

BAB IV HASIL DAN PEMBAHASAN

4.1 Pengolahan Data

4.1.1 Seleksi Data (*Data Selection*)

Pada penelitian ini data yang digunakan merupakan data sekunder, karena data yang digunakan pada penelitian ini penulis peroleh dari dua sumber yaitu, data pertama dari Biro Sistem Informasi (BSI) untuk data yang diperoleh yaitu *Database E-learning* fakultas Teknik tahun 2014-2018, dan data yang kedua dari Tata Usaha program studi Teknik informatika data yang diperoleh yaitu data presensi dan data nilai akhir mahasiswa dari 6 mata kuliah yang digunakan. Untuk melakukan pengolahan data yang telah diberikan oleh Biro Sistem Informasi penulis menggunakan *software My SQL Workbench* dengan cara mengimpor *table* yang telah diberikan kedalam *My SQL Workbench* tujuannya yaitu untuk mengetahui nilai dan atribut apa saja yang terdapat pada *tabel-tabel* yang diberikan, selain itu juga untuk mengetahui relasi pada setiap *table*. Langkah pertama yang dilakukan yaitu masuk kedalam *server database My SQL Workbench* terlebih dahulu seperti pada gambar 4.1.



Gambar 4. 1 Connect Database

Setelah dapat terhubung ke server penulis lanjutkan dengan proses *import table* ke dalam *My SQL Workbench* total *table* yang diimpor yaitu sebanyak 396 *table*. Setelah semua *table* di ekspor seperti pada gambar 4.2 penulis lanjutkan

id	username	idnumber	firstname	lastname	department
1	guest		Guest user		
2	admin		admin	learning	
3	wahyui MT.				
4	hendra.bayu.2006@ft.umy.ac.id	1487988827	Hendra	Bayu Dwi Yuwono	Mechanical Engineering
5	nova.setiyawan.2009@ft.umy.ac.id	1487988827	Nova	Setiyawan	Mechanical Engineering
6	rifky.riza.2009@ft.umy.ac.id	1487988828	Rifky	Riza Rifal	Mechanical Engineering
7	pradana.anantyo.2009@ft.umy.ac.id	1487988828	Pradana	Anantyo Nugroho	Mechanical Engineering
8	achmad.susanto.2009@ft.umy.ac.id	1487988828	Achmad	Susanto	Mechanical Engineering
9	ulhi.amri.2009@ft.umy.ac.id	1487988828	Ulhi	Amri	Mechanical Engineering
10	muhammad.istighfar.2009@ft.umy.ac.id	1487988830	Muhammad	Istighfar Nur Islah	Mechanical Engineering
11	budi.prayitno.2009@ft.umy.ac.id	1487988830	Budi	Prayitno	Mechanical Engineering
12	sigit.dwi.2009@ft.umy.ac.id	1487988830	Sigit	Dwi Prasetyo	Mechanical Engineering
13	danang.widhi.2009@ft.umy.ac.id	1487988832	Danang	Widhi Atmaja	Mechanical Engineering
14	m.yahya.2009@ft.umy.ac.id	1487988832	M	Yahya	Mechanical Engineering
15	rizki.prayogi.2010@ft.umy.ac.id	1487988833	Rizki	Prayogi	Mechanical Engineering
16	zefr.yanto.2013@ft.umy.ac.id	1487988835	Ze	Ryanto	Mechanical Engineering
17	kuswinarso.2013@ft.umy.ac.id	1487988837	Kuswinarso	UMY	Mechanical Engineering
18	bagas.yoso.2013@ft.umy.ac.id	1487988839	Bagas	Yoso Kuncoro	Mechanical Engineering
19	fany.joko.2013@ft.umy.ac.id	1487988842	Fany	Joko Sutrisno	Mechanical Engineering
20	agung.nugroho.2013@ft.umy.ac.id	1487988845	Agung	Nugroho	Mechanical Engineering
21	rudy.cahyo.2013@ft.umy.ac.id	1487988848	Rudy	Cahyo Nugroho	Mechanical Engineering
22	dwi.susanto.2013@ft.umy.ac.id	1487988851	Dwi	Susanto	Mechanical Engineering
23	m.ilham.2013@ft.umy.ac.id	1487988853	M	Ilham Saputra	Mechanical Engineering

Gambar 4. 3 Eksport Table User

2. *Table Course*: *Table Course* berisikan data-data mata kuliah dari seluruh jurusan yang ada di fakultas teknik UMY yang menggunakan media pembelajaran *e-learning*. Data mata kuliah terbagi ke dalam 31 atribut dan terdapat 737 mata kuliah. dikarenakan tidak semua atribut mata kuliah yang akan digunakan maka hanya 3 atribut saja yang digunakan yaitu: *id*, *fullname*, dan *shortname* seperti pada gambar 4.4.

	A	B	C	D	E	F	G	H	I
1	id;	fullname;	shortname"						
2	1;	"Engineering eLearning";	"Engineering"						
3	2;	"Pemrograman Komputer";	"PKomp"						
4	3;	"Praktikum CAD";	"CAD"						
5	4;	"Praktikum Komputasi";	"komputasi"						
6	5;	"Material Teknik";	"Material Teknik "						
7	6;	"Kalkulus";	"kalkulus"						
8	7;	"Bahasa Indonesia";	"BIndonesia"						
9	10;	"Getaran Mekanik ";	"GetMek"						
10	11;	"Struktur dan Sifat Material";	"SSM"						
11	12;	"Mekanika Kekuatan Material";	"MKM"						
12	14;	"Metalurgi";	"Metalurgi"						
13	18;	"Matematika Teknik 1";	"MatematikaTeknik1"						
14	19;	"Kiner B";	"Kinematika"						
15	20;	"Kinematika";	"Kinematik"						
16	21;	"Teknik Kendali";	"kendali"						
17	25;	"Hidrolik & Pneumatik";	"Hidrolik"						
18	26;	"Pesawat Angkat dan Angkut";	"angkat"						
19	27;	"Rangkaian Listrik - Electric Circuits";	"TEU 2212"						
20	28;	"Mikrokontroler/Microcontroller";	"TEU 5515"						
21	29;	"Pengolahan Sinyal Digital/Digital Signal Processing";	"TEU 5513"						
22	30;	"Sinyal dan Sistem/Signal and Systems";	"TEU 5512"						
23	31;	"Contoh elearning";	"contoh"						
24	32;	"Agama I - Religious I";	"TEI 1111"						
25	33;	"Bahasa Inggris I - English I";	"TEI 1111"						

Gambar 4. 4 Ekspor Table Course

3. *Table Quiz*: *Table Quiz* berisikan data kuis-kuis yang telah dibuat oleh dosen berdasarkan mata kuliah yang diajarkan. Pada *table quiz* terdapat 2044 kuis yang telah dibuat berdasarkan mata kuliah. *Table quiz* memiliki relasi dengan *table course*, *foreign keynya* yaitu atribut *course*. Pada *table quiz* terdapat 41 atribut tetapi hanya 3 atribut saja yang digunakan yaitu: *id*, *course*, dan *name*. Seperti pada gambar 4.5

	A	B	C	D	E	F	G	H	I
1	Id;"course";"name"								
2	1;19;"Tugas Sintesa 1"								
3	2;2;"Tugas 9 (UK)"								
4	7;19;"Tugas Sintesa 2"								
5	8;21;"Uji Kompetensi II Teknik Kendali (09.30 - 11.30)"								
6	9;21;"Uji Kompetensi II Teknik Kendali (12.30-14.30)"								
7	10;21;"Uj Kompetensi II Teknik Kendali (15.30 - 17.30)"								
8	11;26;"Uji Kompetensi II Pesawat Angkat dan Angkut"								
9	12;25;"Uji Kompetensi II Hidrolik & Pneumatik"								
10	13;25;"Uji Kompetensi II Hidrolik & Pneumatik"								
11	14;25;"Uji Kompetensi II Hidrolik & Pneumatik"								
12	15;25;"Uji Kompetensi II Hidrolik & Pneumatik"								
13	16;19;"Tugas Sintesa 2 A"								
14	19;19;"Tugas nok 1"								
15	20;19;"Tugas Nok2"								
16	23;21;"Uji Kompetensi III kelas A"								
17	24;26;"Uji Kompetensi III"								
18	25;21;"Uji Kompetensi III Teknik Kendali Kelas D"								
19	26;25;"Uji Kompetensi III Hidrolik & Pneumatik Kelas D"								
20	27;25;"Uji Kompetensi Hidrolik & Pneumatik III kelas C"								
21	28;25;"Uji Kompetensi III Hidrolik & Pneumatik Kelas B"								
22	29;25;"Uji Kompetensi III Hidrolik & Pneumatik kelas A"								
23	30;21;"Uji Kompetensi III Teknik Kendali kelas B"								
24	31;21;"Uji Kompetensi III Teknik Kendali kelas C"								
25	34;21;"Uji Kompetensi III Teknik Kendali"								

