan pencilan
x- 20 = 28.03863 --> pencilan
x- 21 = 3.659328 --> bukan pencilan
x- 22 = 3.020929 --> bukan pencilan
x- 23 = 3.507245 --> bukan pencilan
x- 24 = 2.506022 --> bukan pencilan
x- 25 = 5.865617 --> bukan pencilan
x- 26 = 3.313305 --> bukan pencilan
x- 27 = 5.875084 --> bukan pencilan
x- 28 = 1.625246 --> bukan pencilan
x- 29 = 29.91721 --> pencilan
x- 30 = 0.9044848 --> bukan pencilan
x- 31 = 13.40678 --> bukan pencilan
x- 32 = 10.48358 --> bukan pencilan
x- 33 = 13.6121 --> bukan pencilan
x- 34 = 3.065181 --> bukan pencilan
x- 35 = 9.796845 --> bukan pencilan
x- 36 = 5.762363 --> bukan pencilan
x- 37 = 1.320286 --> bukan pencilan
x- 38 = 47.95035 --> pencilan
x- 39 = 11.64082 --> bukan pencilan
x- 40 = 1.970648 --> bukan pencilan
x- 41 = 1.541098 --> bukan pencilan
x- 42 = 4.482041 --> bukan pencilan
```

Lampiran 8. (Lanjutan)

```
x- 43 = 6.562329 --> bukan pencilan
x- 44 = 4.400765 --> bukan pencilan
x- 45 = 1.10644 --> bukan pencilan
x- 46 = 4.446428 --> bukan pencilan
x- 47 = 1.866347 --> bukan pencilan
x- 48 = 11.16618 --> bukan pencilan
x- 49 = 1.863923 --> bukan pencilan
x- 50 = 8.403219 --> bukan pencilan
x- 51 = 1.898697 --> bukan pencilan
x- 52 = 1.523126 --> bukan pencilan
x- 53 = 2.416647 --> bukan pencilan
x- 54 = 8.91107 --> bukan pencilan
x- 55 = 3.565912 --> bukan pencilan
x- 56 = 3.321772 --> bukan pencilan
x- 57 = 3.88258 --> bukan pencilan
x- 58 = 4.678219 --> bukan pencilan
x- 59 = 7.129051 --> bukan pencilan
x- 60 = 6.643121 --> bukan pencilan
x- 61 = 2.900645 --> bukan pencilan
x- 62 = 2.069356 --> bukan pencilan
x- 63 = 3.103019 --> bukan pencilan
x- 64 = 4.808828 --> bukan pencilan
x- 65 = 14.88324 --> pencilan
x- 66 = 2.938497 --> bukan pencilan
x- 67 = 1.815123 --> bukan pencilan
x- 68 = 6.440606 --> bukan pencilan
x- 69 = 2.86089 --> bukan pencilan
x- 70 = 3.889414 --> bukan pencilan
x- 71 = 3.432159 --> bukan pencilan
x- 72 = 1.863377 --> bukan pencilan
x- 73 = 4.629014 --> bukan pencilan
x- 74 = 2.935685 --> bukan pencilan
x- 75 = 4.09309 --> bukan pencilan
x- 76 = 32.62514 --> pencilan
x- 77 = 0.6941274 --> bukan pencilan
x- 78 = 5.178286 --> bukan pencilan
x- 79 = 7.467955 --> bukan pencilan
x- 80 = 1.09946 --> bukan pencilan
x- 81 = 8.118964 --> bukan pencilan
x- 82 = 1.226389 --> bukan pencilan
x- 83 = 2.410656 --> bukan pencilan
x- 84 = 12.45851 --> bukan pencilan
x- 85 = 8.061432 --> bukan pencilan
x- 86 = 1.918053 --> bukan pencilan
x- 87 = 4.337638 --> bukan pencilan
```

Lampiran 8. (Lanjutan)

```
x- 88 = 4.258177 --> bukan pencilan
x- 89 = 5.826647 --> bukan pencilan
x- 90 = 5.282423 --> bukan pencilan
x- 91 = 4.215762 --> bukan pencilan
x- 92 = 5.331427 --> bukan pencilan
x- 93 = 4.484621 --> bukan pencilan
x- 94 = 1.129121 --> bukan pencilan
x- 95 = 2.306537 --> bukan pencilan
x- 96 = 6.341259 --> bukan pencilan
x- 97 = 1.427362 --> bukan pencilan
x- 98 = 4.708533 --> bukan pencilan
x- 99 = 2.128387 --> bukan pencilan
x- 100 = 17.04258 --> pencilan
x- 101 = 7.283198 --> bukan pencilan
x- 102 = 2.933368 --> bukan pencilan
x- 103 = 1.082202 --> bukan pencilan
x- 104 = 3.298622 --> bukan pencilan
x- 105 = 2.507947 --> bukan pencilan
x- 106 = 1.04154 --> bukan pencilan
x- 107 = 4.180094 --> bukan pencilan
x- 108 = 3.421592 --> bukan pencilan
x- 109 = 3.026945 --> bukan pencilan
x- 110 = 3.167152 --> bukan pencilan
x- 111 = 2.567239 --> bukan pencilan
x- 112 = 3.25038 --> bukan pencilan
x- 113 = 8.555438 --> bukan pencilan
x- 114 = 2.881338 --> bukan pencilan
x- 115 = 4.152759 --> bukan pencilan
x- 116 = 7.707567 --> bukan pencilan
x- 117 = 2.85065 --> bukan pencilan
x- 118 = 4.36935 --> bukan pencilan
x- 119 = 1.363828 --> bukan pencilan
x- 120 = 1.593414 --> bukan pencilan
x- 121 = 1.343982 --> bukan pencilan
x- 122 = 1.757274 --> bukan pencilan
x- 123 = 0.992556 --> bukan pencilan
x- 124 = 3.771746 --> bukan pencilan
x- 125 = 7.60513 --> bukan pencilan
x- 126 = 10.55314 --> bukan pencilan
x- 127 = 4.408774 --> bukan pencilan
x- 128 = 5.346334 --> bukan pencilan
x- 129 = 1.901746 --> bukan pencilan
x- 130 = 1.417409 --> bukan pencilan
x- 131 = 2.618426 --> bukan pencilan
x- 132 = 5.948061 --> bukan pencilan
```

Lampiran 8. (Lanjutan)

