

BAB III

METODOLOGI PENELITIAN

3.1 Jenis Penelitian

Jenis penelitian yang digunakan dalam tugas akhir ini adalah penelitian terapan. Metode penelitian ini memiliki tujuan untuk menerapkan, menguji, dan mengevaluasi kemampuan tiap metode *data transformation* yang diimplementasikan pada SIMRS dalam memecahkan permasalahan praktis. Dengan melakukan penelitian ini, diharapkan metode yang sudah diujikan dapat membantu dan meningkatkan kinerja rumah sakit dalam memanajemen sistem informasi.

3.2 Pendekatan Penelitian

Penelitian ini menggunakan metode deskriptif-studi kasus dengan pendekatan jenis data kualitatif dan kuantitatif.

Metode deskriptif adalah metode prosedur pemecahan masalah yang meneliti suatu objek dengan menggambarkan keadaannya. Penelitian deskriptif digunakan dalam membuat deskripsi atau gambaran secara sistematis, faktual, dan akurat.

Metode studi kasus merupakan strategi penelitian untuk melakukan pengujian secara terpusat pada satu objek tertentu. Tujuan dari penelitian studi kasus adalah untuk memberikan gambaran secara detail tentang sifat dan karakter dari kasus yang diteliti.

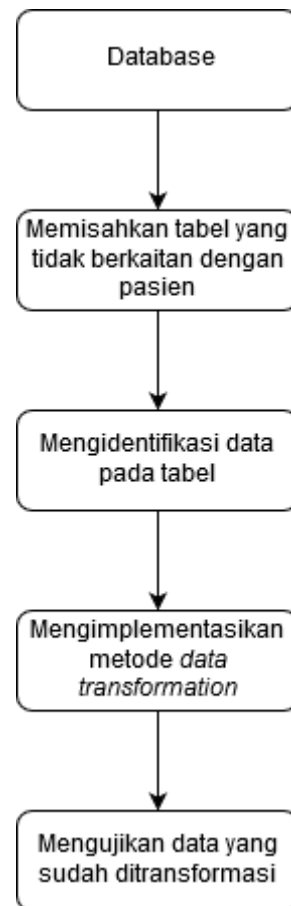
Pendekatan jenis data kualitatif adalah jenis pendekatan yang menggunakan data kualitatif seperti kalimat, pernyataan, gambar, dan indeks tertentu, sedangkan kuantitatif merupakan pendekatan yang menggunakan data angka atau data yang dapat dikonversi menjadi angka.

3.3 Jenis dan Sumber Data

Dalam tugas akhir ini, Sumber data yang digunakan sebagai objek penelitian adalah data primer. Data primer adalah data yang diperoleh langsung dari pihak yang berkaitan langsung/sumber pertama. Data yang diambil dari sumber data adalah *database* sistem informasi manajemen rumah sakit di Jawa Tengah, Indonesia. *Database* tersebut adalah *database* Oracle versi 10.2.0.1.0 dengan besar + 300 MB yang berisi 171 tabel.

3.4 Langkah Analisis Data

Pada subbab ini akan dilakukan langkah-langkah analisis dan perancangan penelitian seperti Gambar 3.1.



Gambar 3.1 Langkah analisis data

3.4.1 Database

Database Sistem Informasi Manajemen Rumah Sakit (SIMRS) diambil dari Ainayya yang sudah dibersihkan dengan metode *data cleansing*. *Database* tersebut di-*import* secara manual ke dalam komputer penulis. Setelah itu data di-*export* menjadi *file* berekstensi *xlsx*.

3.4.2 Memisahkan Tabel yang Tidak Berkaitan dengan Pasien

Dengan menyesuaikan batasan masalah, yaitu tabel yang digunakan hanya berkaitan dengan pasien, maka dari 171 tabel yang ada di *database* akan dikurangi hingga menyisakan 14 tabel. Adapun tabel-tabel yang tersisa terlihat pada Tabel 3.1.