Gambar 4. 5 Eksport Table Quiz

4. *Table Quiz Grades: Table Quiz Grades* berisikan data nilai-nilai kuis mahasiswa, *table quiz grades* memiliki relasi dengan dua *table*: pertama *table quiz* atribut *quiz* sebagai *foreign key*, dan *table* kedua *table user* atribut *user id* sebagai *foreign key*. Pada *table* nilai kuis terdapat 116.506 data nilai kuis mahasiswa dan terdapat 5 atribut didalamnya, dan hanya satu atribut saja yang dihilangkan. Seperti pada gambar 4.6

	A	B	C	D	E	F	G
1	id;"quiz";"userid";"grade"						
2	1;1;401;2.0						
3	2;1;568;4.0						
4	3;1;585;4.0						
5	4;1;526;4.0						
6	5;1;561;4.0						
7	6;1;388;2.0						
8	7;1;494;4.0						
9	8;1;423;2.0						
10	9;1;452;4.0						
11	10;1;591;2.0						
12	11;1;467;4.0						
13	12;1;441;2.0						
14	13;1;479;2.0						
15	14;1;414;2.0						
16	15;1;453;2.0						
17	16;1;437;4.0						
18	17;1;390;4.0						
19	18;1;416;4.0						
20	19;1;558;4.0						
21	20;1;586;2.0						
22	21;1;421;2.0						

Gambar 4. 6 Eksport Table Quiz Grades

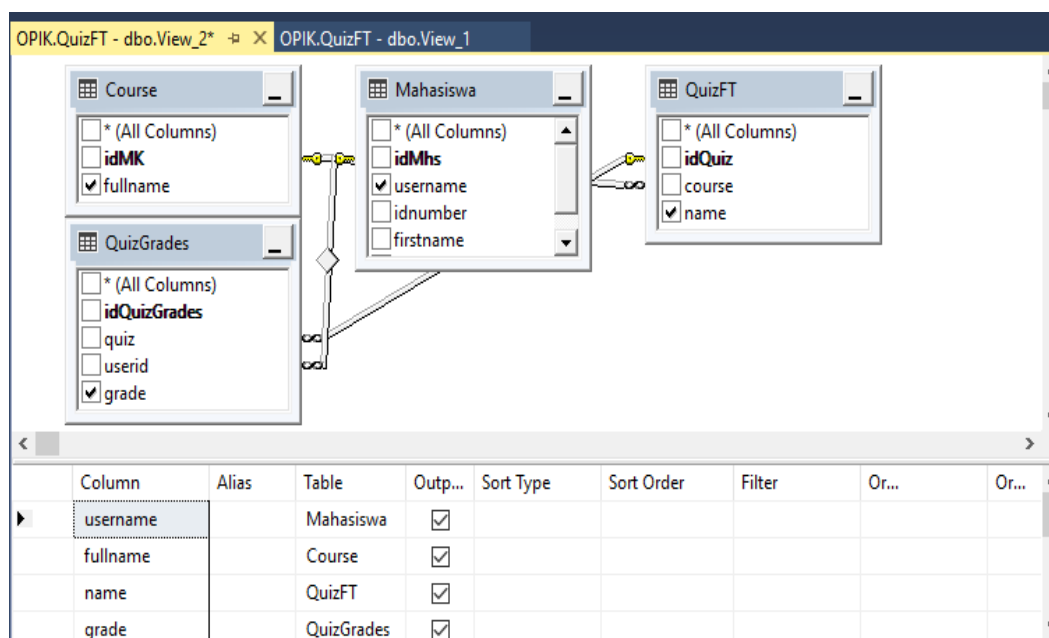
4.1.2 Pembersihan Data (*Data Cleaning*)

Setelah data melewati proses pengolahan data yang pertama selanjutnya *table* yang telah diekspor menjadi *file *.csv* selanjutnya diubah format *filenya* menjadi **.xls* untuk memudahkan dalam proses *cleaning* data. Pada tahap ini dilakukan *cleaning* terhadap *value* dari atribut-atribut yang ada pada masing-masing *table*. Dikarenakan masih adanya *value* atribut yang tidak bisa digunakan karena terdapat banyak data yang *null*, *noise*, dan tidak relevan sehingga diperlukannya pembersihan data untuk memastikan *value* dari masing-masing atribut telah sesuai dengan yang dibutuhkan.

Tahap ini memerlukan proses yang panjang dikarenakan penulis ingin mendapatkan hasil akurasi yang terbaik sehingga dalam proses pembersihan data ini sangat menjadi *concern* tersendiri bagi penulis dengan memastikan data yang nantinya akan digunakan sudah benar-benar bersih dari data yg *null*, data yang tidak konsisten, dan dari data yang tidak relevan. Pada tahap ini dimulai dengan pembersihan nilai atribut pada *table user* hingga *table quiz grades*.

4.1.3 Integrasi Data (*Data Integration*)

Data Integration merupakan proses yang bertujuan untuk menggabungkan data. Agar mempermudah dalam proses pengambilan data-data yang dibutuhkan sehingga ditahap ini penulis menggunakan 2 *software* yaitu: *Visual Studio 2017* dan *Microsoft SQL Server 2017*. Pada *software Visual Studio* penulis menggunakan *Integration Service Project* untuk melakukan integrasi data *excel* ke dalam sebuah *database*. *Microsoft SQL Server* berguna sebagai wadah yang menampung data *excel* yang dimasukkan. Setelah *database* selesai dibuat selanjutnya penulis membuat *view* seperti gambar 4.7 tujuannya agar mempermudah dalam proses pengambilan data dari atribut-atribut yang dibutuhkan saja untuk dijadikan sebuah *dataset*.



Gambar 4. 7 Pembuatan View

Setelah data yang dibutuhkan sudah didapat, selanjutnya data tersebut kembali dimasukkan ke dalam *Microsoft Excel* untuk dilakukan *filtering* terhadap data kuis yang akan digunakan. ada pun data yang nantinya yang akan digunakan adalah data kuis *e-learning* mahasiswa jurusan Teknik informatika angkatan 2014 dan 2015, dan untuk mata kuliah yang digunakan yaitu:

1. Mata Kuliah Object Oriented Analysis *Design*
2. Mata Kuliah Software *Testing* and Quality Assurance
3. Mata Kuliah Web Application Development
4. Mata Kuliah Web Component Development
5. Mata Kuliah Web Component Development (JSF, Hibernate, and Spring Framework)
6. Mata Kuliah Business Intelligence System

Setelah data 6 mata kuliah diatas didapatkan selanjutnya menghitung total kuis yang dimiliki masing-masing mata kuliah dan menghitung nilai rata-rata nilai kuis mahasiswa yang mengambil mata kuliah tersebut. Kemudian data yang telah diterima dari staff Tata Usaha Jurusan Teknik informatika digabungkan menjadi satu didalam *file* excel yang sama. Seperti dapat dilihat pada *table* 4.1. Pada atribut *username* penulis sengaja mengganti 2 angka terakhir demi menjaga kerahasiaan data mahasiswa.

Tabel 4. 1 Integration Data

username	fullname	Quiz I	Quiz II	Total Quiz	Rata-Rata	Presensi	Nilai Akhir
201401400AA	Object Ori	70	40	2	55	100.00 %	AB
201401400BB	Object Ori	30	30	2	30	85.71 %	B
201401400CC	Object Ori	60	80	2	70	100.00 %	AB
201401400DD	Object Ori	40	40	2	40	100.00 %	B
201401400EE	Object Ori	50	50	2	50	100.00 %	B
201401400FF	Object Ori	30	60	2	45	85.71 %	BC
201401400GG	Object Ori	70	60	2	65	100.00 %	AB
201401400HH	Object Ori	30	50	2	40	100.00 %	AB
201401400II	Object Ori	60	60	2	60	100.00 %	AB
201401400JJ	Object Ori	50	60	2	55	100.00 %	B
201401400KK	Object Ori	70	80	2	75	100.00 %	A

4.1.4 Transformasi Data (*Data Transformation*)

Data Transformation merupakan proses yang dilakukan dengan tujuan mengubah nilai dari beberapa atribut yang akan digunakan menjadi bentuk yang sesuai untuk di proses dengan *data mining* menggunakan *software RapidMiner*. Beberapa atribut yang di ubah yaitu: presensi, jumlah kuis, dan nilai akhir.

