

LAPORAN PRAKTIKUM DATA WAREHOUSING DAN DATA MINING

PERTEMUAN 7

“ALGORITMA KLASIFIKASI : ARTIFICIAL NEURAL NETWORK”



Oleh:

NAMA : Daffa Putra Alwansyah

NIM : L200190031

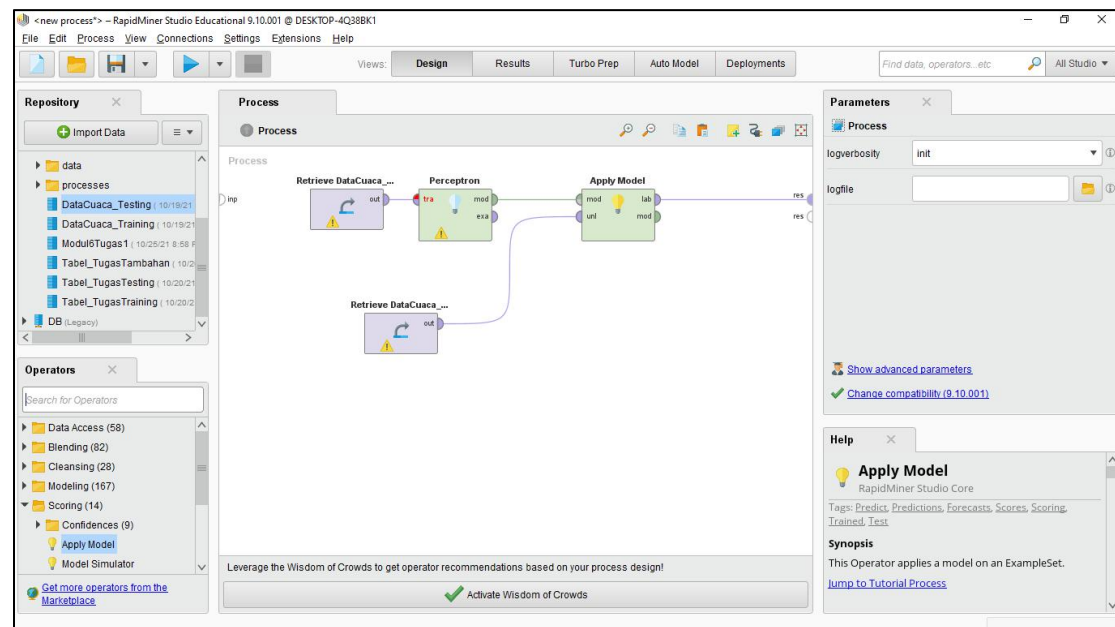
KELAS : B

PRODI : INFORMATIKA

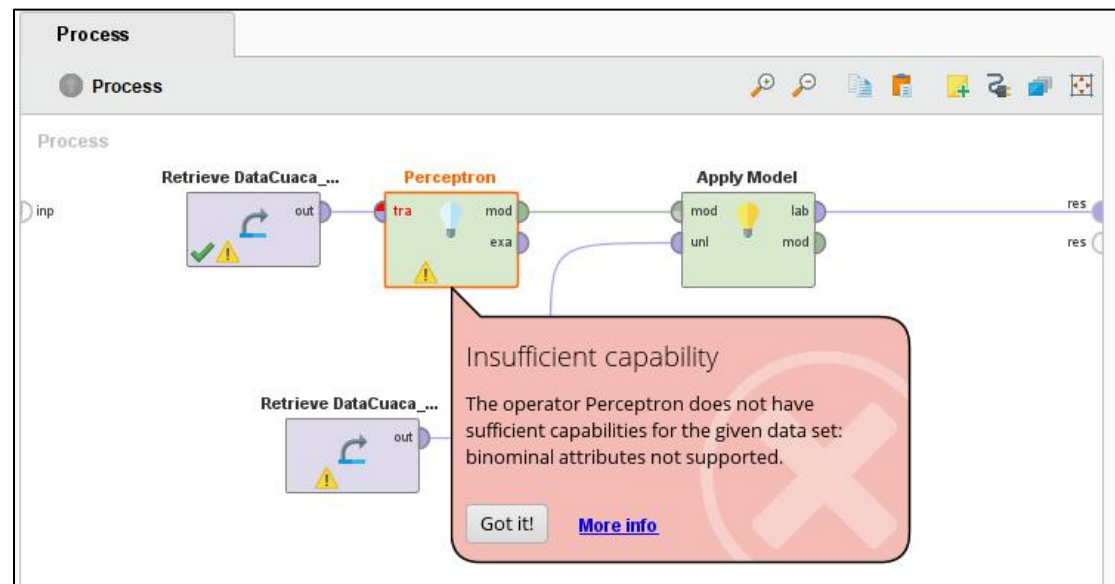
**Fakultas Komunikasi dan Informatika
Universitas Muhammadiyah Surakarta**