Tabel 3.1 Empat belas tabel yang berkaitan dengan pasien

Nama Tabel
ASURANSI_KEPESERTAAN_VISIT
ANTRIAN
BILLING
HASIL_PEMERIKSAAN_LAB
KUNJUNGAN
KUNJUNGAN_BPJS
LOG_BATAL_KUNJUNGAN
PASIEN
PEMBAYARAN
PENDUDUK
PENJUALAN_RESEP
RAWAT_DARURAT
RAWAT_JALAN
VISIT

ASURANSI_KEPESERTAAN_VISIT

Tabel ASURANSI_KEPESERTAAN_VISIT memiliki jumlah data yang diinput sebanyak 500 *records* dengan 5 atribut. Selengkapnya dapat dilihat pada Gambar 3.2.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
NO_POLIS	VARCHAR2(20)	No	-	-
ID_VISIT	NUMBER(24,0)	No	-	-
ID_ASURANSI_PRODUK	NUMBER(10,0)	No	-	-
ID_PERUSAHAAN_ASURANSI	NUMBER	Yes	-	-
				1 - 5

Gambar 3.2 Tabel ASURANSI_KEPESERTAAN_VISIT

ANTRIAN

Tabel ANTRIAN memiliki jumlah data yang diinput sebanyak 214 *records* dengan 5 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.3.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
ID_KUNJUNGAN	NUMBER(10,0)	No	-	-
ID_ANTRI	NUMBER(10,0)	No	-	-
ID_BED	NUMBER(10,0)	No	-	-
STATUS	VARCHAR2(255)	No	-	-
				1 - 5

Gambar 3.3 Tabel ANTRIAN

BILLING

Tabel BILLING memiliki jumlah data yang diinput sebanyak 500 *records* dengan 7 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.4.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
ID_VISIT	NUMBER(24,0)	No	-	-
WAKTU	TIMESTAMP(6)	No	-	-
ID_PEGAWAI_PETUGAS	NUMBER(10,0)	No	-	-
STATUS PEMBAYARAN	VARCHAR2(4000)	No	-	-
TOTAL_TAGIHAN	FLOAT	Yes	-	-
JENIS	VARCHAR2(10)	Yes	-	-
				1 - 7

Gambar 3.4 Tabel BILLING

HASIL_PEMERIKSAAN_LAB

Tabel HASIL_PEMERIKSAAN_LAB memiliki jumlah data yang diinput sebanyak 500 *records* dengan 8 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.5.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(11,0)	No	-	1
TANGGAL	TIMESTAMP(6)	No	-	-
KETERANGAN	VARCHAR2(4000)	Yes	-	-
ID_VISIT	NUMBER(24,0)	No	-	-
ID_PEGAWAI	NUMBER(10,0)	Yes	-	-
ID_DOKTER_PENGIRIM	NUMBER(10,0)	Yes	-	-
JENIS_DOKTER_PENGIRIM	VARCHAR2(50)	Yes	-	-
DOKTER_PENGIRIM_LUAR	VARCHAR2(500)	Yes	-	-
				1 - 8

Gambar 3.5 Tabel HASIL_PEMERIKSAAN_LAB

KUNJUNGAN

Tabel KUNJUNGAN memiliki jumlah data yang diinput sebanyak 500 *records* dengan 17 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.6.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
NO_ANTRIAN	NUMBER(24,0)	No	-	-
WAKTU	TIMESTAMP(6)	No	-	-
ID_PASIEN	NUMBER(10,0)	No	-	-
NO_KUNJUNGAN_PASIEN	NUMBER(10,0)	No	'1'	-
ID_LAYANAN	NUMBER(10,0)	Yes	-	-
ID_PENDUDUK_DPJP	NUMBER(10,0)	Yes	-	-
ID_PENDUDUK_PENANGGUNGJAWAB	NUMBER(10,0)	Yes	NULL	-
ID_PENDUDUK_PENGANTAR	NUMBER(10,0)	Yes	NULL	-
ID_RUJUKAN	NUMBER(10,0)	Yes	NULL	-
RENCANA_CARA_BAYAR	VARCHAR2(4000)	No	-	-
ID_BED	NUMBER(10,0)	Yes	-	-
STATUS	VARCHAR2(4000)	No	-	-
WAKTU_KELUAR	TIMESTAMP(6)	Yes	NULL	-
ID_VISIT	NUMBER(24,0)	No	-	-
ID_KUNJUNGAN_INDUK	NUMBER(24,0)	Yes	-	-
ID_PENDUDUK_KELUARGA	NUMBER(10,0)	Yes	-	-