Selanjutnya penulis melakukan beberapa inisialisasi terhadap beberapa atribut yang telah disebutkan menjadi seperti pada *table* 4.2, 4.3, dan 4.4:

Tabel 4. 2 Inisialisasi Jumlah Presensi

Presensi	Inisialisasi
Kehadiran \geq 75%	Tertib
Kehadiran $<$ 75%	Tidak tertib

Tabel 4. 3 Inisiasi Jumlah Kuis

Jumlah Kuis	Inisialisasi
Jumlah Kuis yang dikerjakan \geq Setengah dari Jumlah Kuis	Rajin
Jumlah Kuis yang dikerjakan $<$ Setengah dari Jumlah Kuis	Tidak Rajin

Tabel 4. 4 Inisiasi Table Nilai Akhir

Nilai Akhir	Inisialisasi
Nilai Akhir A dan AB	Sangat Paham
Nilai Akhir B	Paham
Nilai Akhir BC dan C	Cukup Paham
Nilai Akhir D dan E	Tidak Paham

Atribut yang telah di inisialisasi selanjutnya di transformasikan ke dalam *file excel*, untuk atribut jumlah kuis penulis melakukan perubahan nama menjadi Kerajinan, karena *table* ini digunakan untuk mengukur tingkat kerajinan mahasiswa dalam mengerjakan kuis-kuis yang diberikan. Tentunya atribut kerajinan ini sangat berguna untuk tolah ukur dalam prediksi calon asiten dosen. Dikarenakan atribut yang sudah ada dinilai masih kurang cukup maka perlu ditambahkan satu atribut lagi yaitu atribut keterangan untuk mengkategorikan layak dan tidak layak berdasarkan atribut-atribut yang ada. Atribut keterangan ini

nantinya akan digunakan sebagai *label* pada saat *training* dan *testing* dengan software *RapidMiner*. Dalam mengkategorikan apakah seorang mahasiswa layak untuk menjadi asisten dosen tentunya mahasiswa tersebut harus memiliki presensi yang tertib, rajin dalam mengerjakan kuis yang diberikan, nilai akhir pada *range* sangat paham atau paham, dan tentunya harus memiliki nilai rata-rata yang sesuai standar. Dalam menentukan standar nilai rata-rata yang akan digunakan untuk mengkategorikan seorang mahasiswa layak atau tidak untuk menjadi asisten dosen, penulis menggunakan Surat Keputusan Rektor Universitas Muhammadiyah Yogyakarta tahun 2017 dengan nomor surat: 206 /SK-UMY/IX/2017. Tentang Standar Nilai Akhir Mata Kuliah Pada Program Studi Di Lingkungan Universitas Muhammadiyah Yogyakarta Sebagai Acuan Dalam Menentukan Standar Nilai Rata-Rata Yang Digunakan.

Table 4.5 adalah data hasil dari proses transformasi data, didapatkan 5 atribut yang nantinya akan digunakan sebagai *input*, dan data inilah yang nantinya akan dimasukkan kedalam *software RapidMiner*. Kemudian data tersebut nantinya akan dilatih menggunakan metode *Neural Network*. Pada atribut NIM yang merupakan data identitas mahasiswa penulis mengganti 2 angka terakhir untuk menjaga kerahasiaan mahasiswa.

Tabel 4. 5 Data Transformation

MATA KULIAH	NIM	Rata-Rata	Presensi	Kerajinan	Nilai Akhir	Keterangan
Object Oriented	201401400AA	55	Tertib	Rajin	Sangat Paham	Tidak Layak
Object Oriented	201401400BB	30	Tertib	Rajin	Paham	Tidak Layak
Object Oriented	201401400CC	70	Tertib	Rajin	Sangat Paham	Layak
Object Oriented	201401400DD	40	Tertib	Rajin	Paham	Tidak Layak
Object Oriented	201401400EE	50	Tertib	Rajin	Paham	Tidak Layak
Object Oriented	201401400FF	45	Tertib	Rajin	Cukup Paham	Tidak Layak
Object Oriented	201401400GG	65	Tertib	Rajin	Sangat Paham	Layak
Object Oriented	201401400HH	40	Tertib	Rajin	Sangat Paham	Tidak Layak
Object Oriented	201401400II	60	Tertib	Rajin	Sangat Paham	Tidak Layak

4.2 Implementasi Algoritma

Pada tahap implementasi ini penulis melakukan pemodelan data dengan metode yang dipakai yaitu Algoritma *Neural Network*. Data yang telah melalui proses seleksi, implementasi dan transformasi akan diolah dengan menggunakan algoritma *Neural Network* dengan menggunakan *software RapidMiner*. Metode *Neural Network* menggunakan data historis untuk memprediksi peluang di masa yang akan datang.

Sebelum memasukkan data hasil *preprocessing data*, atribut mata kuliah dan NIM dihapus terlebih dahulu, karena 2 atribut tersebut tidak digunakan sebagai *input* untuk mengukur layak atau tidak. Data latih yang digunakan dari jurusan Teknik informatika memiliki jumlah data sebanyak 711 data. Data tersebut nantinya akan digunakan sebagai data latih (*training*) dan data uji (*testing*). Pada pengujian yang pertama menggunakan data 2014 sebanyak 463 data sebagai data latih dan data angkatan 2015 sebanyak 248 data sebagai data uji, seperti *table 4.6* merupakan data latih pertama yang digunakan merupakan data mahasiswa 2014, dan *table 4.7* merupakan sebagai data uji yang digunakan merupakan data mahasiswa 2015. Pada pengujian kedua tidak dilakukan pembagian seperti pengujian pertama melainkan total keseluruhan data digunakan sebagai data latih dan data uji kemudian dilakukan pengujian dengan menggunakan metode *cross validation* datang yang digunakan seperti *table 4.8*. Seluruh data yang digunakan menggunakan format *.xlsx yang nantinya akan di masukkan kedalam *software RapidMiner*.

Tabel 4. 6 Data Training Pengujian Pertama Angkatan 2014

Rata-Rata	Presensi	Kerajinan	Nilai Akhir	Keterangan
55	Tertib	Rajin	Sangat Paham	Tidak Layak
30	Tertib	Rajin	Paham	Tidak Layak
70	Tertib	Rajin	Sangat Paham	Layak
40	Tertib	Rajin	Paham	Tidak Layak
50	Tertib	Rajin	Paham	Tidak Layak
45	Tertib	Rajin	Cukup Paham	Tidak Layak
65	Tertib	Rajin	Sangat Paham	Layak
40	Tertib	Rajin	Sangat Paham	Tidak Layak

Tabel 4. 7 Data Testing Pengujian Pertama Angkatan 2015

Rata-Rata	Presensi	Kerajinan	Nilai Akhir	Keterangan
50	Tertib	Rajin	Cukup Paham	Tidak Layak
45	Tertib	Rajin	Paham	Tidak Layak
55	Tertib	Rajin	Cukup Paham	Tidak Layak
50	Tertib	Rajin	Paham	Tidak Layak
60	Tertib	Rajin	Sangat Paham	Tidak Layak
52	Tertib	Rajin	Sangat Paham	Tidak Layak
40	Tertib	Rajin	Cukup Paham	Tidak Layak
32	Tertib	Rajin	Cukup Paham	Tidak Layak

Tabel 4. 8 Pengujian Kedua Menggunakan Seluruh

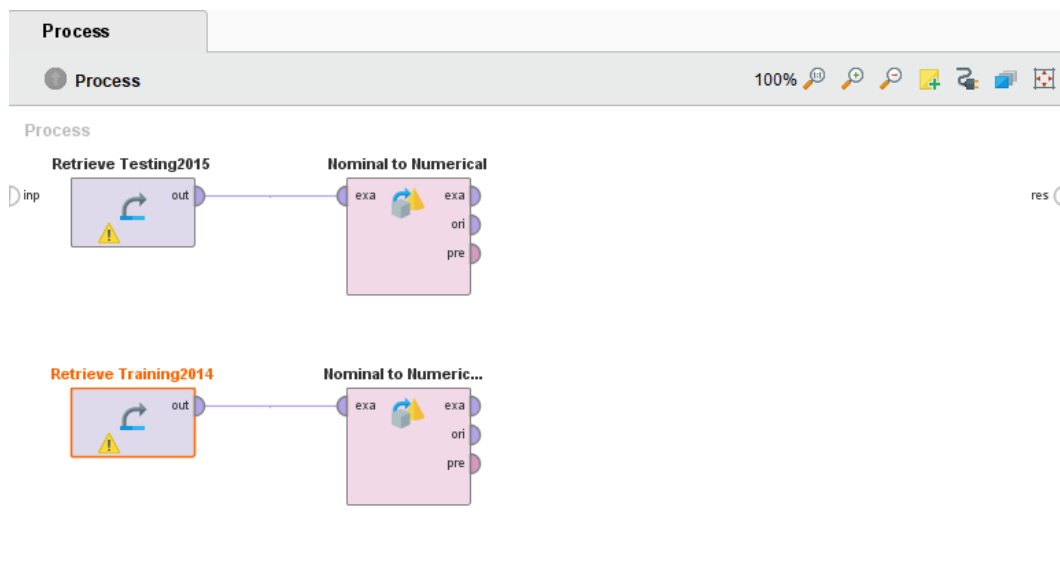
Rata-Rata	Presensi	Kerajinan	Nilai Akhir	Keterangan
0	Tertib	Kurang Rajin	Sangat Paham	Tidak Layak
0	Tertib	Kurang Rajin	Sangat Paham	Tidak Layak
0	Tertib	Kurang Rajin	Sangat Paham	Tidak Layak
16.25	Tidak Tertib	Kurang Rajin	Paham	Tidak Layak
20	Tertib	Kurang Rajin	Paham	Tidak Layak
20	Tertib	Rajin	Cukup Paham	Tidak Layak
21.25	Tertib	Kurang Rajin	Cukup Paham	Tidak Layak
21.6666667	Tertib	Rajin	Paham	Tidak Layak