```
x- 133 = 9.751434 --> bukan pencilan
x- 134 = 1.406132 --> bukan pencilan
x- 135 = 2.356152 --> bukan pencilan
x- 136 = 3.828443 --> bukan pencilan
x- 137 = 5.59947 --> bukan pencilan
x- 138 = 11.59043 --> bukan pencilan
x- 139 = 6.697419 --> bukan pencilan
x- 140 = 5.329527 --> bukan pencilan
x- 141 = 3.709287 --> bukan pencilan
x- 142 = 1.440949 --> bukan pencilan
x- 143 = 3.413239 --> bukan pencilan
x- 144 = 2.310296 --> bukan pencilan
x- 145 = 7.372948 --> bukan pencilan
x- 146 = 4.868659 --> bukan pencilan
x- 147 = 2.66717 --> bukan pencilan
x- 148 = 5.217969 --> bukan pencilan
x- 149 = 6.590799 --> bukan pencilan
x- 150 = 1.912753 --> bukan pencilan
x- 151 = 2.972542 --> bukan pencilan
x- 152 = 2.96528 --> bukan pencilan
x- 153 = 4.11894 --> bukan pencilan
x- 154 = 1.733827 --> bukan pencilan
x- 155 = 9.06528 --> bukan pencilan
x- 156 = 10.51987 --> bukan pencilan
x- 157 = 7.672255 --> bukan pencilan
x- 158 = 4.711986 --> bukan pencilan
x- 159 = 1.218863 --> bukan pencilan
x- 160 = 3.901874 --> bukan pencilan
x- 161 = 10.1265 --> bukan pencilan
x- 162 = 2.748407 --> bukan pencilan
x- 163 = 9.005035 --> bukan pencilan
x- 164 = 17.18919 --> pencilan
x- 165 = 7.756285 --> bukan pencilan
x- 166 = 16.43843 --> pencilan
x- 167 = 9.09356 --> bukan pencilan
x- 168 = 4.556666 --> bukan pencilan
x- 169 = 2.554843 --> bukan pencilan
x- 170 = 0.9343641 --> bukan pencilan
x- 171 = 15.58512 --> pencilan
x- 172 = 3.206036 --> bukan pencilan
x- 173 = 4.036187 --> bukan pencilan
x- 174 = 24.36485 --> pencilan
x- 175 = 3.97227 --> bukan pencilan
x- 176 = 7.029423 --> bukan pencilan
x- 177 = 2.258643 --> bukan pencilan
```

Lampiran 8. (Lanjutan)

```
x- 178 = 5.698873 --> bukan pencilan
x- 179 = 9.334963 --> bukan pencilan
x- 180 = 9.857915 --> bukan pencilan
x- 181 = 4.918234 --> bukan pencilan
x- 182 = 1.406132 --> bukan pencilan
x- 183 = 9.353105 --> bukan pencilan
x- 184 = 5.096452 --> bukan pencilan
x- 185 = 1.013768 --> bukan pencilan
x- 186 = 5.002972 --> bukan pencilan
x- 187 = 2.434243 --> bukan pencilan
x- 188 = 21.36972 --> pencilan
x- 189 = 6.091292 --> bukan pencilan
x- 190 = 1.887848 --> bukan pencilan
x- 191 = 8.261426 --> bukan pencilan
x- 192 = 14.03157 --> bukan pencilan
x- 193 = 2.875879 --> bukan pencilan
x- 194 = 1.878528 --> bukan pencilan
x- 195 = 2.301888 --> bukan pencilan
x- 196 = 6.071577 --> bukan pencilan
x- 197 = 8.535084 --> bukan pencilan
x- 198 = 3.934746 --> bukan pencilan
x- 199 = 3.298025 --> bukan pencilan
x- 200 = 6.098662 --> bukan pencilan
x- 201 = 4.671386 --> bukan pencilan
x- 202 = 1.115203 --> bukan pencilan
x- 203 = 5.141861 --> bukan pencilan
x- 204 = 3.182624 --> bukan pencilan
x- 205 = 1.430414 --> bukan pencilan
x- 206 = 1.511167 --> bukan pencilan
x- 207 = 8.131676 --> bukan pencilan
x- 208 = 4.4635 --> bukan pencilan
x- 209 = 3.773689 --> bukan pencilan
x- 210 = 3.895821 --> bukan pencilan
x- 211 = 11.87688 --> bukan pencilan
x- 212 = 10.7425 --> bukan pencilan
x- 213 = 2.638155 --> bukan pencilan
x- 214 = 1.374148 --> bukan pencilan
x- 215 = 4.429781 --> bukan pencilan
x- 216 = 1.83111 --> bukan pencilan
x- 217 = 4.096527 --> bukan pencilan
x- 218 = 5.593291 --> bukan pencilan
x- 219 = 5.352352 --> bukan pencilan
x- 220 = 2.124904 --> bukan pencilan
x- 221 = 7.261482 --> bukan pencilan
x- 222 = 5.424089 --> bukan pencilan
```

Lampiran 8. (Lanjutan)

```
x- 223 = 2.406377 --> bukan pencilan
x- 224 = 5.218489 --> bukan pencilan
x- 225 = 8.792117 --> bukan pencilan
x- 226 = 5.074421 --> bukan pencilan
x- 227 = 19.83259 --> pencilan
x- 228 = 2.553763 --> bukan pencilan
x- 229 = 7.001665 --> bukan pencilan
x- 230 = 1.430414 --> bukan pencilan
x- 231 = 3.37703 --> bukan pencilan
x- 232 = 7.957727 --> bukan pencilan
x- 233 = 4.930023 --> bukan pencilan
x- 234 = 16.69107 --> pencilan
x- 235 = 7.746423 --> bukan pencilan
x- 236 = 16.69107 --> pencilan
x- 237 = 8.011609 --> bukan pencilan
x- 238 = 3.612687 --> bukan pencilan
x- 239 = 16.69107 --> pencilan
x- 240 = 3.871542 --> bukan pencilan
x- 241 = 7.618914 --> bukan pencilan
x- 242 = 4.854416 --> bukan pencilan
x- 243 = 13.1535 --> bukan pencilan
x- 244 = 7.287795 --> bukan pencilan
x- 245 = 2.446769 --> bukan pencilan
x- 246 = 22.36761 --> pencilan
```

Lampiran 9. Koefisien Diskriminan Linier *Robust, Cutting point* dan Ketepatan Klasifikasi Setiap *Breakdown point*

- *Breakdown point* 0.15
  Koefisien diskriminan linier *robust*:
  β_{s1} = -0.0878 β_{s4} = -0.0541
  β_{s2} = -0.7887 β_{s5} = 0.4668
  β_{s3} = 0.0518 β_{s6} = 0.2712
  Cutting point = 3.9236

Aktual	Prediksi		Total
	1	2	
1	48	8	56
2	4	2	6
Total	52	10	62

$APER = \frac{(4+8)}{62} = 0.1935 = 19.35\%$

Ketepatan klasifikasi = 100% - 19.35% = 80.65%

- *Breakdown point* 0.25
  Koefisien diskriminan linier *robust*:
  β_{s1} = -0.0990 β_{s4} = -0.0690
  β_{s2} = -0.9540 β_{s5} = 0.4695
  β_{s3} = 0.0540 β_{s6} = 0.3206
  Cutting point = 3.3404

Aktual	Prediksi		Total
	1	2	
1	47	9	56
2	4	2	6
Total	51	11	62

$APER = \frac{(4+9)}{62} = 0.2097 = 20.97\%$

Ketepatan klasifikasi = 100% - 20.97% = 79.03%

Lampiran 9. (Lanjutan)

- *Breakdown point* 0.50
  Koefisien diskriminan linier *robust*:
  β_{s1} = -0.1346 β_{s4} = 0.1407
  β_{s2} = -0.5245 β_{s5} = 0.5781
  β_{s3} = -0.0457 β_{s6} = 0.3845
  Cutting point = 4.2996

Aktual	Prediksi		Total
	1	2	
1	41	15	56
2	4	2	6
Total	45	17	62

$APER = \frac{(4+15)}{62} = 0.3065 = 30.65\%$

Ketepatan klasifikasi = 100% - 30.65% = 69.35%

Lampiran 10. Nilai c , b , d_i^2 *robust* dan $\hat{s}$ untuk *Training set* Kelompok 1