Gambar 3.6 Tabel KUNJUNGAN

KUNJUNGAN_BPJS

Tabel KUNJUNGAN_BPJS memiliki jumlah data yang diinput sebanyak 42 records dengan 22 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.7.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
ID_BILLING	NUMBER(10,0)	No	-	-
ID_POLI_SEP	NUMBER(2,0)	Yes	-	-
KODE_PISAT	VARCHAR2(10)	No	-	-
KODE_JENIS_BPJS	NUMBER(2,0)	Yes	-	-
NAMA_JENIS_BPJS	VARCHAR2(100)	Yes	-	-
JENIS_PERAWATAN	NUMBER(2,0)	Yes	-	-
ID_KELAS_BPJS	NUMBER(2,0)	Yes	-	-
KELAS_BPJS	VARCHAR2(50)	Yes	-	-
ID_PROVIDER	VARCHAR2(100)	Yes	-	-
NAMA_PROVIDER	VARCHAR2(100)	Yes	-	-
NO_RUJUKAN	VARCHAR2(200)	Yes	-	-
TANGGAL_RUJUKAN	TIMESTAMP(6)	Yes	-	-
TANGGAL_SEP	TIMESTAMP(6)	No	-	-
ID_DIAGNOSA_AWAL	NUMBER(11,0)	No	-	-
CATATAN	CLOB	Yes	-	-
ID_VISIT	NUMBER(10,0)	No	-	-
NO_SEP	VARCHAR2(100)	Yes	-	-
LAKALANTAS	NUMBER(2,0)	Yes	-	-
LOKASILAKA	VARCHAR2(200)	Yes	-	-
ID_ASAL_RUJUKAN	VARCHAR2(200)	Yes	-	-
ASAL_RUJUKAN	VARCHAR2(1000)	Yes	-	-

1 - 22

Gambar 3.7 Tabel KUNJUNGAN_BPJS

LOG_BATAL_KUNJUNGAN

Tabel LOG_BATAL_KUNJUNGAN memiliki jumlah data yang diinput sebanyak 15 *records* dengan 5 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.8.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
TANGGAL_BATAL	TIMESTAMP(6)	No	-	-
ALASAN	VARCHAR2(1000)	Yes	-	-
INFO	VARCHAR2(4000)	Yes	-	-
ID_VISIT	NUMBER(10,0)	No	-	-
				1 - 5

Gambar 3.8 Tabel LOG_BATAL_KUNJUNGAN

PASIEN

Tabel PASIEN memiliki jumlah data yang diinput sebanyak 500 *records* dengan 2 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.9.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
ID_PENDUDUK	NUMBER(10,0)	No	-	-
				1 - 2

Gambar 3.9 Tabel PASIEN

PEMBAYARAN

Tabel PEMBAYARAN memiliki jumlah data yang diinput sebanyak 500 *records* dengan 8 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.10.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
ID_VISIT	NUMBER(24,0)	No	-	-
TOTAL_TAGIHAN	FLOAT	No	-	-
JUMLAH_BAYAR	FLOAT	No	-	-
SISA_TAGIHAN	FLOAT	No	-	-
TOTAL_BILLING_PENJUALAN	FLOAT	No	-	-
KLAIM_BPJS	FLOAT	No	-	-
DISKON	FLOAT	No	-	-
				1 - 8

Gambar 3.10 Tabel PEMBAYARAN

PENDUDUK

Tabel PENDUDUK memiliki jumlah data yang diinput sebanyak 500 *records* dengan 10 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.11.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
NO_IDENTITAS	VARCHAR2(30)	Yes	-	-
NAMA	VARCHAR2(50)	No	-	-
JENIS_KELAMIN	VARCHAR2(4000)	No	-	-
GOL_DARAH	CHAR(2)	Yes	-	-
TANGGAL_LAHIR	DATE	Yes	-	-
NO_KARTU_KELUARGA	VARCHAR2(20)	Yes	-	-
POSISI_DI_KELUARGA	VARCHAR2(4000)	Yes	-	-
SIP	VARCHAR2(50)	Yes	-	-
IBU_KANDUNG	VARCHAR2(300)	Yes	-	-
				1 - 10

Gambar 3.11 Tabel PENDUDUK

PENJUALAN_RESEP

Tabel PENJUALAN_RESEP memiliki jumlah data yang diinput sebanyak 500 *records* dengan 4 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.12.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
ID_PENJUALAN	NUMBER(10,0)	No	-	-
ID_VISIT	NUMBER(10,0)	Yes	-	-
ID_DOKTER	NUMBER(10,0)	Yes	-	-
				1 - 4