4.1 Prediksi nilai atribut dengan Neuron Perceptron

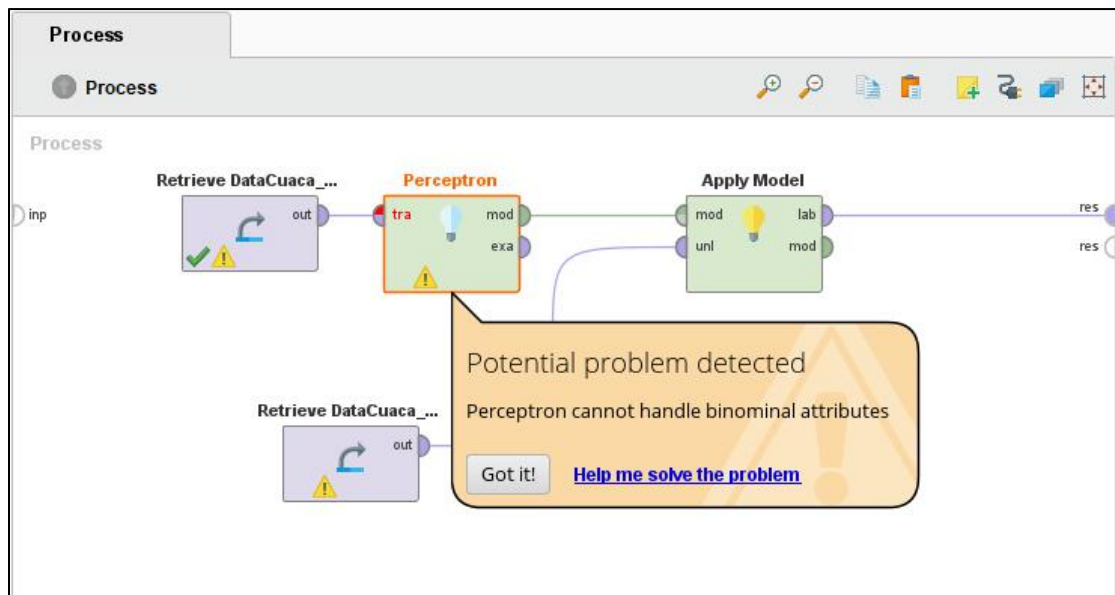
1. Buka aplikasi RapidMiner, Drag DataCuaca_Training dan Data_CuacaTesting, beserta operatornya yaitu Perceptron dan Apply Model.



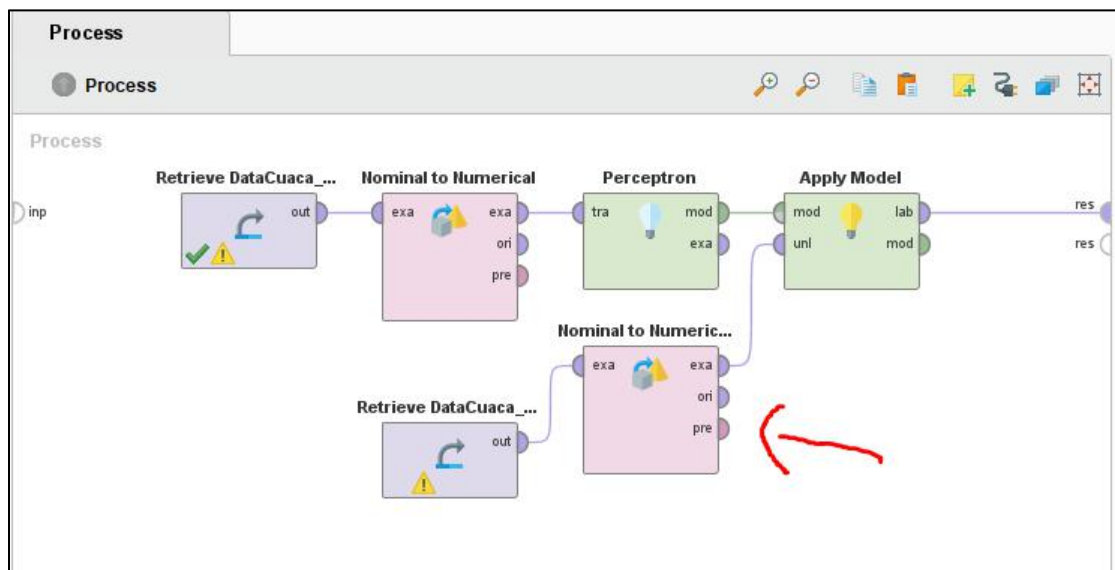
2. Setelah itu ada notifikasi Error, yang menyatakan nilai kelas data dalam data Training yang bertipe polynomial tidak dapat diproses, tutup pesan tersebut.



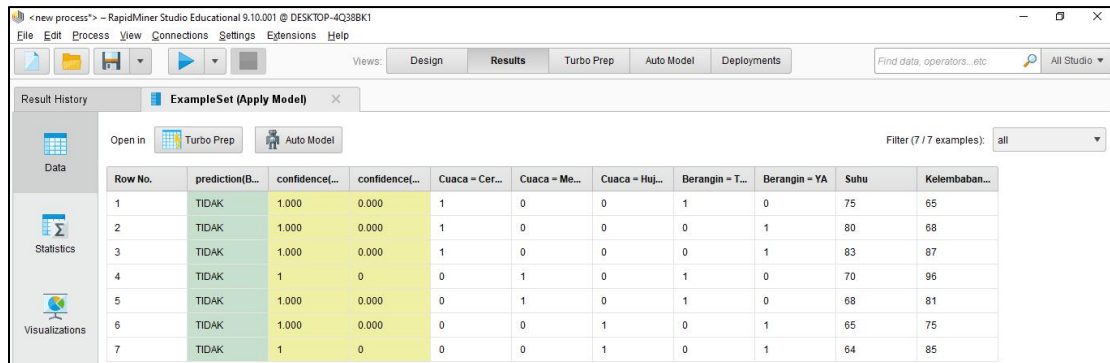
3. Klik pada Perceptron, lalu “*Help me solve the problem*”