```
c
    9.7392
b
    2.7559

Robust Mahalanobis Distance
  Columns 1 through 7
 1.0787   2.2481   1.9144   3.1386   2.8441   3.4267   2.3330

  Columns 8 through 14
 1.9474   1.5283   2.3804   1.9426   3.4508   1.6789   2.3920

  Columns 15 through 21
 1.2177   3.6908   4.5875   2.7845   3.0902   6.7514   1.8679

  Columns 22 through 28
 1.9480   1.8464   1.6083   2.4020   1.9224   2.4617   1.1992

  Columns 29 through 35
 5.4729   0.9517   4.2388   3.2516   3.6753   1.7727   3.1681

  Columns 36 through 42
 2.6548   1.0927   8.0503   3.4237   1.4432   1.1712   2.4156

  Columns 43 through 49
 2.5363   2.2675   0.9796   2.0888   1.4147   3.3201   1.6466

  Columns 50 through 56
 2.9599   1.3041   1.1750   1.6158   3.0868   1.8217   1.9833

  Columns 57 through 63
 2.0477   2.1541   3.1199   2.6432   1.8106   1.3692   1.8345

  Columns 64 through 70
 2.5228   3.9996   1.6544   1.3482   2.5856   1.7223   1.9586

  Columns 71 through 77
 1.8317   1.3661   2.0579   1.6378   1.9628   7.5953   0.8635

  Columns 78 through 84
 2.2409   2.9632   0.9906   2.8128   1.3410   1.5231   3.5621

  Columns 85 through 91
 2.9836   1.6392   1.9902   1.9895   2.7133   2.6256   2.2755

  Columns 92 through 98
 2.5029   2.1938   1.1234   1.4682   2.5295   1.1102   2.1255
```

Lampiran 10. (Lanjutan)

```
  Columns 99 through 105
1.5734   4.1344   2.7728   1.6750   1.0445   1.8722   1.6636

  Columns 106 through 112
1.0490   2.1507   1.8506   1.7404   1.7516   1.5736   1.7545

  Columns 113 through 119
2.9917   1.6155   2.0348   2.8963   1.7118   2.3549   1.1720

  Columns 120 through 126
1.1863   1.0976   1.3226   1.0337   1.9096   3.6106   3.4482

  Columns 127 through 133
2.2389   2.4704   1.3275   1.1612   1.5706   2.3475   3.0206

  Columns 134 through 140
1.1486   1.6512   1.8600   2.3017   3.9426   2.7868   2.3338

  Columns 141 through 147
1.9292   1.1362   1.8709   1.5577   2.8182   2.1599   1.6366

  Columns 148 through 154
2.2927   2.6840   1.5569   1.7648   2.1594   2.6153   1.2333

  Columns 155 through 161
3.0784   3.2897   3.3499   2.1391   1.0243   2.3648   3.3333

  Columns 162 through 168
1.7747   3.0769   4.7055   3.2844   3.9964   3.3189   2.2037

  Columns 169 through 175
1.6839   1.0042   5.1890   1.8108   2.0985   5.3930   2.0025

  Columns 176 through 182
2.8402   1.6099   2.3898   3.1432   3.0227   2.1446   1.1486

  Columns 183 through 189
3.0694   2.3731   1.0080   2.2576   1.4945   4.6774   2.6687

  Columns 190 through 196
1.2782   3.0698   3.8776   1.6317   1.3248   1.5958   2.5534

  Columns 197 through 203
2.8716   1.9790   2.1500   2.3894   2.2504   0.9723   2.1713

  Columns 204 through 210
1.7764   1.1550   1.3038   2.9428   2.2510   1.9521   1.9740

  Columns 211 through 217
3.5824   3.5569   1.5482   1.2297   2.1429   1.7567   1.9466
```

Lampiran 10. (Lanjutan)

```
Columns 218 through 220
 2.6734   2.4150   1.3836

Robust scale
    1.9384
```

Lampiran 11. Fungsi Pengaruh $\psi(d_i)$ untuk *Training set* Kelompok 1

```
psi
  Columns 1 through 7
 2.0399   3.9057   3.4296   4.8858   4.6130   5.0995   4.0181

  Columns 8 through 14
 3.4790   2.8184   4.0793   3.4719   5.1150   3.0638   4.0942

  Columns 15 through 21
 2.2872   5.2470   5.3842   4.5512   4.8446   3.5307   3.3593

  Columns 22 through 28
 3.4800   3.3265   2.9498   4.1069   3.4416   4.1815   2.2546

  Columns 29 through 35
 4.9667   1.8096   5.3985   4.9761   5.2396   3.2123   4.9102

  Columns 36 through 42
 4.4098   2.0650   1.5654   5.0976   2.6760   2.2050   4.1241

  Columns 43 through 49
 4.2721   3.9318   1.8607   3.6849   2.6277   5.0268   3.0120

  Columns 50 through 56
 4.7265   2.4380   2.2118   2.9620   4.8416   3.2884   3.5322

  Columns 57 through 63
 3.6260   3.7769   4.8702   4.3966   3.2713   2.5502   3.3081

  Columns 64 through 70
 4.2561   5.3583   3.0245   2.5141   4.3304   3.1329   3.4956

  Columns 71 through 77
 3.3038   2.5448   3.6408   2.9976   3.5019   2.2592   1.6476

  Columns 78 through 84
 3.8959   4.7297   1.8807   4.5807   2.5020   2.8098   5.1811

  Columns 85 through 91
 4.7488   3.0000   3.5423   3.5412   4.4748   4.3767   3.9425

  Columns 92 through 98
 4.2320   3.8319   2.1200   2.7180   4.2639   2.0965   3.7370

  Columns 99 through 105
 2.8929   5.3860   4.5387   3.0575   1.9784   3.3658   3.0393

  Columns 106 through 112
 1.9865   3.7723   3.3328   3.1615   3.1792   2.8930   3.1838
```

Lampiran 11. (Lanjutan)

```
Columns 113 through 119
 4.7563   2.9615   3.6074   4.6651   3.1163   4.0467   2.2064

  Columns 120 through 126
 2.2319   2.0740   2.4700   1.9588   3.4224   5.2074   5.1133

  Columns 127 through 133
 3.8933   4.1923   2.4785   2.1874   2.8881   4.0371   4.7829

  Columns 134 through 140
 2.1650   3.0193   3.3473   3.9771   5.3428   4.5536   4.0192

  Columns 141 through 147
 3.4518   2.1430   3.3639   2.8669   4.5863   3.7850   2.9957

  Columns 148 through 154
 3.9652   4.4424   2.8658   3.1999   3.7846   4.3649   2.3145