Gambar 3.12 Tabel PENJUALAN_RESEP

RAWAT_DARURAT

Tabel RAWAT_DARURAT memiliki jumlah data yang diinput sebanyak 57 *records* dengan 37 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.13.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
ID_VISIT	NUMBER(24,0)	No	-	-
WAKTU	TIMESTAMP(6)	No	-	-
ID_ALASAN_DATANG	NUMBER(10,0)	No	-	-
KETERANGAN	CLOB	Yes	-	-
WAKTU Kejadian	TIMESTAMP(6)	Yes	-	-
TEMPAT Kejadian	VARCHAR2(50)	Yes	-	-
WAKTU_TIBA	TIMESTAMP(6)	Yes	-	-
TRANSPORTASI	VARCHAR2(50)	Yes	-	-
ID_PETUGAS	NUMBER(10,0)	No	-	-
WAKTU_PERIKSA	TIMESTAMP(6)	Yes	-	-
ID_JENIS_KASUS	NUMBER	Yes	-	-
WAKTU_MULAI	TIMESTAMP(6)	Yes	-	-
WAKTU_SELESAI	TIMESTAMP(6)	Yes	-	-
KATEGORI	NUMBER(10,0)	Yes	-	-
KET_KATEGORI	VARCHAR2(50)	Yes	-	-
WARNA	VARCHAR2(20)	Yes	-	-
ANAMNESE	CLOB	Yes	-	-
KEADAAN_UMUM	CLOB	Yes	-	-
KEADAAN_KHUSUS	CLOB	Yes	-	-
PENUNJANG	CLOB	Yes	-	-
OKSIGEN	VARCHAR2(50)	Yes	-	-
INFUS	VARCHAR2(50)	Yes	-	-
ORAL	VARCHAR2(50)	Yes	-	-
INJEKSI	VARCHAR2(50)	Yes	-	-
ID_TINDAK_LANJUT	NUMBER(10,0)	No	-	-
CATATAN	CLOB	Yes	-	-
ID_USER	NUMBER(10,0)	No	-	-
RUJUKAN	NUMBER(10,0)	No	-	-
ID_KEADAAN_KELUAR	NUMBER(10,0)	Yes	-	-
ID_CARA_KELUAR	NUMBER(10,0)	Yes	-	-
ID_KELUAR_PEMERIKSAAN_LANJUT	NUMBER(24,0)	Yes	-	-
ID_DOKTER_JAGA	NUMBER(10,0)	Yes	-	-
ID_PERAWAT_JAGA	NUMBER(10,0)	Yes	-	-
SHIFT	VARCHAR2(40)	Yes	-	-
ID_BED	NUMBER(10,0)	Yes	-	-
ID_RS_RUJUKAN	NUMBER(10,0)	Yes	-	-

1 - 37

Gambar 3.13 Tabel RAWAT_DARURAT

RAWAT_JALAN

Tabel RAWAT_JALAN memiliki jumlah data yang diinput sebanyak 63 *records* dengan 24 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.14.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
ID_VISIT	NUMBER(24,0)	No	-	-
WAKTU	TIMESTAMP(6)	No	-	-
ID_BED	NUMBER	Yes	-	-
ID_DOKTER	NUMBER(10,0)	Yes	-	-
ANAMNESE	CLOB	Yes	-	-
ID_JENIS_KASUS	NUMBER(10,0)	Yes	-	-
ID_TINDAK_LANJUT	NUMBER(10,0)	Yes	-	-
CATATAN	CLOB	No	-	-
ID_USER	NUMBER(10,0)	Yes	-	-
TENSI	VARCHAR2(30)	No	-	-
NADI	VARCHAR2(30)	No	-	-
SUHU	VARCHAR2(30)	No	-	-
NAFAS	VARCHAR2(30)	No	-	-
TINGGI	VARCHAR2(30)	No	-	-
BERAT	VARCHAR2(30)	No	-	-
ID_KEADAAN_KELUAR	NUMBER(10,0)	Yes	-	-
ID_CARA_KELUAR	NUMBER(10,0)	Yes	-	-
ID_KELUAR_PEMERIKSAAN_LANJUT	NUMBER(10,0)	Yes	-	-
ID_RUJUKAN	NUMBER(10,0)	Yes	-	-
ID_UNIT_TUJUAN	NUMBER(10,0)	Yes	-	-
WAKTU_KONTROL	DATE	Yes	-	-
WAKTU_DIPERIKSA	TIMESTAMP(6)	Yes	-	-
TERAPI	CLOB	Yes	-	-
				1 - 24

Gambar 3.14 Tabel RAWAT_JALAN

VISIT

Tabel VISIT memiliki jumlah data yang diinput sebanyak 500 *records* dengan 6 atribut dari tabel. Selengkapnya dapat dilihat pada Gambar 3.15.