Data

4.2.1 Pengujian Pertama

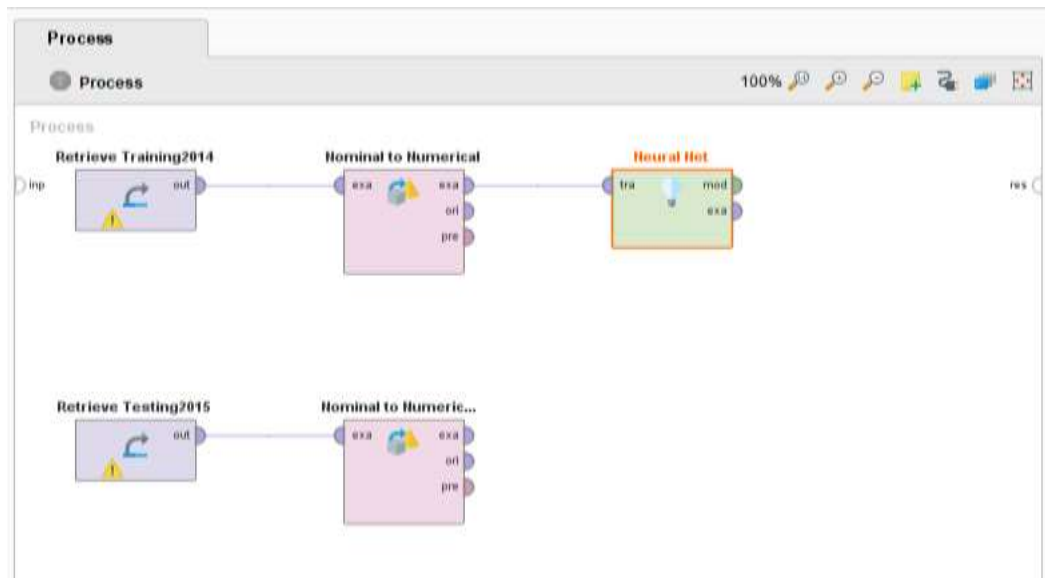
Pada tahap pengujian dengan menggunakan *software RapidMiner* penulis akan melakukan analisis terhadap mahasiswa yang layak untuk menjadi asisten dosen berdasarkan data historis mereka. Data historis yang digunakan yaitu data nilai rata-rata kuis yang telah di kerjakan di *e-learning*, data presensi, dan data nilai akhir. Proses pertama yang dilakukan yaitu melakukan *import data* yang telah di transformasikan kedalam *software RapidMiner*.

Pada pengujian pertama proses yang dilakukan yaitu memasukkan data yang akan digunakan dengan melakukan *drag dan drop file* data latih dan data uji kedalam lembar *design* kemudian tambahkan operator *nominal to numerical* untuk masing-masing *file* dan hubungkan *port out* pada masing *file* ke ke *port exa* operator *nominal to numerical* seperti pada gambar 4.8.



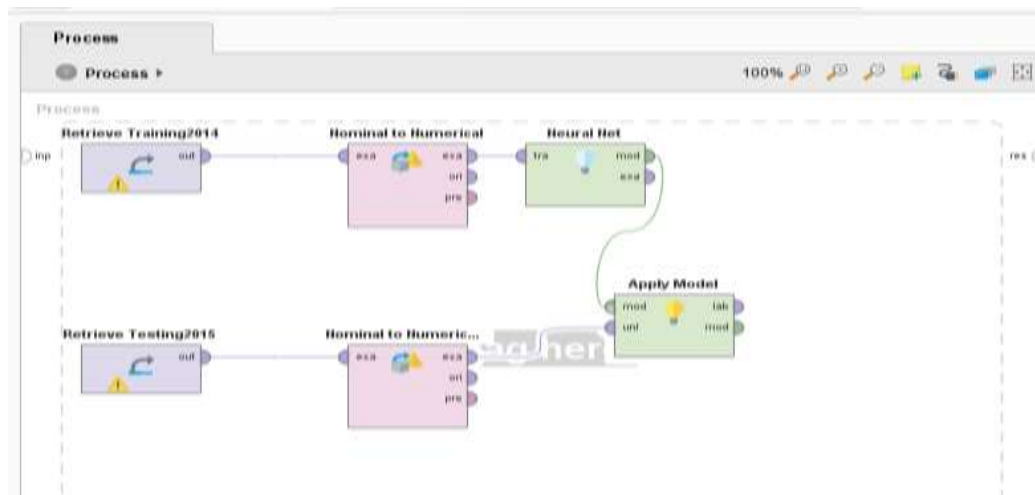
Gambar 4. 8 Menambahkan Data dan operator Nominal to Numerical

Langkah selanjutnya menambahkan algoritma yang akan digunakan dengan memasukkan operator *Neural Net*. Operator *Neural Net* hanya digunakan untuk data latih saja dan hubungkan *port exa* pada operator *nominal to numerical* ke *port tra* pada operator *Neural Net* seperti pada gambar 4.9.



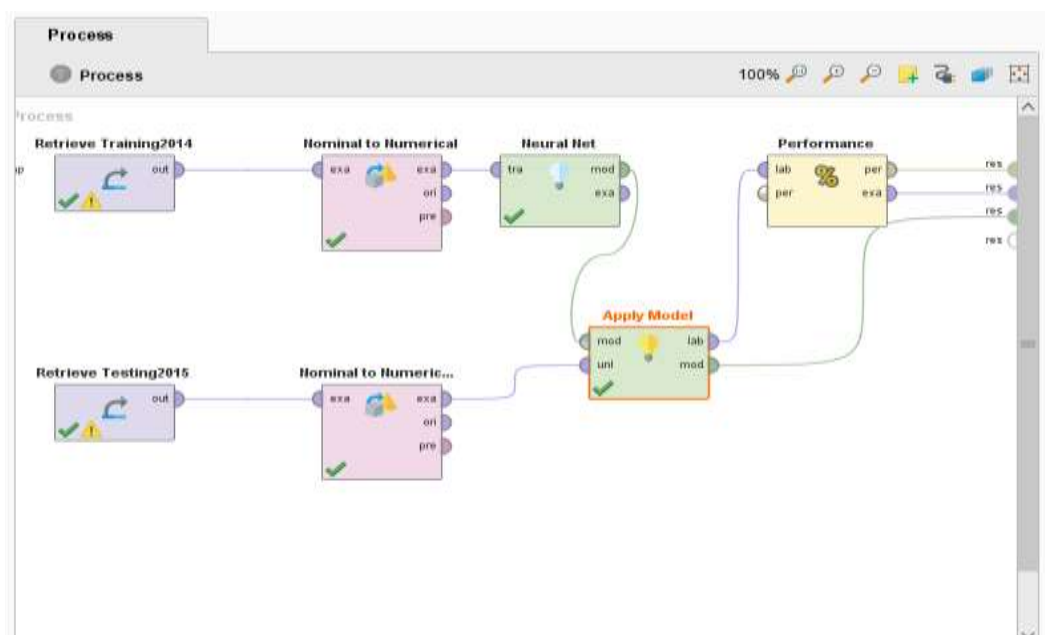
Gambar 4. 9 Menambahkan Operator Neural Net

Langkah selanjutnya yaitu menambahkan operator *Apply Model* kedalam lembar *design*, selanjutnya hubungkan *port mod* yang ada pada operator *Neural Net* dengan *port mod* yang ada pada operator *Apply Model*, selanjutnya hubungkan *port exa* yang ada pada operator *nominal to numerical* dengan *port uni* yang ada pada operator *Apply Model* seperti pada gambar 4.10.