4. Setelah itu muncul “Nominal to Numerical” pada Training, tambahkan juga pada Testing.



5. Akan ditunjukkan hasil prediksi data testing menggunakan JST(Perceptron).

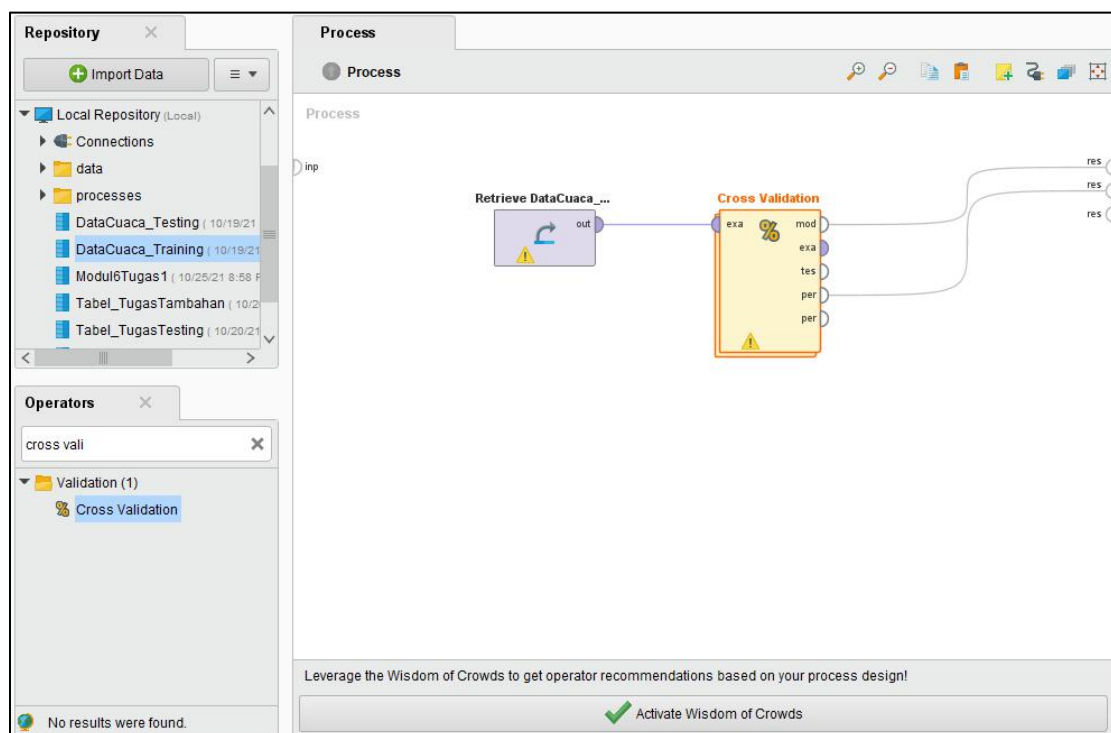


Row No.	prediction(B...	confidence[...	confidence[...	Cuaca = Cer...	Cuaca = Me...	Cuaca = Huj...	Berangin = T...	Berangin = YA	Suhu	Kelembaban...
1	TIDAK	1.000	0.000	1	0	0	1	0	75	65
2	TIDAK	1.000	0.000	1	0	0	0	1	80	68
3	TIDAK	1.000	0.000	1	0	0	0	1	83	87
4	TIDAK	1	0	0	1	0	1	0	70	96
5	TIDAK	1.000	0.000	0	1	0	1	0	68	81
6	TIDAK	1.000	0.000	0	0	1	0	1	65	75
7	TIDAK	1	0	0	0	1	0	1	64	85

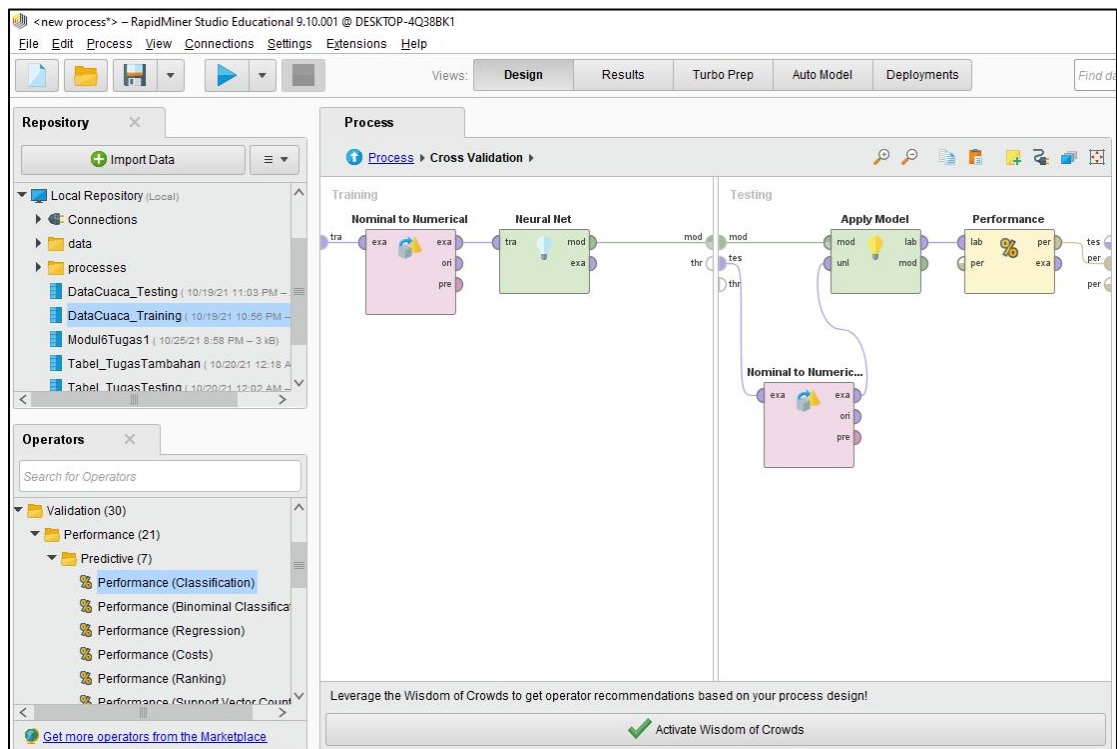
(Hasil prediksi menunjukkan bahwa semua data akan memiliki kelas bermain_tenis = TIDAK, berdasarkan perbandingan tingkat confidence masing-masing nilai kelas YA dan TIDAK.)