  Columns 155 through 161
 4.8343   5.0047   5.0480   3.7560   1.9417   4.0595   5.0362

  Columns 162 through 168
 3.2154   4.8332   5.3598   5.0009   5.3575   5.0260   3.8455

  Columns 169 through 175
 3.0718   1.9054   5.1582   3.2715   3.6988   5.0258   3.5603

  Columns 176 through 182
 4.6088   2.9525   4.0913   4.8897   4.7848   3.7638   2.1650

  Columns 183 through 189
 4.8265   4.0700   1.9123   3.9185   2.7621   5.3666   4.4254

  Columns 190 through 196
 2.3929   4.8269   5.3223   2.9878   2.4738   2.9294   4.2924

  Columns 197 through 203
 4.6406   3.5259   3.7714   4.0908   3.9087   1.8473   3.8009

  Columns 204 through 210
 3.2181   2.1764   2.4374   4.7102   3.9096   3.4860   3.5185

  Columns 211 through 217
 5.1922   5.1781   2.8513   2.3084   3.7614   3.1874   3.4778

  Columns 218 through 220
 4.4307   4.1232   2.5748
```

Lampiran 12. Fungsi Pembobot $w(d_i)$ untuk *Training set* Kelompok 1

```
u
Columns 1 through 7
 0.9756   0.8963   0.9242   0.8031   0.8367   0.7677   0.8885

  Columns 8 through 14
 0.9216   0.9514   0.8841   0.9220   0.7647   0.9415   0.8830

  Columns 15 through 21
 0.9690   0.7334   0.6055   0.8432   0.8088   0.2698   0.9278

  Columns 22 through 28
 0.9216   0.9294   0.9462   0.8820   0.9236   0.8763   0.9699

  Columns 29 through 35
 0.4682   0.9810   0.6570   0.7895   0.7355   0.9348   0.7996

  Columns 36 through 42
 0.8569   0.9750   0.1003   0.7681   0.9566   0.9713   0.8807

  Columns 43 through 49
 0.8690   0.8945   0.9799   0.9101   0.9582   0.7811   0.9436

  Columns 50 through 56
 0.8238   0.9645   0.9711   0.9457   0.8092   0.9313   0.9188

  Columns 57 through 63
 0.9135   0.9046   0.8053   0.8581   0.9321   0.9609   0.9303

  Columns 64 through 70
 0.8703   0.6911   0.9431   0.9620   0.8640   0.9384   0.9208

  Columns 71 through 77
 0.9305   0.9610   0.9127   0.9442   0.9204   0.1534   0.9843

  Columns 78 through 84
 0.8969   0.8234   0.9794   0.8401   0.9624   0.9517   0.7503

  Columns 85 through 91
 0.8211   0.9441   0.9182   0.9183   0.8508   0.8599   0.8938

  Columns 92 through 98
 0.8723   0.9011   0.9736   0.9551   0.8696   0.9742   0.9070

  Columns 99 through 105
 0.9485   0.6721   0.8445   0.9417   0.9771   0.9275   0.9425

  Columns 106 through 112
 0.9769   0.9048   0.9291   0.9372   0.9364   0.9485   0.9361
```

Lampiran 12. (Lanjutan)

```
Columns 113 through 119
 0.8202   0.9457   0.9146   0.8309   0.9392   0.8865   0.9712

  Columns 120 through 126
 0.9705   0.9748   0.9635   0.9776   0.9246   0.7440   0.7650

  Columns 127 through 133
 0.8971   0.8755   0.9632   0.9718   0.9487   0.8872   0.8169

  Columns 134 through 140
 0.9724   0.9433   0.9284   0.8914   0.6991   0.8429   0.8885

  Columns 141 through 147
 0.9231   0.9730   0.9276   0.9495   0.8395   0.9041   0.9443

  Columns 148 through 154
 0.8922   0.8539   0.9495   0.9354   0.9041   0.8610   0.9682

  Columns 155 through 161
 0.8102   0.7848   0.7774   0.9059   0.9780   0.8856   0.7794

  Columns 162 through 168
 0.9347   0.8103   0.5876   0.7855   0.6916   0.7812   0.9002

  Columns 169 through 175
 0.9411   0.9788   0.5128   0.9321   0.9093   0.4808   0.9172

  Columns 176 through 182
 0.8371   0.9461   0.8832   0.8025   0.8166   0.9054   0.9724

  Columns 183 through 189
 0.8112   0.8848   0.9787   0.8954   0.9535   0.5919   0.8555

  Columns 190 through 196
 0.9659   0.8112   0.7081   0.9447   0.9633   0.9470   0.8673

  Columns 197 through 203
 0.8337   0.9191   0.9049   0.8832   0.8961   0.9802   0.9031

  Columns 204 through 210
 0.9346   0.9721   0.9645   0.8257   0.8960   0.9213   0.9195

  Columns 211 through 217
 0.7477   0.7510   0.9501   0.9684   0.9055   0.9360   0.9217

  Columns 218 through 220
 0.8550   0.8808   0.9600
```

Lampiran 13. Nilai c , b , d_i^2 *robust* dan $\hat{s}$ untuk *Training set* Kelompok 2

```
c
    9.7392

b
    2.7559

Robust Mahalanobis Distance
  Columns 1 through 7
 2.1324   2.6149   0.8355   2.3046   2.5612   3.0685   3.8892

  Columns 8 through 14
 1.4295   2.1738   1.4092   2.1490   3.2022   2.1529   2.4511

  Columns 15 through 21
 2.9323   2.4511   2.3893   2.0401   2.4511   1.9336   2.6572

  Columns 22 through 26
 1.4991   2.8042   1.9457   1.7103   3.8924

Robust scale
    2.0697
```

Lampiran 14. Fungsi Pengaruh $\psi(d_i)$ untuk *Training set* Kelompok 2

```
psi
  Columns 1 through 7
 4.0004   4.6599   1.7039   4.2505   4.5930   5.1526   5.6868

  Columns 8 through 14
 2.8324   4.0619   2.7957   4.0252   5.2720   4.0309   4.4507

  Columns 15 through 21
 5.0185   4.4507   4.3677   3.8600   4.4507   3.6927   4.7112

  Columns 22 through 26
 2.9573   4.8815   3.7119   3.3249   5.6881
```

Lampiran 15. Fungsi Pembobot $w(d_i)$ untuk *Training set* Kelompok 2

```
u
  Columns 1 through 7
 0.9064   0.8610   0.9853   0.8912   0.8665   0.8113   0.7065

  Columns 8 through 14
 0.9574   0.9028   0.9586   0.9050   0.7955   0.9047   0.8773

  Columns 15 through 21
 0.8269   0.8773   0.8833   0.9142   0.8773   0.9227   0.8567

  Columns 22 through 26
 0.9532   0.8411   0.9218   0.9393   0.7060
```

Lampiran 16. Penduga Parameter *Robust* dan Koefisien Diskriminan Linier *Robust*