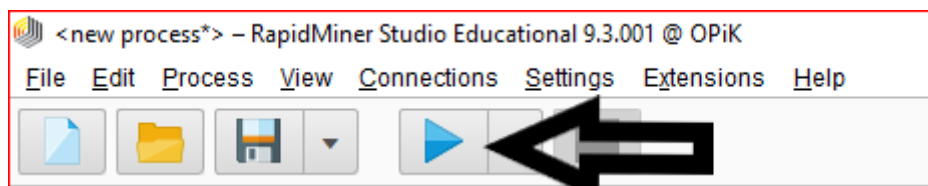


Gambar 4. 10 Menambahkan Operator Apply Model

Selanjutnya yaitu menambahkan operator *performance* kedalam lembar *design*, setelah operator *Performance* ditambahkan selanjutnya hubungkan *port lab* yang ada pada operator *Apply Model* dengan *port lab* yang ada pada operator *Performance*. *Port* yang ada pada operator *Performance* yaitu *port per* dan *exa* dihubungkan dengan *port res* yang ada pada lembar *design*, *port mod* yang ada pada operator *Apply Model* juga dihubungkan ke *port res* yang ada pada lembar *design* seperti pada gambar 4.11.



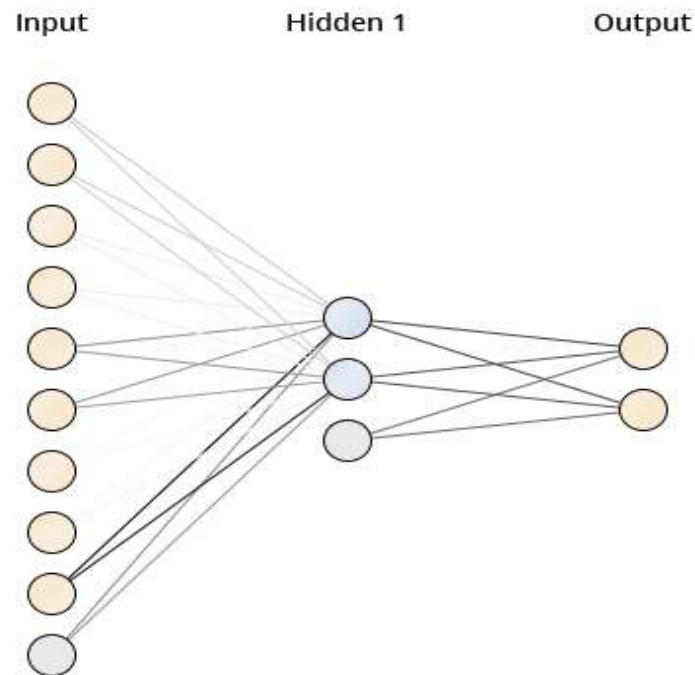
Gambar 4. 11 Menambahkan Operator Performance



Gambar 4. 12 Run RapidMiner

Setelah semuanya operator dan *file* yang digunakan terhubung, untuk mengetahui hasil dari pengujian klik *button run* yang ada pada *toolbar* bagian atas seperti gambar 4.12. Setelah *button run* di klik tunggu beberapa saat untuk komputer melakukan proses.

Setelah menunggu beberapa saat tampilan *RapidMiner* akan berpindah ke lembar *Result* yang menampilkan hasil dari proses menggunakan *Neural Network*. Hasil dari pemodelan *Neural Network* yang didapatkan yaitu topologi *feedforward* seperti pada gambar 4.13. Pemodelan *Neural Network* yang didapatkan terdiri dari 3 *layer* yaitu: *input layer*, *hiden layer* dan *output layer*. Pada *input layer* terdapat 9 *nodes*, masing-masing *nodes* berasal dari nilai yang ada pada masing-masing atribut yang digunakan. Pada *hidden layer* terdapat 2 *nodes*, setiap *nodes* pada *hidden layer* menerima *input* dari setiap *nodes* yang ada di *layer input*. Untuk *output layer* terdapat 2 *nodes* yang mana *nodes* tersebut sesuai dengan jumlah data yang ada pada atribut yang menjadi *label*.



Gambar 4. 13 Pemodelan Neural Network

Tab Performance Vektor akan menampilkan seberapa besar tingkat akurasi yang dihasilkan dengan menggunakan *Neural Network*. Pada *table 4.9* dapat dilihat bahwa tingkat akurasi yang didapatkan menggunakan *Neural Network* yaitu sebesar 97.18%. Untuk hasil *class precision* layak yaitu sebesar 96.30%, dan *class precision* tidak layak yaitu sebesar 97.60%. Untuk *class recall* layak yaitu sebesar 95.12% dan *class recall* tidak layak sebesar 98.19%.

Tabel 4. 9 Confusion Matrix Pengujian Pertama

Accuracy: 97.18%			
	True Layak	True Tidak Layak	Class Precision
Pred. Layak	82	7	92.13%
Pred. Tidak Layak	0	159	100.00%
Class Recall	100.00%	95.78%	

Akurasi adalah nilai yang mendefinisikan tingkat kedekatan hasil prediksi yang diproses oleh sistem *RapidMiner* dengan data yang telah diberikan. *Precision* merupakan *class* yang mengukur tingkat ketepatan informasi dari sistem *RapidMiner* dengan data yang telah diberikan. *Recall* merupakan *class* yang mengukur tingkat keberhasilan proses sistem *RapidMiner* untuk menemukan kembali informasi. *Table 4.10* merupakan *table confusion matrix*, *table* ini penulis tambahkan untuk membantu menjelaskan penjabaran rumus untuk mencari hasil akurasi, *precision*, dan *recall* seperti pada *Table 4.10*.

Tabel 4. 10 Table Confussion Matrix

Nilai Sebenarnya			
		TRUE	FALSE
Nilai Prediksi	TRUE	TP (True Positive) Correct Result	FP (False Positive) Unexpected Result
	FALSE	FN (False Negative) Missing Result	TN (True Negative) Correct Absence of Result

- Rumus untuk menghitung *class Precision*

$$Precision = \frac{TP}{TP + FP}$$

- Rumus untuk menghitung *class Recall*

$$Recall = \frac{TP}{TP + FN}$$

- Rumus menghitung tingkat akurasi

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Dari penjabaran rumus diatas maka dapat dilakukan perhitungan matematis untuk menghitung *precision*, *recall*, dan akurasi sebagai berikut:

- Precision* dan *recall class* Layak

$$Precision = \frac{82}{82 + 7} = \frac{82}{89} = 0.921348 = 92.13\%$$

$$Recall = \frac{82}{82 + 0} = \frac{82}{82} = 1 = 100.00\%$$

b. *Precision* dan *recall class* Tidak Layak

$$Precision = \frac{159}{159 + 0} = \frac{159}{159} = 1 = 100.00\%$$

$$Recall = \frac{159}{159 + 7} = \frac{159}{166} = 0.957831 = 95.78\%$$

c. Tingkat akurasi

$$Accuracy = \frac{159 + 82}{159 + 82 + 7 + 0} = \frac{241}{248} = 0.971774 = 97.18\%$$

Dari perhitungan manual diatas maka didapatkan hasil yang sama dengan gambar 4.19 dengan pengolahan oleh sisten *RapidMiner* dengan menggunakan metode *Neural Network*. Maka dapat diartikan bahwa pengujian *Neural Network* dengan *software RapidMiner* memiliki hasil ketepatan yang sangat baik dibuktikan dengan hasil *class precision* yang tinggi, begitu juga dengan hasil *class recall* yang tinggi membuktikan bahwa tingkat tingkat keberhasilan proses *Neural Network* sangat baik, dan hasil akurasi yang mencapai 97.18% membuktikan bahwa *Neural Network* memiliki tingkat kedekatan yang sangat baik.

4.2.2 Pengujian Kedua

Pengujian kedua dilakukan dengan menggunakan operator *cross validation*, pengujian dengan menggunakan *cross validation* sedikit berbeda dengan pengujian pertama yang sebelumnya telah dilakukan. Pada pengujian ini data latih dan data uji tidak dilakukan pemisahan manual melainkan tetap didalam satu *file excel*, yang akan melakukan pemisahan data yaitu operator *cross validation*, untuk pembagian jumlah *data training* dan *data testing* bergantung pada seberapa banyak jumlah pengujian yang diinginkan.