4.2 Mengetahui Nilai performance Vector pada jaringan saraf tiruan

1. Menggunakan DataCuaca_Training dan operator Cross Validation, hubungkan port sesuai di modul, setelah itu klik 2x pada Cross Validation.



2. Cari Operator untuk disebelah kiri yaitu Nominal to Numerical & Neural Net, sedangkan disebelah kanan yaitu Apply Model, Performance dan Nominal to Numerical, setelah itu process dan klik tombol RUN.



3. Hasilnya:

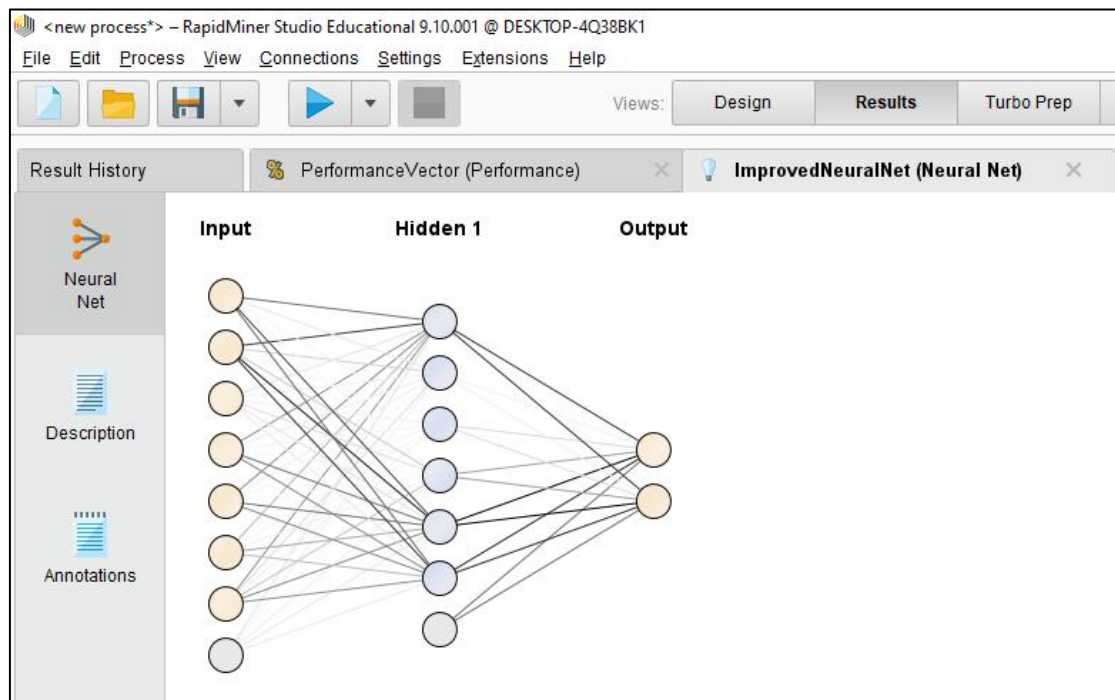
- Tab PerformanceVector(Performance), untuk menunjukkan tingkat akurasi, presisi, recall dan lain-lain dalam bentuk tabel dan plot view.

The screenshot shows the PerformanceVector(Performance) tab in RapidMiner Studio. The tab displays a table of performance metrics for the 'ImprovedNeuralNet (Neural Net)' model. The table includes accuracy, precision, recall, and class precision.

	true TIDAK	true YA	class precision
pred. TIDAK	2	3	40.00%
pred. YA	3	6	66.67%
class recall	40.00%	66.67%	

- Tab ImprovedNeuralNet (Neural Net), memperlihatkan arsitektur jaringan syaraf yang digunakan untuk proses. Ada 2 mode view yaitu:

A. Neural Net, digunakan untuk melihat bentuk arsitektur JST



Berdasarkan arsitektur yang terbentuk, dapat dilihat bahwa jumlah node (simpul) masing-masing layer sebagai berikut :