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(24,0)	No	-	1
ID_PASIEN	NUMBER(10,0)	No	-	-
JENIS_KUNJUNGAN	NUMBER(2,0)	No	-	-
WAKTU	TIMESTAMP(6)	No	-	-
STATUS	NUMBER(2,0)	Yes	-	-
CARA_BOOKING	NUMBER(2,0)	Yes	-	-
				1 - 6

Gambar 3.15 Tabel VISIT

3.4.3 Mengidentifikasi Data pada Tabel

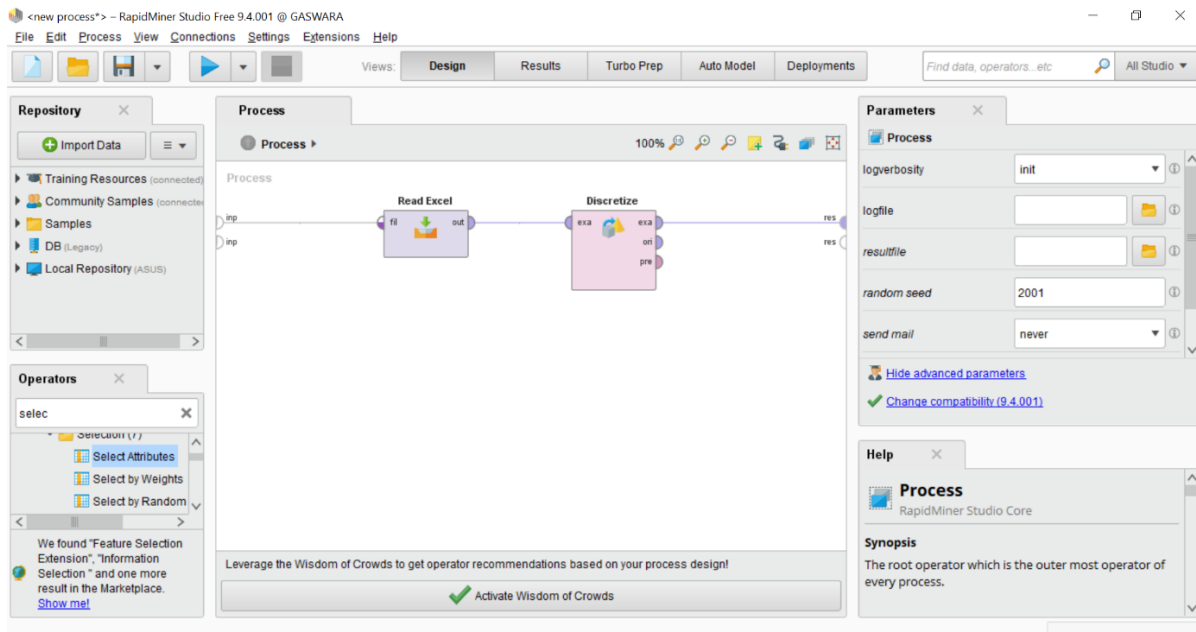
Pada tahap ini dilakukan identifikasi masalah yang ada pada tiap tabel. Proses identifikasi dilakukan dengan memperhatikan tipe data, atribut/kolom, dan metode *data transformation* apa saja yang dapat diterapkan pada tabel.