Pada pengujian kedua menggunakan *cross validation* terdapat 2 parameter yang wajib ditentukan, yaitu :

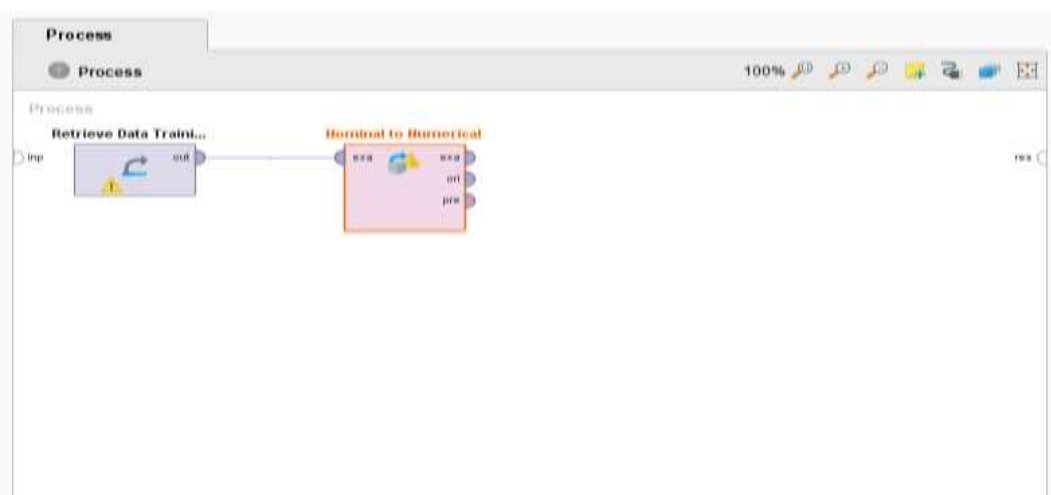
1. *Number of Folds*: parameter *number of folds* berguna untuk menentukan seberapa banyak pengujian yang diinginkan, jika *number of folds* dimasukkan angka 10 maka akan dilakukan 10 kali pengujian, dan untuk pembagian *data training* dan *data testing* dibagi secara *default* oleh nominal *cross validation* dengan komposisi 90% dari data akan menjadi *data training* dan sisanya 10% akan menjadi *data testing*. Proses ini akan terus berulang sebanyak jumlah *fold* yang dimasukkan dan untuk pada masing pengujian akan dilakukan pertukaran *data taraining* dan *data testing*.
2. *Sampling Type*: Parameter *sampling type* berguna untuk menentukan jenis pengambilan data. Masing-masing jenis pengambilan data di *cross validation* memiliki pola pengambilan data yang berbeda. Terdapat 3 jenis pengambilan data yaitu:
 - a. *Linear sampling*, cara penambilan data secara *linear* tanpa merubah urutan contoh yang ada pada *data set*.
 - b. *shuffled_sampling*: proses pengambilan data secara acak dari *data set*, dan
 - c. *statified sampling*: proses pengambilan data *statified sampling* hampir sama dengan *shuffled sampling* yaitu dengan cara diacak, tetapi yang membedakan yaitu pada *sattisfied sampling* memastikan distribusi atribut pada *subset* sama dengan di seluruh *data set*.

Untuk melakukan pengujian menggunakan operator *cross validation* Langkah pertama yang dilalui pada proses pengujian ini yaitu seperti pada gambar 4.14 lakukan *drag* dan *drop file* yang akan digunakan.



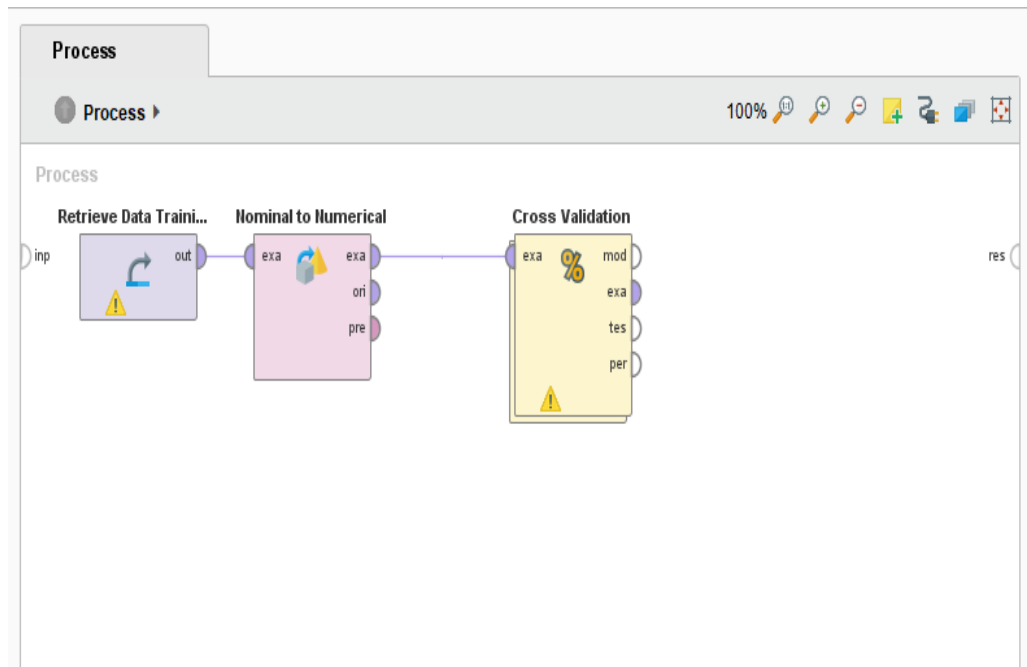
Gambar 4. 14 Drag dan Drop File yang Digunakan

Langkah selanjutnya tambahkan operator *nominal to numerical* seperti pada gambar 4.15. Operator *nominal to numerical* ini berfungsi untuk melakukan konversi nilai yang non-numeric pada masing-masing atribut menjadi *numeric*.



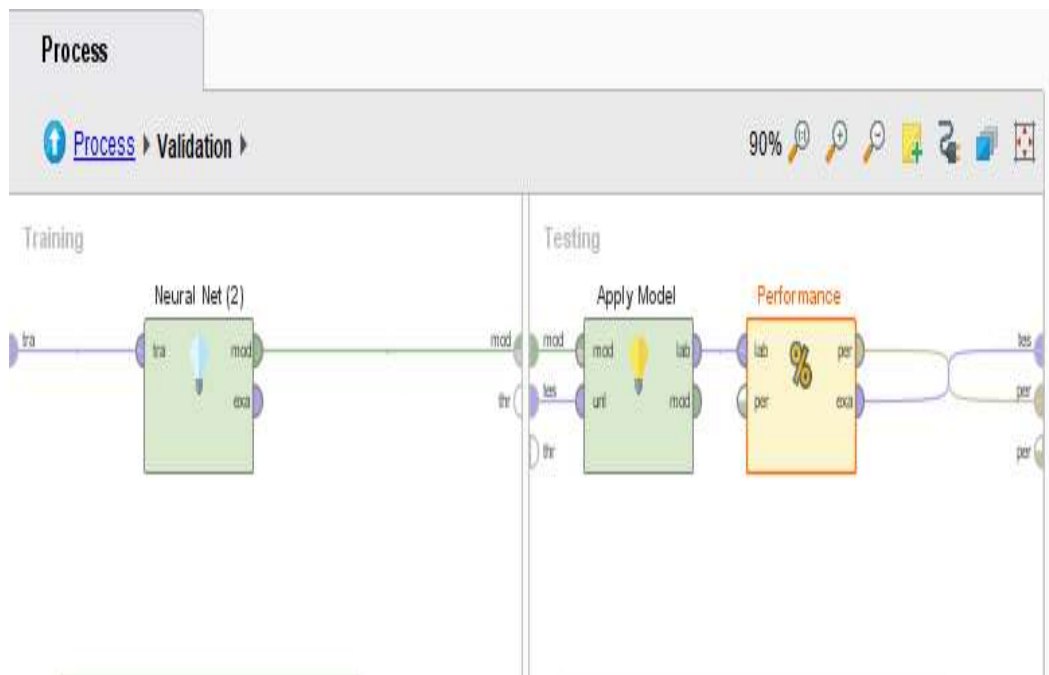
Gambar 4. 15 Nominal to numerical

Langkah selanjutnya yaitu menambahkan operator *cross validation* kemudian hubungkan *port exa* di operator *nominal to numerical* dengan *port exa* pada operator *cross validation* seperti pada gambar 4.16.