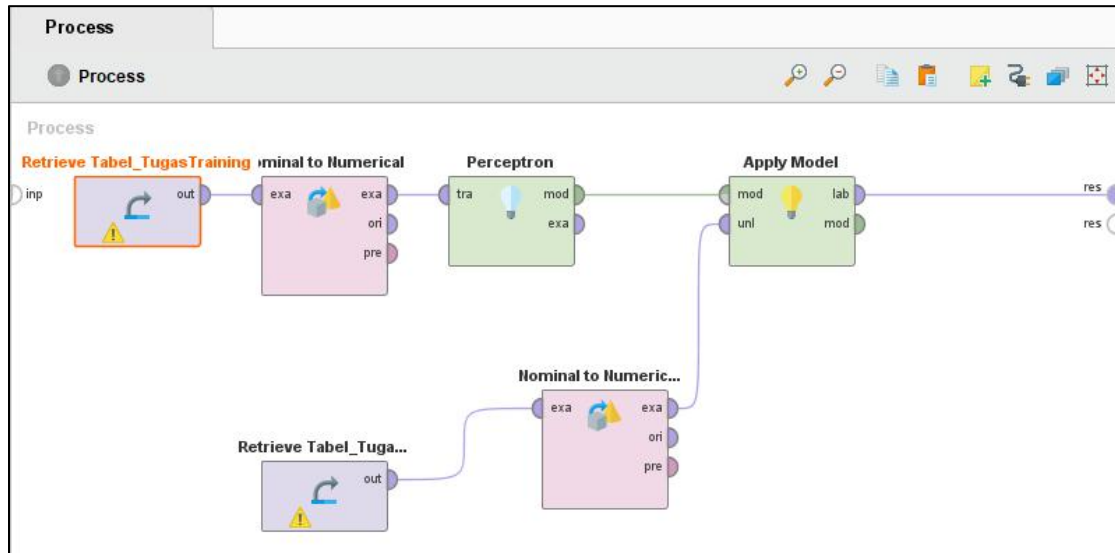
1. Input Layer = 7 node input, dan 1 node berbobot 1
2. Hidden Layer = 6 node hidden, dan 1 node berbobot 1
3. Output Layer = 2 node (YA, TIDAK)

B. Description, digunakan untuk melihat nilai bobot sigmoid masing-masing node (simpul) pada hidden layer dan output layer.

ImprovedNeuralNet Hidden 1 ===== Node 1 (Sigmoid) ----- Cuaca = Cerah: -0.990 Cuaca = Mendung: 1.338 Cuaca = Hujan: -0.192 Berangin = TIDAK: 0.742 Berangin = YA: -0.794 Suhu: -0.484 Kelembaban_udara: -0.737 Bias: -0.178 Node 2 (Sigmoid) ----- Cuaca = Cerah: -0.157 Cuaca = Mendung: 0.313 Cuaca = Hujan: -0.092 Berangin = TIDAK: 0.115 Berangin = YA: -0.107 Suhu: -0.021 Kelembaban_udara: -0.235 Bias: -0.093	Node 3 (Sigmoid) ----- Cuaca = Cerah: -0.058 Cuaca = Mendung: 0.043 Cuaca = Hujan: 0.095 Berangin = TIDAK: 0.066 Berangin = YA: -0.070 Suhu: 0.060 Kelembaban_udara: 0.090 Bias: 0.019 Node 4 (Sigmoid) ----- Cuaca = Cerah: 0.047 Cuaca = Mendung: -0.372 Cuaca = Hujan: 0.229 Berangin = TIDAK: -0.032 Berangin = YA: 0.027 Suhu: 0.104 Kelembaban_udara: 0.402 Bias: 0.110 Node 5 (Sigmoid) ----- Cuaca = Cerah: -1.234 Cuaca = Mendung: 1.590 Cuaca = Hujan: -0.192 Berangin = TIDAK: 0.956 Berangin = YA: -0.978 Suhu: -0.598 Kelembaban_udara: -0.805 Bias: -0.151	Node 6 (Sigmoid) ----- Cuaca = Cerah: -1.038 Cuaca = Mendung: 1.407 Cuaca = Hujan: -0.159 Berangin = TIDAK: 0.826 Berangin = YA: -0.818 Suhu: -0.476 Kelembaban_udara: -0.776 Bias: -0.153 Output ===== Class 'TIDAK' (Sigmoid) ----- Node 1: -1.370 Node 2: -0.090 Node 3: 0.253 Node 4: 0.652 Node 5: -1.742 Node 6: -1.471 Threshold: 0.979 Class 'YA' (Sigmoid) ----- Node 1: 1.385 Node 2: 0.106 Node 3: -0.235 Node 4: -0.662 Node 5: 1.759 Node 6: 1.456 Threshold: -0.996
---	---	--

TUGAS!

- Menggunakan file excel Tugas nomor 1 Modul 6 (Training) & Tugas nomor 1 Modul 8 (Testing), lalu masukan operator Nominal to Numerical, Perceptron dan Apply Model.