```
mu1
   26.3114    0.8047   51.5504   25.0487   11.3170    2.5258

mu2
   28.5494    1.4148   49.7353   24.8239   11.1682    2.5514

cov1
   25.2319    3.6861   10.3459    2.8208    0.0760   11.0903
    3.6861    1.0606    2.4344    0.6309   -0.0039    2.3232
   10.3459    2.4344   88.2912   19.6887   -0.0021    5.3209
    2.8208    0.6309   19.6887    6.9146   -0.1404    1.6248
    0.0760   -0.0039   -0.0021   -0.1404    0.2489    0.0215
   11.0903    2.3232    5.3209    1.6248    0.0215   10.1335

cov2
   42.5072    6.7775    1.2740   -0.7323   -1.5957   10.7098
    6.7775    2.8132    3.0274    0.6409   -0.2539    2.8126
    1.2740    3.0274   66.2531   13.6944    0.4465   -4.1972
   -0.7323    0.6409   13.6944    6.1490    0.0465   -0.5628
   -1.5957   -0.2539    0.4465    0.0465    0.1819   -0.3804
   10.7098    2.8126   -4.1972   -0.5628   -0.3804    7.7449

sigma
   27.0578    4.0128    9.3871    2.4453   -0.1007   11.0501
    4.0128    1.2458    2.4971    0.6320   -0.0304    2.3749
    9.3871    2.4971   85.9619   19.0552    0.0453    4.3149
    2.4453    0.6320   19.0552    6.8337   -0.1207    1.3936
   -0.1007   -0.0304    0.0453   -0.1207    0.2418   -0.0210
   11.0501    2.3749    4.3149    1.3936   -0.0210    9.8810

linear   : [-0.0878 -0.7887 0.0518 -0.0541 0.4668 0.2712]
constant : 3.9236
```

Lampiran 17. Prediksi Pengklasifikasian Obyek pada *Testing set*

No.	X_1	X_2	X_3	X_4	X_5	X_6	Aktual	Skor	Prediksi
1	22	0	46	24	10.8	0	1	4.1942	1
2	24	0	45	24	11.4	0	1	4.2469	1
3	23	0	79	24	12.3	0	1	6.516	1
4	18	0	80	30	11.4	0	1	6.2621	1
5	31	1	55	26	12.5	5	1	5.1229	1
6	22	0	45	24	22.6	0	1	9.6507	1
7	26	2	49	25	10.8	1	1	2.6381	2
8	24	1	56	24	11.2	4	1	5.0195	1
9	38	3	52	29	10.8	8	1	2.6332	2
10	21	0	68	24	11.7	0	1	5.8418	1
11	25	1	50	23.5	11	2	1	4.0122	1
12	31	1	48	25	11	5	1	4.1142	1
13	20	0	51	24	11.8	0	1	5.0956	1
14	36	1	56	25.2	11.3	4	1	3.9476	1
15	26	2	45	23.5	11.3	1.5	1	2.8811	2
16	21	1	47	24	11	2.5	1	4.3165	1
17	24	0	50	24.5	11	0	1	4.2922	1
18	37	1	56	29	11	5	1	3.7854	2
19	35	1	47	24	11	8	1	4.5789	1
20	29	1	61	32	11.4	5	1	4.7712	1
21	24	0	47	23.5	10.8	0	1	4.0975	1
22	21	0	46	22.5	11.2	0	1	4.5499	1
23	36	2	42	24.1	11.8	11	1	4.625	1
24	24	1	70	31	11	4	1	5.2726	1
25	28	1	50	27	11	11	1	6.0002	1
26	25	1	50	23.5	11.2	4	1	4.6479	1
27	21	0	48	22.5	11.4	0	1	4.7469	1
28	34	2	47	23.6	11.5	6	1	3.5906	2

Lampiran 17. (Lanjutan)

No.	X_1	X_2	X_3	X_4	X_5	X_6	Aktual	Skor	Prediksi
29	29	1	59	26	11	6	1	5.0767	1
30	28	1	46	24	12	4	1	4.5237	1
31	22	0	45	25	11.6	0	1	4.4618	1
32	27	1	65	31	11.8	4	1	5.1236	1
33	18	0	42.5	23	10.2	0	1	4.1382	1
34	21	0	48	23.5	11	0	1	4.5061	1
35	23	1	51	22	11	0.7	1	3.9681	1
36	20	0	38	23.5	11	0	1	4.0759	1
37	28	1	49.5	24	11	6	1	4.7806	1
38	20	0	48.5	22.5	11	0	1	4.6739	1
39	27	1	52	24	11	5	1	4.7267	1
40	22	0	46	21	11.5	0	1	4.6833	1
41	24	1	44	22.5	10.8	6	1	4.8347	1
42	22	0	41	20	11.8	0	1	4.6184	1
43	21	1	52	23.5	11	0	1	3.9246	1
44	27	1	57	25	10.4	5	1	4.6515	1
45	32	0	55	28	11	0	1	3.6594	2
46	26	0	59	30	11.4	0	1	4.4719	1
47	30	2	45.5	24.5	11	3	1	2.7685	2
48	21	0	80	32	11.6	0	1	5.9839	1
49	21	0	50	24	11.8	0	1	4.956	1
50	32	2	48	23	11.4	4	1	3.2614	2
51	29	1	43	23.4	11.7	10	1	5.8001	1
52	32	2	54	25	11	7	1	4.0909	1
53	22	1	75	28	11.4	1.2	1	5.2969	1
54	24	1	70	27	11.4	4	1	5.6757	1
55	26	0	53	25	12	0	1	4.7117	1
56	30	2	40	24	11.2	10	1	4.5024	1

Lampiran 17. (Lanjutan)

No.	X_1	X_2	X_3	X_4	X_5	X_6	Aktual	Skor	Prediksi
57	26	0	38	22	11	0	2	3.6302	2
58	24	0	31	18	11	0	2	3.6596	2
59	24	0	47	24	11	0	2	4.1638	1
60	24	0	45.5	21	11.8	0	2	4.6218	1
61	25	0	50	25	12	0	2	4.6441	1
62	20	0	45	24	11	0	2	4.4114	1

Lampiran 18. *Syntax* Pengujian Sebaran Normal Multivariat

Source Code	Keterangan
`d<-read.delim("e:\\syntax multivariate\\multivariate normal\\data.txt")`	Memanggil data dalam sub file "multivariate normal" dalam file "syntax multivariate" yang terletak pada Local Disk (E:).
`source("e:\\syntax multivariate\\multivariate normal\\input data.txt")`	Memanggil fungsi input data dalam bentuk matriks.
`source("e:\\syntax multivariate\\multivariate normal\\mu.txt")`	Memanggil fungsi perhitungan vektor rata-rata ($\underline{\boldsymbol{\mu}}$).
`source("e:\\syntax multivariate\\multivariate normal\\input x.mu.txt")`	Memanggil fungsi perhitungan $\left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)$
`source("e:\\syntax multivariate\\multivariate normal\\covariance.txt")`	Memanggil fungsi perhitungan matriks ragam peragam ($\boldsymbol{\Sigma}$).
`source("e:\\syntax multivariate\\multivariate normal\\mahalanobis.txt")`	Memanggil fungsi perhitungan jarak *Mahalanobis* $d_i^2 = \left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)^T \boldsymbol{\Sigma}^{-1} \left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)$
`source("e:\\syntax multivariate\\multivariate normal\\chis.txt")`	Memanggil fungsi perhitungan nilai $\chi^2_{p,\alpha}$
`source("e:\\syntax multivariate\\multivariate normal\\qqplot.txt")`	Memanggil fungsi Q-Q plot (plot antara d_i^2 dengan χ^2).

`data=input.data(d)`
`mu=input.mu(data)`
`x.mu=input.xmu(data,mu)`
`cova=covariance(d)`
`mahalanobis=mahalanobis(d,x.mu,cova)`
`chi.sq=chi(d)`
`qqplot=grafikqq(mahalanobis,chi.sq)`
	Menjalankan semua fungsi yang telah dipanggil sebelumnya.

Lampiran 18a. Fungsi input data dalam bentuk matriks