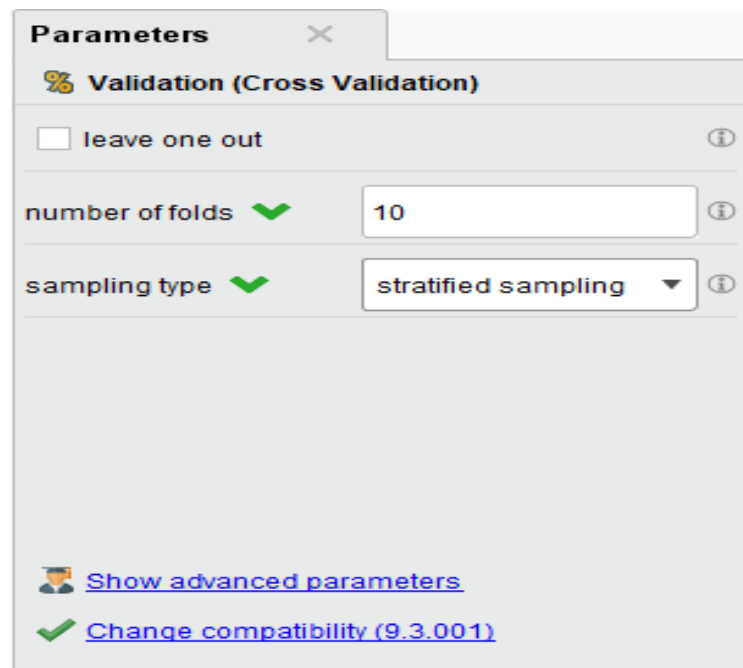
Gambar 4. 16 Cross Validation

Setelah operator *cross validation* berhasil ditambahkan selanjutnya mengatur operator *cross validation* dengan cari *double klik* pada operator *cross validation*. Kemudian tambahkan operator *Neural Net* pada kolom *training* dan hubungkan *port tra* dan *port mod* pada kolom *training*, kemudian untuk di kolom *testing* tambahkan dua operator yaitu operator *apply model* dan operator *performance*, selanjutnya hubungkan *port mod* dan *port tes* yang ada di kolom *testing* dengan *port mod* dan *port uni* di operator *apply model*, kemudian hubungkan *port lab* di operator *apply model* dengan *port lab* di operator *performance*, kemudian hubungkan *port per* ke *port per* di kolom *testing* dan *port exa* ke *port tes* di kolom *testing* seperti gambar 4.17.



Gambar 4. 17 Setting Cross Validation

Langkah selanjutnya yaitu melakukan pengaturan *parameters cross validation*. Pada langkah ini parameter *number of folds* masukkan angka 10. Angka yang dimasukkan pada parameter *number of folds* menentukan jumlah pengujian yang akan dilakukan. Pada parameter *sampling type* gunakan *stratified sampling*. Seperti pada gambar 4.18.



Gambar 4. 18 Parameters Cross Validation

Sebelum pengujian kedua menggunakan operator *cross validation* dijalankan pastikan semua operator telah terhubung, selanjutnya klik *button run* pada *toolbar* seperti pada gambar 4.12, dan tunggu beberapa saat untuk komputer melakukan pelatihan dan pengujian model.

Hasil pengujian dengan menggunakan *cross validation* dapat dilihat pada *table 4.11*. Pada hasil pengujian kedua menggunakan *cross validation* memiliki tingkat akurasi yang lebih tinggi dari pengujian pertama, sehingga dapat disimpulkan bahwa proses *training* dan *testing* dengan *cross validation* memiliki tingkat kedekatan data yang sangat baik antara data yang diolah dengan *software* dengan data yang diberikan. Untuk hasil *class precision* dan *class recall* terjadi sedikit penurunan.

Tabel 4. 11 Hasil Confussion Matrix Cross Validation

Accuracy: 98.18%			
	True Layak	True Tidak Layak	Class Precision
Pred. Layak	255	9	96.72%
Pred. Tidak Layak	4	433	99.08%
Class Recall	98.51%	97.96%	

Pada pengujian menggunakan *cross validation* terdapat beberapa jenis pengambilan data, untuk pengujian ini penulis tidak hanya menggunakan 1 jenis pengambilan data melainkan menggunakan 3 jenis pengambilan data seperti pada *table 4.12*, tujuannya yaitu untuk membandingkan dan untuk mengetahui jenis pengambilan data mana yang terbaik. Ada pun hasilnya dapat di lihat pada *table 4.12*. Dapat dilihat bahwa jenis pengambilan *Stratified sampling* mendapatkan hasil yang sama dengan *shuffled sampling* tingkat akurasi sebesar 98.74%. *Stratified sampling* merupakan jenis pengambilan data dengan cara membangun himpunan acak, yang membedakan dengan *shuffle sampling* yaitu *stratified sampling* memastikan bahwa distribusi *subset* sama dengan seluruh *example set*.

Tabel 4. 12 Hasil Sampling Type

Sampling Type	Hasil
Stratified Sampling	98.18%
Shuffled Sampling	98.18%
Linear Sampling	93.95%

4.3 Visualisasi Hasil

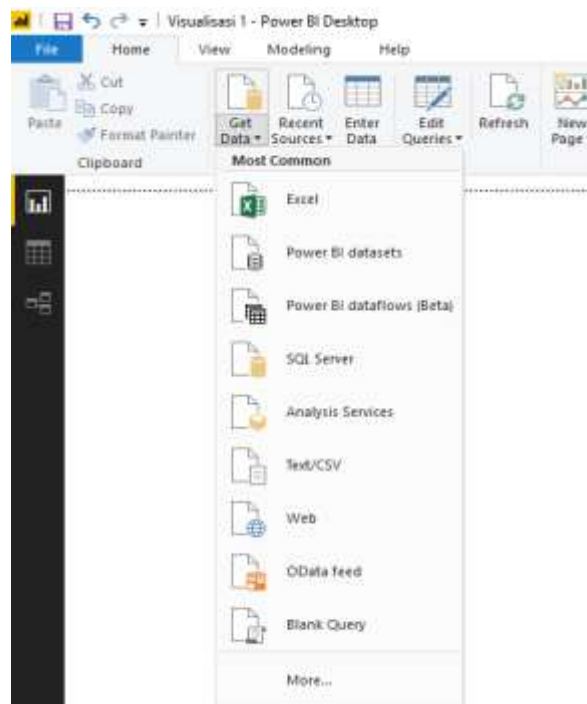
Setelah melakukan pengujian dengan algoritma *Neural Network* menggunakan *software RapidMiner* untuk melihat seberapa besar tingkat kecocokan hasil pengujian *Neural Network* dengan data yang diberikan, selanjutnya dilakukan penggabungan data hasil prediksi menggunakan

RapidMiner kedalam *file Excel*, seperti pada *table 4.13*. Tujuan dilakukannya penggabungan data seperti pada *table 4.13* yaitu untuk melihat didata yang mana terjadi perbedaan hasil prediksi antara data yang diberikan dengan hasil prediksi dengan *Neural Network*. Sebagaimana yang telah di jabarkan pada pengujian pertama didapatkan hasil prediksi *Neural Netwrok* yaitu sebesar 97.18%. Selain itu dilakukannya penggabungan data penulis juga akan membuatkan hasil visualisasi prediksi dengan *software Power BI*.

Tabel 4. 13 Penggabungan Data Hasil Prediksi

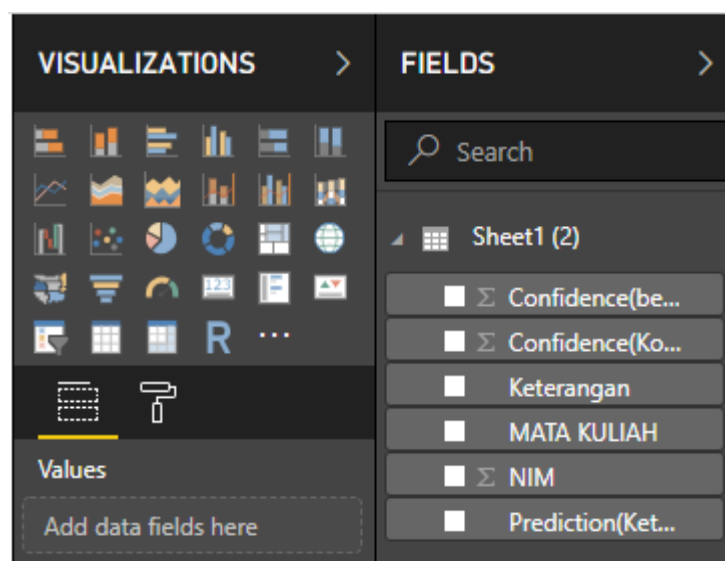
MATA KULIAH	NIM	Keterangan	Prediction(Keterangan)
Object Oriented Analysis and Design	201501400AA	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400BB	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400CC	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400DD	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400EE	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400FF	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400GG	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400HH	Belum Kompeten	Belum Kompeten
Object Oriented Analysis and Design	201501400II	Belum Kompeten	Belum Kompeten

Data hasil prediksi dengan *Neural Network* yang telah digabungkan seperti pada *table 4.13*, selanjutnya data tersebut dimasukkan kedalam *software Power BI*. Dengan melakukan proses pertama yaitu: *Get Data*, kemudian pilih *Excel*. Seperti pada gambar 4.19.