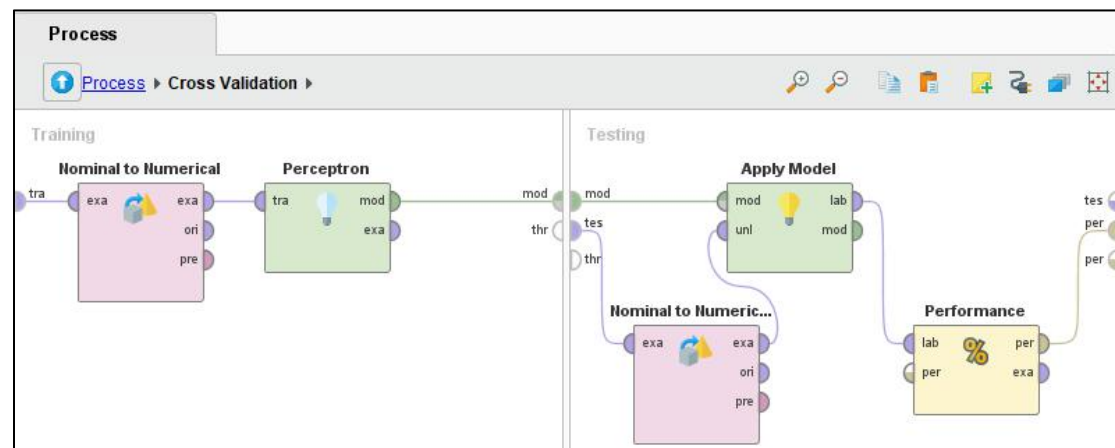
- Hasil prediksi terhadap data testing lama studi mahasiswa dengan menggunakan model Perceptron.

Result History: ExampleSet (Apply Model) x ExampleSet (//Local Repository/Tabel_TugasTraining)

Filter (10 / 10 examples): all

Row No.	prediction(L...	confidence(L...	confidence_...	Jurusan_S...	Jurusan_S...	Jurusan_S...	Gender = W...	Gender = PR...	Asal_Sekola...	Asal_Sekola...	Asisten = TL...	Asis
1	TEPAT	0.462	0.538	1	0	0	1	0	1	0	1	0
2	TEPAT	0.385	0.615	0	1	0	0	1	1	0	0	1
3	TERLAMBAT	0.536	0.464	1	0	0	0	1	1	0	1	0
4	TERLAMBAT	0.579	0.421	0	0	1	0	1	0	1	1	0
5	TEPAT	0.465	0.535	1	0	0	1	0	1	0	1	0
6	TEPAT	0.325	0.675	0	1	0	1	0	0	1	0	1
7	TEPAT	0.458	0.542	0	1	0	0	1	1	0	1	0
8	TEPAT	0.455	0.545	0	1	0	0	1	1	0	1	0
9	TERLAMBAT	0.576	0.424	0	0	1	0	1	0	1	1	0
10	TEPAT	0.462	0.538	1	0	0	1	0	1	0	1	0

- Menggunakan PerformanceVector untuk mencari tingkat akurasi, presisi dan recall!



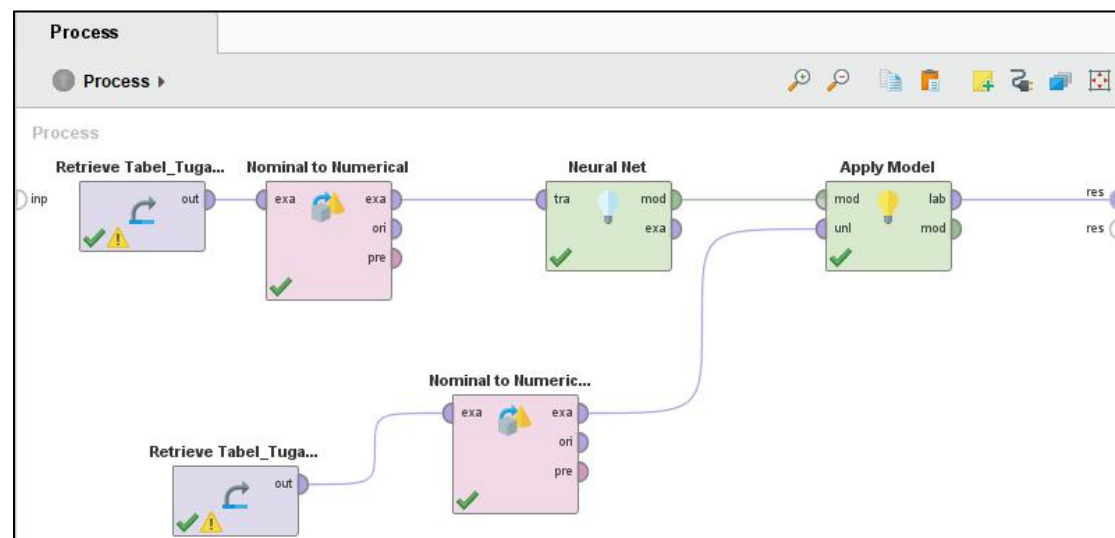
- Hasil dengan menggunakan model Perceptron

Table View Plot View

accuracy: 40.00% +/- 31.62% (micro average: 40.00%)

	true TERLAMBAT	true TEPAT	class precision
pred. TERLAMBAT	4	9	30.77%
pred. TEPAT	3	4	57.14%
class recall	57.14%	30.77%	