```
input.data=function(d)
{
n=nrow(d)
m=ncol(d)

d=t(t(d))
data=matrix(d,n,m)
}
```

Lampiran 18b. Fungsi perhitungan vektor rata-rata ($\underline{\mu}$)

```
input.mu=function(data)
{
m=ncol(data)
mu=array(c(1),m)
for (i in 1:m)
{
mu[i]=mean(data[ ,i])}
i=1:m
mu=mu[i]
mu=t(mu)
}
```

Lampiran 18c. Fungsi perhitungan $\left(\underline{x}_i - \underline{\mu}\right)$

```
input.xmu=function(d,mu)
{
n=nrow(d)
m=ncol(d)
DATA=matrix(t(data))
i=0
j=0
l=0
k=c()
M=c()
v=c()
z=matrix()
```

Lampiran 18c. (Lanjutan)

```
for (j in 1:n) {
a1=(1+m*(j-1))
a2= (m+m*(j-1))
v[[l+j]]=c(t(DATA[a1:a2])-mu)
}
M=unlist(v)
x.mu=matrix(M,m,n)
x.mu=t(x.mu)
}
```

Lampiran 18d. Fungsi perhitungan *inverse* dari matriks ragam peragam (Σ^{-1})

```
covariance=function(d)
{
cova=cov(d)
cova.1=solve(cova)
}
```

Lampiran 18e. Fungsi perhitungan jarak *Mahalanobis* (d_i^2)

```
mahalanobis=function(d,x.mu,cova.1)
{
n=nrow(d)
for(i in 1:n)
{
x.mu[i,1:6]=x.mu[i,1:6]}

i=1:n
md.2=x.mu[i,1:6]%*%cova.1%*%t(x.mu[i,1:6])
md.2=t(t(sort(diag(md.2))))
}
```

Lampiran 18f. Fungsi perhitungan nilai $\chi^2_{p,\alpha}$

```
chi=function(d)
{
n=nrow(d)
m=ncol(d)
alpha=array(c(1),n)
for (i in 1:n)
{
alpha[i]=(i-0.5)/n}
i=1:n
alpha=alpha[i]

chi.sq=qchisq(c(alpha), df=m, lower.tail=FALSE)
chi.sq=t(t(chi.sq))
}
```

Lampiran 18g. Fungsi Q-Q plot

```
grafikqq=function(md.2,chi.sq)
{
qqplot(md.2,chi.sq,xlab="Mahalanobis
Distance",ylab="Chi_Square", main= "Normality Test -
QQ Plot")
}
```

Lampiran 19. *Syntax* Identifikasi Pencilan

Source Code	Keterangan
`d<-read.delim("e:\\syntax multivariate\\multivariate outlier\\data.txt")`	Memanggil data dalam sub file "multivariate outlier" dalam file "syntax multivariate" yang terletak pada Local Disk (E:).
`source("e:\\syntax multivariate\\multivariate outlier\\input data.txt")`	Memanggil fungsi input data dalam bentuk matriks.
`source("e:\\syntax multivariate\\multivariate outlier\\mu.txt")`	Memanggil fungsi perhitungan vektor rata-rata ($\underline{\boldsymbol{\mu}}$).
`source("e:\\syntax multivariate\\multivariate outlier\\input x.mu.txt")`	Memanggil fungsi perhitungan $\left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)$
`source("e:\\syntax multivariate\\multivariate outlier\\covariance.txt")`	Memanggil fungsi perhitungan matriks ragam peragam ($\boldsymbol{\Sigma}$).
`source("e:\\syntax multivariate\\multivariate outlier\\mahalanobis.txt")`	Memanggil fungsi perhitungan jarak *Mahalanobis* $d_i^2 = \left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)^T \boldsymbol{\Sigma}^{-1} \left(\underline{\boldsymbol{x}}_i - \underline{\boldsymbol{\mu}}\right)$
`source("e:\\syntax multivariate\\multivariate outlier\\decision.txt")`	Memanggil fungsi perhitungan identifikasi pencilan (d_i^2 dibandingkan dengan χ^2).

`data=input.data(d)`
`mu=input.mu(data)`
`x.mu=input.xmu(data,mu)`
`cova=covariance(d)`
`mahalanobis=mahalanobis(d,x.mu,cova)`
`decision=decision(mahalanobis,chi.sq)`
	Menjalankan semua fungsi yang telah dipanggil sebelumnya.

Lampiran 19a. Fungsi input data dalam bentuk matriks

```
input.data=function(d)
{
n=nrow(d)
m=ncol(d)

d=t(t(d))
data=matrix(d,n,m)
}
```

Lampiran 19b. Fungsi perhitungan vektor rata-rata ($\underline{\mu}$)

```
input.mu=function(data)
{
m=ncol(data)
mu=array(c(1),m)
for (i in 1:m)
{
mu[i]=mean(data[ ,i])}
i=1:m
mu=mu[i]
mu=t(mu)
}
```

Lampiran 19c. Fungsi perhitungan $\left(\underline{x}_i - \underline{\mu}\right)$

```
input.xmu=function(d,mu)
{
n=nrow(d)
m=ncol(d)
DATA=matrix(t(data))
i=0
j=0
l=0
k=c()
M=c()
v=c()
z=matrix()
for (j in 1:n)
```

Lampiran 19c. (Lanjutan)

```
{
a1=(1+m*(j-1))
a2= (m+m*(j-1))
v[[l+j]]=c(t(DATA[a1:a2])-mu)
}
M=unlist(v)
x.mu=matrix(M,m,n)
x.mu=t(x.mu)
}
```

Lampiran 19d. Fungsi perhitungan *inverse* dari matriks ragam peragam (Σ^{-1})

```
covariance=function(d)
{
cova=cov(d)
cova.1=solve(cova)
}
```

Lampiran 19e. Fungsi perhitungan jarak *Mahalanobis* (d_i^2)

```
mahalanobis=function(d,x.mu,cova.1)
{
n=nrow(d)
for(i in 1:n)
{
x.mu[i,1:6]=x.mu[i,1:6]}

i=1:n
md.2=x.mu[i,1:6]%*%cova.1%*%t(x.mu[i,1:6])
md.2=t(t(diag(md.2)))
}
```

Lampiran 19f. Fungsi identifikasi pencilan