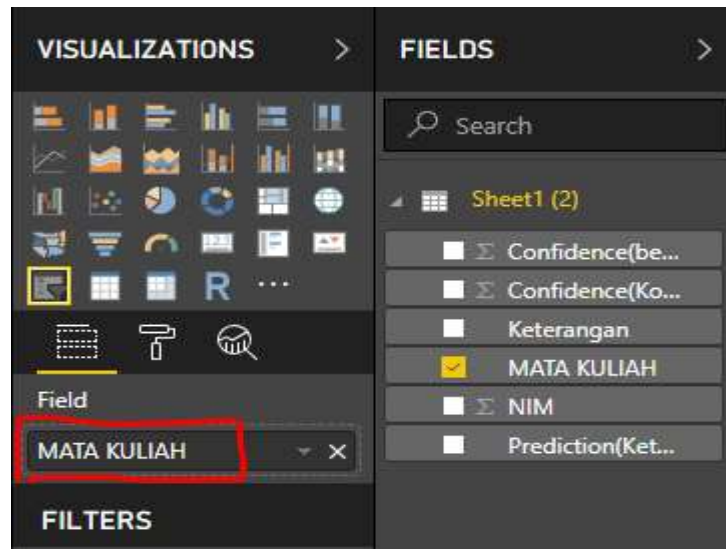
Gambar 4. 19 Get Data Power BI

Setelah berhasil memasukkan data *Excel* yang akan digunakan, maka seluruh kolom yang ada didalam file *Excel* akan muncul pada bagian *Fields* *Power BI* seperti pada gambar 4.20.



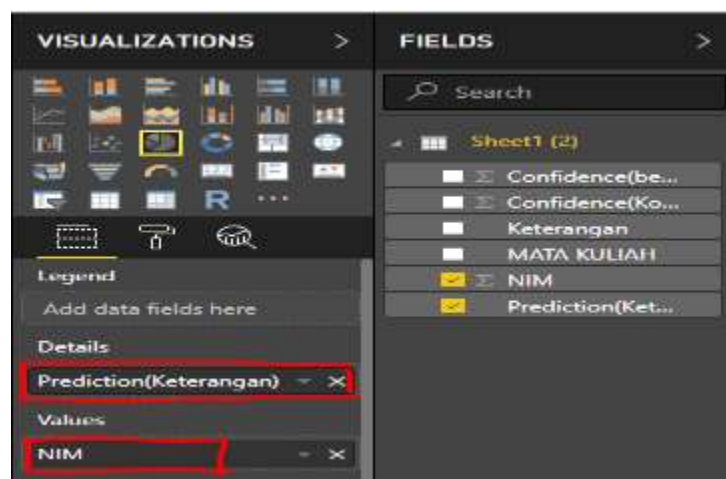
Gambar 4. 20 Tampilan Fields

Setelah dipastikan keseluruhan kolom sudah tertera pada bagian *Fields*, selanjutnya memilih visualisasi yang akan digunakan dengan memilih *Slicer*, *Pie chart*, dan *table* pada bagian *visualization Power BI*. Pada visualisasi *Slicer* masukkan kolom Mata Kuliah pada bagian *field*, seperti pada gambar 4.21.



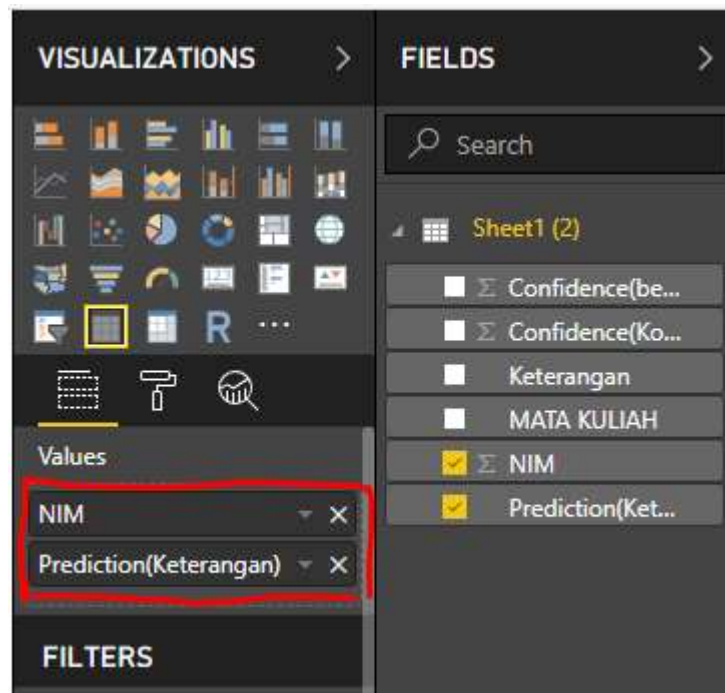
Gambar 4. 21 Visualisasi Slicer

Kemudian pada visualisasi *Pie chart* masukkan kolom *Prediction(Keterangan)* pada bagian *Details* dan kolom NIM pad bagian *Values*, Seperti pada gambar 4.22.



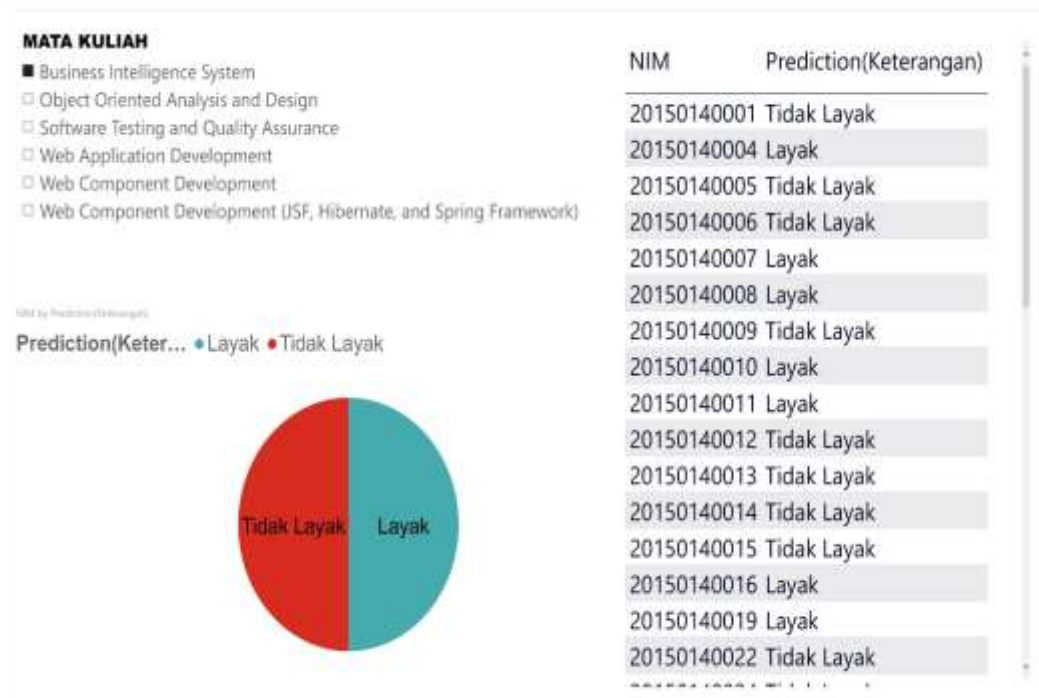
Gambar 4. 22 Visualisasi Pie Chart

Selanjutnya untuk visualisasi *table* masukkan kolom NIM dan *Prediction(Keterangan)* pada bagian values, seperti pada gambar 4.23.



Gambar 4. 23 Visualisasi Table

Maka ketika dipilih salah satu mata kuliah pada tampilan visualisasi *Slicer* akan menampilkan presentase seberapa banyak mahasiswa yang layak menjadi calon asisten dosen atau tidak hasil persentase ditampilkan pada visualisasi *Pie Chart*, dan pada visualisasi *table* akan menampilkan list NIM mahasiswa yang diprediksi layak dan tidak layak, seperti pada gambar 4.24. Dari hasil visualisasi pada gambar 4.24, mata kuliah yang dipilih yaitu *Business Intelligence System* dari hasil visualisasi diketahui bahwa dari keseluruhan mahasiswa yang mengambil mata kuliah tersebut diprediksi sebanyak 50% mahasiswa yang layak untuk menjadi calon asisten dosen dan 50% sisanya belum layak menjadi calon asisten dosen. Dengan komposisi 40 orang mahasiswa yang mengambil mata kuliah *Business Intelligence System*, diprediksi 20 orang mahasiswa yang layak menjadi calon asisten dosen dan 20 orang mahasiswa yang belum layak menjadi asisten calon asisten dosen.



Gambar 4. 24 Hasil Visualisasi Power BI