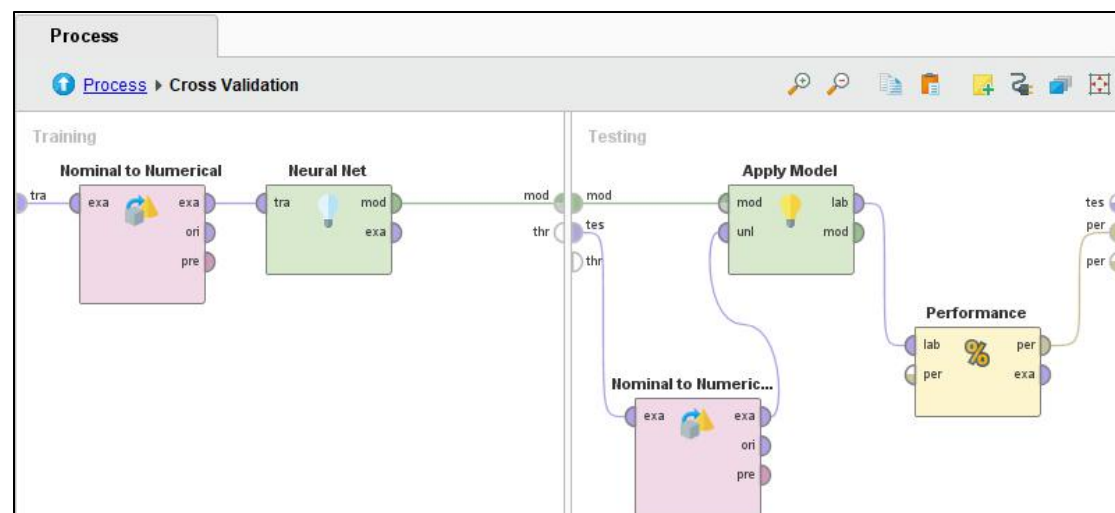
- Mengganti operator Perceptron menjadi Neural Net.



- Hasil prediksi terhadap data testing lama studi mahasiswa dengan menggunakan model Neural Net.

Row No.	prediction(L...	confidence(...	confidence(...	Jurusan_S...	Jurusan_S...	Jurusan_S...	Gender = W...	Gender = PR...	Asal_Sekola...	Asal_Sekola...	Asisten = TL...	Asis
1	TEPAT	0.331	0.669	1	0	0	1	0	1	0	1	0
2	TEPAT	0.027	0.973	0	1	0	0	1	1	0	0	1
3	TERLAMBAT	0.588	0.412	1	0	0	0	1	1	0	1	0
4	TERLAMBAT	0.679	0.321	0	0	1	0	1	0	1	1	0
5	TEPAT	0.399	0.601	1	0	0	1	0	1	0	1	0
6	TEPAT	0.032	0.968	0	1	0	1	0	0	1	0	1
7	TEPAT	0.399	0.601	0	1	0	0	1	1	0	1	0
8	TEPAT	0.325	0.675	0	1	0	0	1	1	0	1	0
9	TERLAMBAT	0.655	0.345	0	0	1	0	1	0	1	1	0
10	TEPAT	0.331	0.669	1	0	0	1	0	1	0	1	0

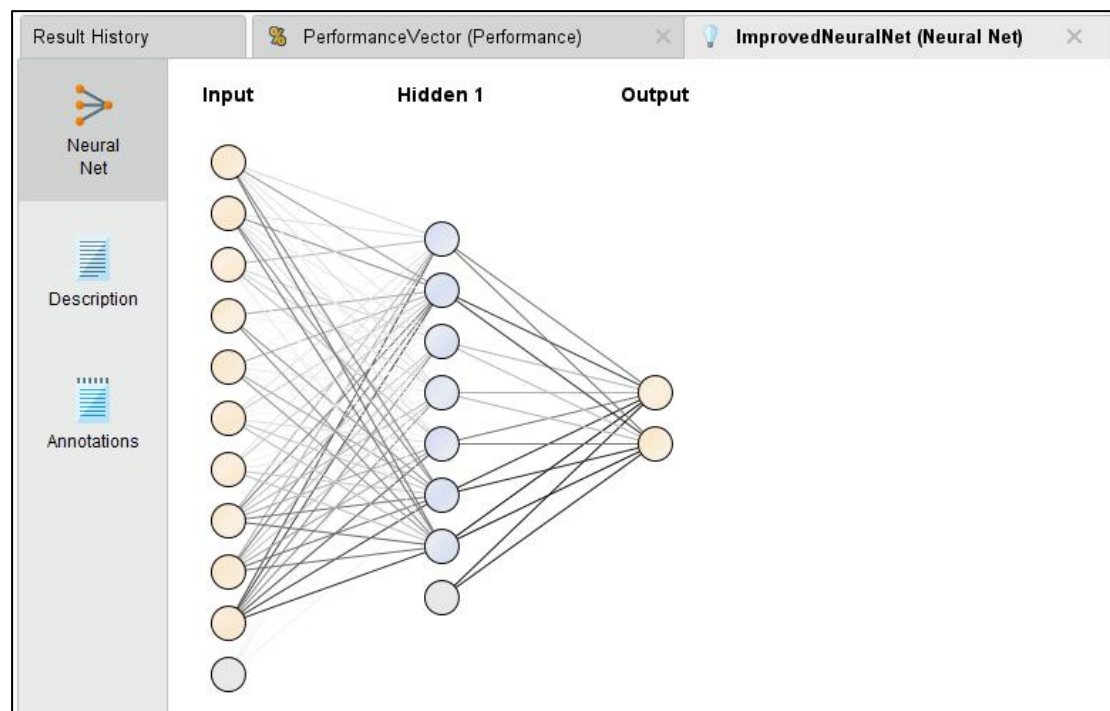
- Menggunakan PerformanceVector untuk mencari tingkat akurasi, presisi dan recall!