```
decision=function(md.2,chi.sq)
{
N=nrow(md.2)
m=ncol(data)
chi.sq=qchisq(c(0.025), df=m, lower.tail=FALSE)


cat ("\n")
cat ("Chi.sq =",chi.sq,"\n")
cat ("\n")

for (i in 1:N)
{
if(md.2[i] < chi.sq)
cat ("x-",i,"=",md.2[i],"--> bukan pencilan\n") else
cat ("x-",i,"=",md.2[i],"--> pencilan\n")
}

cat ("\n")
}
```

Lampiran 20. *Syntax* Pendugaan Parameter *Robust* dengan Metode Penduga-S menggunakan Algoritma *SURREAL*

Source Code	Keterangan
`function res = Sm(x,nsamp,bdp)`	Fungsi untuk menjalankan algoritma *SURREAL* penduga-S.
`tol=1e-5;` `s=10e10;`	Batas penduga parameter konsisten.
`[n,p] = size(x);`	Data dalam bentuk matriks **X** berukuran *n* x *p*.
`c = Tbsc(bdp,p);`	Nilai *c* untuk mencapai *breakdown point* (*bdp*) yang diinginkan.
`b = (c/6)* Tbsb(c,p);` `la = 1;`	b adalah nilai harapan dari $\rho(d_i)$ yang diasumsikan menyebar Multivariat Gaussian.
`for loop=1:nsamp`	Dilakukan iterasi/ perulangan hingga *nsamp* kali.
`ranset=randomset(n,p+1);`	Memilih sub contoh acak *p*+1 dari data berukuran *n*.
`xj = x(ranset,:);`	Sebanyak *p*+1 pengamatan yang diambil secara acak.
`mu=mean(xj);`	Menghitung vektor rata-rata.
`xjcenter = xj-repmat(mu,p+1,1);`	Menghitung $\left(\underline{x_i} - \underline{\boldsymbol{\mu}}\right)$
`cov = (xjcenter'*xjcenter)/(p+1);`	Menghitung matriks ragam peragam (**Σ**).
`determ = det(cov);`	Menghitung $\lvert \boldsymbol{\Sigma} \rvert$

`if determ > 1e-15`
`if determ^(1/p)> 1e-5`
`cov = (determ^(-1/p)).*cov;`
`if loop > ceil(nsamp/5)`
`if loop == ceil(nsamp/2)`
`la=2;`
`elseif loop == ceil(nsamp*(0.8))`
`la=4;`
`end`
`random = rand(1)^la;`
`mu = random*mu+(1-random)*mu_opt;`
	Dilakukan iterasi/ perulangan hingga diperoleh $\lvert \boldsymbol{\Sigma} \rvert$ yang paling minimum

Lampiran 20. (Lanjutan)

`cov = random*cov+(1-random)*cov_opt;`
`end`
`determ = det(cov);`
`cov = (determ^(-1/p)).*cov;`
`md = mahalanobis(x,mu,cov,n,p);`
`md = md.^(1/2);`
	Menghitung jarak *Mahalanobis* dengan $\|\boldsymbol{\Sigma}\|$ minimum.

`if mean(rhobiweight(md/s,c)) < b`	Batasan algoritma *SURREAL* $\frac{1}{n}\sum_{i=1}^{n}\rho(d_i) = b$

`if s < 5e10`
`s = sestck(md,s,c,b,tol);`
`else`
`s = sestck(md,0,c,b,tol);`
`end`
	Mencari nilai penduga skala *robust* ($\hat{s}$) minimum.

`mu_opt = mu;`
`cov_opt = cov;`
`md_opt = md;`
	Vektor rata-rata, matriks ragam peragam, dan jarak *Mahalanobis* optimal

`psi = psibiweight(md,s*c);`	Menghitung fungsi pengaruh ($\psi(d_i)$)
`u = psi./md;`	Menghitung fungsi pembobot ($w(d_i)$)
`ubig = repmat(u',1,p);` `mu = mean(ubig.*x)./mean(u);`	Menghitung penduga vektor rata-rata *robust* setiap kelompok, *k*=1,2 ($\underline{\widehat{\boldsymbol{\mu}}}_{Sk}$)

`xcenter = x-repmat(mu,n,1);`
`cov =((ubig.*xcenter)'*xcenter);`
`cov = (det(cov)^(-1/p)).*cov;`
`okay = 0;`
`jj = 1;`
`while ((jj<3) & (okay~=1))`
`jj = jj+1;`
	Menghitung penduga matriks ragam peragam *robust* setiap kelompok, *k*=1,2 ($\widehat{\boldsymbol{\Sigma}}_{Sk}$)

`md =mahalanobis(x,mu,cov,n,p);` `md = md.^(1/2);`	Menghitung jarak *Mahalanobis* yang *robust*

`if mean(rhobiweight(md/s,c))<b`
`mu_opt = mu;`
`cov_opt = cov;`
`md_opt = md;`
`okay = 1;`
`if s < 5e10`
	Jika $\frac{1}{n}\sum_{i=1}^{n}\rho(d_i/\hat{s}) < b$, maka $\underline{\boldsymbol{\mu}}_{\boldsymbol{sk}}$ = vektor rata-rata optimal, $\boldsymbol{\Sigma}_{\boldsymbol{sk}}$ = matriks ragam peragam optimal, d_{is}^2 = jarak *Mahalanobis* optimal dengan

Lampiran 20. (Lanjutan)

`s = sestck(md,s,c,k,tol);`
`else`
`s = sestck(md,0,c,k,tol);`
`end`
`else`
`mu = (mu+mu_opt)/2;`
`cov = (cov+cov_opt)./2;`
`cov = (determ^(-1/p)).*cov;`
`end`
`end`
`end`
`end`
`end`
`end`
`cov_opt=s^2*cov_opt;`
`md_opt=md_opt/s;`
	nilai penduga skala *robust* ($\hat{s}$) minimal.
	
	Jarak *Mahalanobis* dengan nilai penduga skala *robust* ($\hat{s}$) minimal.

`res.mean = mu_opt;`
`res.covariance = cov_opt;`
`res.distances = md_opt;`
`res.scale = s;`
 	
`disp('c');`
`disp([c]);`
`disp('k');`
`disp([k]);`
`disp('psi');`
`disp([psi]);`
`disp('u');`
`disp([u]);`
`fprintf('\nScale = %8.3f\n',s);`
`disp('Robust Mahalanobis Distance');`
`disp([md_opt]);`
`disp('Robust scale');`
`disp([s]);`
	Keluaran hasil perhitungan algoritma *SURREAL* yaitu $\underline{\boldsymbol{\mu}}_{\boldsymbol{sk}}$, $\boldsymbol{\Sigma}_{\boldsymbol{sk}}$, d_{is}^2 , dan $\hat{s}$.
	
	Menampilkan nilai-nilai yang ingin diketahui selain keluaran hasil perhitungan algoritma *SURREAL*.

Lampiran 20a. Fungsi perhitungan jarak *Mahalanobis*

```
function mah = mahalanobis(dat,meanvct,covmat,n,p)

for k=1:p
    d = covmat(k,k);
    covmat(k,:) = covmat(k,:)/d;
    rows = setdiff(1:p,k);
    b = covmat(rows,k);
    covmat(rows,:) = covmat(rows,:)-b*covmat(k,:);
    covmat(rows,k) = -b/d;
    covmat(k,k) = 1/d;
end

hlp = dat-repmat(meanvct,n,1);
mah = sum(hlp*covmat.*hlp,2)';
```

Lampiran 20b. Fungsi perhitungan $\psi(d_i)$ dengan *c* konstan untuk semua

```
function psi = psibiweight(x,c)

hulp = x-2.*x.^3/(c^2)+x.^5/(c^4);
psi = hulp.*(abs(x)<c);
```

Lampiran 20c. Fungsi untuk pengambilan sub contoh secara acak sebanyak *p*+1 dari data berukuran n.