- Hasil dengan menggunakan model Neural Net.

<div> <input checked="" type="radio"/> Table View <input type="radio"/> Plot View </div>			
accuracy: 60.00% +/- 31.62% (micro average: 60.00%)			
	true TERLAMBAT	true TEPAT	class precision
pred. TERLAMBAT	3	4	42.86%
pred. TEPAT	4	9	69.23%
class recall	42.86%	69.23%	

➤ Arsitektur Jaringan saraf yang terbentuk



➤ Berdasarkan arsitektur yang terbentuk, dapat dilihat bahwa jumlah node (simpul) masing-masing layer sebagai berikut :

1. Input layer = 10 node input, dan 1 node berbobot 1
2. Hidden layer = 7 node hidden, dan 1 node berbobot 1
3. Output layer = 2 node (TEPAT dan TERLAMBAT)

- Nilai - nilai bobot sigmoid masing-masing node (simpul) pada hidden layer dan output layer :

ImprovedNeuralNet	
<p>Hidden 1 =====</p> <p>Node 1 (Sigmoid) -----</p> <p>Jurusan_SMA = IPS: 0.200 Jurusan_SMA = IPA: 0.160 Jurusan_SMA = LAIN: -0.361 Gender = WANITA: 0.000 Gender = PRIA: 0.055 Asal_Sekolah = SURAKARTA: 0.146 Asal_Sekolah = LUAR: -0.124 Asisten = TIDAK: -0.399 Asisten = YA: 0.420 Rerata_SKS: 0.927 Bias: 0.035</p> <p>Node 2 (Sigmoid) -----</p> <p>Jurusan_SMA = IPS: -0.506 Jurusan_SMA = IPA: 0.561 Jurusan_SMA = LAIN: -0.020 Gender = WANITA: 0.443 Gender = PRIA: -0.404 Asal_Sekolah = SURAKARTA: 0.258 Asal_Sekolah = LUAR: -0.277 Asisten = TIDAK: -0.706 Asisten = YA: 0.698 Rerata_SKS: 0.961 Bias: -0.048</p>	<p>Node 3 (Sigmoid) -----</p> <p>Jurusan_SMA = IPS: 0.121 Jurusan_SMA = IPA: 0.120 Jurusan_SMA = LAIN: -0.215 Gender = WANITA: 0.076 Gender = PRIA: -0.050 Asal_Sekolah = SURAKARTA: 0.054 Asal_Sekolah = LUAR: -0.024 Asisten = TIDAK: -0.296 Asisten = YA: 0.323 Rerata_SKS: 0.608 Bias: -0.072</p> <p>Node 4 (Sigmoid) -----</p> <p>Jurusan_SMA = IPS: 0.168 Jurusan_SMA = IPA: 0.151 Jurusan_SMA = LAIN: -0.220 Gender = WANITA: 0.024 Gender = PRIA: -0.101 Asal_Sekolah = SURAKARTA: 0.051 Asal_Sekolah = LUAR: -0.059 Asisten = TIDAK: -0.395 Asisten = YA: 0.316 Rerata_SKS: 0.578 Bias: -0.017</p>

<pre> Node 5 (Sigmoid) ----- Jurusan_SMA = IPS: 0.066 Jurusan_SMA = IPA: 0.172 Jurusan_SMA = LAIN: -0.330 Gender = WANITA: 0.043 Gender = PRIA: -0.056 Asal_Sekolah = SURAKARTA: 0.173 Asal_Sekolah = LUAR: -0.114 Asisten = TIDAK: -0.437 Asisten = YA: 0.476 Rerata_SKS: 0.902 Bias: 0.024 Node 6 (Sigmoid) ----- Jurusan_SMA = IPS: -0.676 Jurusan_SMA = IPA: 0.604 Jurusan_SMA = LAIN: 0.009 Gender = WANITA: 0.559 Gender = PRIA: -0.466 Asal_Sekolah = SURAKARTA: 0.300 Asal_Sekolah = LUAR: -0.288 Asisten = TIDAK: -0.716 Asisten = YA: 0.747 Rerata_SKS: 1.017 Bias: -0.061 </pre>	<pre> Node 7 (Sigmoid) ----- Jurusan_SMA = IPS: -0.758 Jurusan_SMA = IPA: 0.736 Jurusan_SMA = LAIN: 0.101 Gender = WANITA: 0.620 Gender = PRIA: -0.547 Asal_Sekolah = SURAKARTA: 0.349 Asal_Sekolah = LUAR: -0.333 Asisten = TIDAK: -0.777 Asisten = YA: 0.752 Rerata_SKS: 1.079 Bias: -0.015 </pre>
	<pre> Output ===== Class 'TERLAMBAT' (Sigmoid) ----- Node 1: -0.711 Node 2: -1.001 Node 3: -0.386 Node 4: -0.409 Node 5: -0.718 Node 6: -1.133 Node 7: -1.282 Threshold: 1.195 Class 'TEPAT' (Sigmoid) ----- Node 1: 0.732 Node 2: 1.040 Node 3: 0.364 Node 4: 0.391 Node 5: 0.706 Node 6: 1.128 Node 7: 1.248 Threshold: -1.185 </pre>

- Kesimpulan saya yaitu, adanya perubahan pada akurasi, presisi dan recall, dimana pada Perceptron (40%) lebih kecil dari pada Neural Net (60%), jadi lebih akurat pada Neural Net.