```
function ranset = randomset(tot,nel)

ranper=randperm(tot);
ranset=ranper(1:nel);
```

Lampiran 20d. Fungsi perhitungan $\rho(d_i)$ dengan *c* konstan untuk semua

```
function rho = rhobiweight(x,c)

hulp = x.^2/2-x.^4/(2*c^2)+x.^6/(6*c^4);
rho = hulp.*(abs(x)<c)+c^2/6.*(abs(x)>=c);
```

Lampiran 20e. Fungsi perhitungan penduga skala *robust* ($\hat{s}$)

```
function scale = sestck(x,start,c,k,tol)

% Computes Tukey's biweight objectief function (scale)
corresponding
% with the mahalanobis distances x.

if start>0
    s = start;
else y=abs(x);
    [n,p]=size(x);
    if  ((n==1) & (p>1))
        x = x'; [n,p] = size(x);
    end
    x = sort(x);
    if  floor(n/2) == n/2
        s = (x(n/2,:) + x((n+2)/2,:))/2;
    else
        s = x((n+1)/2,:);
    end
    s = s/0.6745;
end
crit = 2*tol;
rhoold = mean(rhobiweight(x/s,c))-k;
while crit >= tol
    delta =
rhoold/mean(psibiweight(x/s,c).*(x/(s^2)));
    isqu = 1; okay = 0;
    while ((isqu<10) & (okay~=1))
        rhonew = mean(rhobiweight(x/(s+delta),c))-k;
        if abs(rhonew) < abs(rhoold)
            s = s+delta;
            okay = 1;
        else
```

Lampiran 20e. (Lanjutan)

```
            delta = delta/2;
            isqu = isqu+1;
        end
    end
    if isqu == 10
        crit = 0;
    else
        crit = (abs(rhoold)-
abs(rhonew))/(max([abs(rhonew),tol]));
    end
    rhoold = rhonew;
end
scale = abs(s);
```

Lampiran 20f. Fungsi perhitungan nilai *b* dengan asumsi sebaran Multivariat Gaussian

```
function res = Tbsb(c,p);
y1 = ksiint(c,1,p)*3/c-
ksiint(c,2,p)*3/(c^3)+ksiint(c,3,p)/(c^5);
y2 = c*(1-chi2cdf(c^2,p));
res = y1+y2;
```

Lampiran 20g. Fungsi perhitungan konstanta *c*

```
function res = Tbsc(alpha,p);

% constant for Tukey Biweight S

talpha = sqrt(chi2inv(1-alpha,p));
maxit = 13;
eps = 1e-8;
diff = 1e6;
ctest = talpha;
iter = 1;
while ((diff>eps) & iter<maxit);
    c = ctest;
    ctest = Tbsb(c,p)/alpha;
    diff = abs(c-ctest);
    iter = iter+1;
end;
res = (ctest);
```

Lampiran 21. *Syntax* Klasifikasi Ulang Obyek

Source Code	Keterangan
`function res = da(data,'s','l')`	Fungsi untuk melakukan klasifikasi obyek dengan analisis diskriminan linier *robust*.
`nsamp=800;`	Menetapkan banyak iterasi/ perulangan.
`bdp=0.15;`	Memasukkan besar *breakdown point* yang diinginkan.
`[n,p] = size(data);`	Input data dalam bentuk matriks berukuran *nxp*.

`dim = p-1;`
`X1 = data(find(data(:,p)==1),1:dim);`
`X2 = data(find(data(:,p)==2),1:dim);`
`n1 = size(X1,1);`
	Membagi data kelompok 1 dan kelompok 2.

`if strcmp(option1,'c')`
`  mu1  = mean(X1); mu2  = mean(X2);`
`  cov1 = cov(X1);  cov2 = cov(X2);`
	Perhitungan penduga parameter menggunakan MLE.

`elseif strcmp(option1,'s');`
`  S1=Sm(X1,nsamp,bdp);`
`  S2=Sm(X2,nsamp,bdp);`
`  mu1 = S1.mean; mu2 = S2.mean;`
`  cov1 = S1.covariance; cov2 = S2.covariance;`
`else`
`  error('Option1 must be ''c'' or ''s''.');`
`end`
	Perhitungan penduga parameter menggunakan penduga-S.
	Memilih salah satu untuk menjalankan fungsi klasifikasi.

`if option2 == 'l'`
`  sigma  = (n1*cov1+(n-n1)*cov2)/n;`
`  res.linear  = (mu1-mu2)*inv(sigma);`
`  res.constant = 1/2*res.linear*(mu1+mu2)';`
`  res.scores  = res.linear*data(1:n,1:dim)' - res.constant ;`
	Perhitungan klasifikasi menggunakan analisis diskriminan linier.

`elseif option2 == 'q'`
`  inverscov1 = inv(cov1);`
`inverscov2=inv(cov2);`
	Perhitungan klasifikasi menggunakan

Lampiran 21. (Lanjutan)

`res.quadratic = -1/2*(inverscov1-inverscov2);`
`res.linear = mu1*inverscov1-mu2*inverscov2;`
`res.constant = 1/2*log(det(cov1)/det(cov2))+...1/2*(mu1*inverscov1*mu1'-mu2*inverscov2*mu2');`
`Z = (data(1:n,1:dim)*res.quadratic).*data(1:n,1:dim);`
`Z = sum(Z');`
`res.scores = Z+res.linear*data(1:n,1:dim)'-res.constant;`
	analisis diskriminan kuadratik.

`else`
`error('Option2 must be ''l'' (linear) or ''q''(quadratic).');`
`end`
	Memilih salah satu untuk menjalankan fungsi klasifikasi.

`res.group = (res.scores < 0) + 1;` `res.miscl = mean(res.group ~= data(:,p)');`	Keluaran hasil klasifikasi ulang menggunakan analisis diskriminan linier *robust*.

`disp('mu1');`
`disp([mu1]);`
`disp('mu2');`
`disp([mu2]);`
`disp('cov1');`
`disp([cov1]);`
`disp('cov2');`
`disp([cov2]);`
`disp('sigma');`
`disp([sigma]);`
	Menampilkan nilai-nilai yang ingin diketahui selain keluaran hasil klasifikasi ulang.