3.4.4 Mengimplementasikan Metode Data Transformation

Tahap ini merupakan penerapan metode *data transformation* pada tiap tabel. Penerapan tiap metode didasarkan pada hasil identifikasi data pada tabel. Metode yang akan diimplementasikan ada 6 (enam) seperti yang sudah dijelaskan sebelumnya. Metode tersebut akan diimplementasikan dengan bantuan *software* RapidMiner. Adapun cara penggunaan dalam mengimplementasikan tiap metode *data transformation* dengan bantuan RapidMiner sebagai berikut:

Binning

- Tujuan: Meningkatkan akurasi model prediksi dengan mengurangi *noise* atau non-linearitas. Metode ini mengubah variabel numerik menjadi variabel kategorikal. *Binning* memungkinkan mengidentifikasi *outlier*, *invalid variable*, dan *missing variable*. Pada proses *binning* biasanya banyak informasi yang hilang.
- Syarat Data: Atribut atau kolom memiliki tipe data *integer* atau *float*. Atribut yang digunakan merupakan sifat dari data itu sendiri/bukan *id foreign key* (misal. umur, gaji, dan harga).
- Keluaran: Variabel yang sudah dikategorikan berdasarkan *bin* yang dibuat.



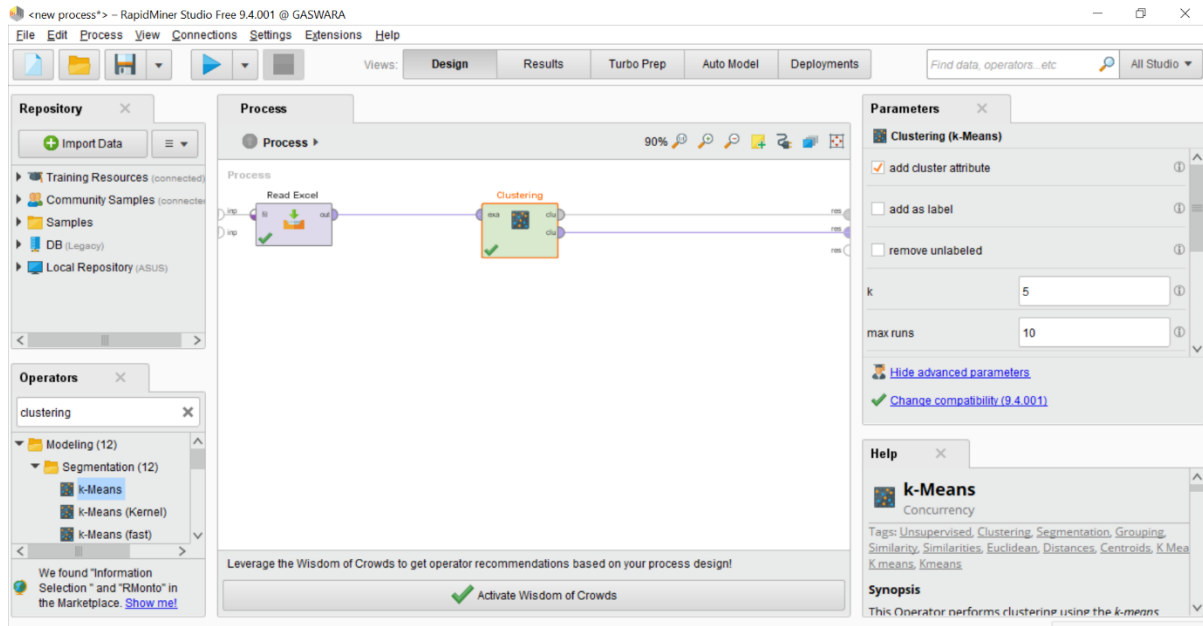
Gambar 3.16 Cara penggunaan *binning* dengan RapidMiner

Langkah-langkah proses penggunaan *binning* dengan RapidMiner dijelaskan sebagai berikut:

- Import file* xlsx ke RapidMiner dengan operator *Read Excel*.
- Drag and drop* operator *Discretize by Binning*.
- Pada parameter *discretize*, ubahlah *attribute filter type* menjadi *subset*. Setelah itu, pilih atribut yang ingin diterapkan dengan *binning*.
- Jalankan program.

Clustering

- Tujuan: Mengelompokkan data ke dalam kluster menurut tingkat kemiripannya dan menyingkirkan *outliers*. Metode ini mengubah variabel numerik menjadi variabel kategorikal.
- Syarat Data: Atribut atau kolom memiliki tipe data *integer* atau *float*. Atribut yang digunakan merupakan sifat dari data itu sendiri/bukan *id foreign key* (misal. umur, gaji, dan harga).
- Keluaran: Variabel yang sudah dikategorikan berdasarkan kluster yang dibuat.



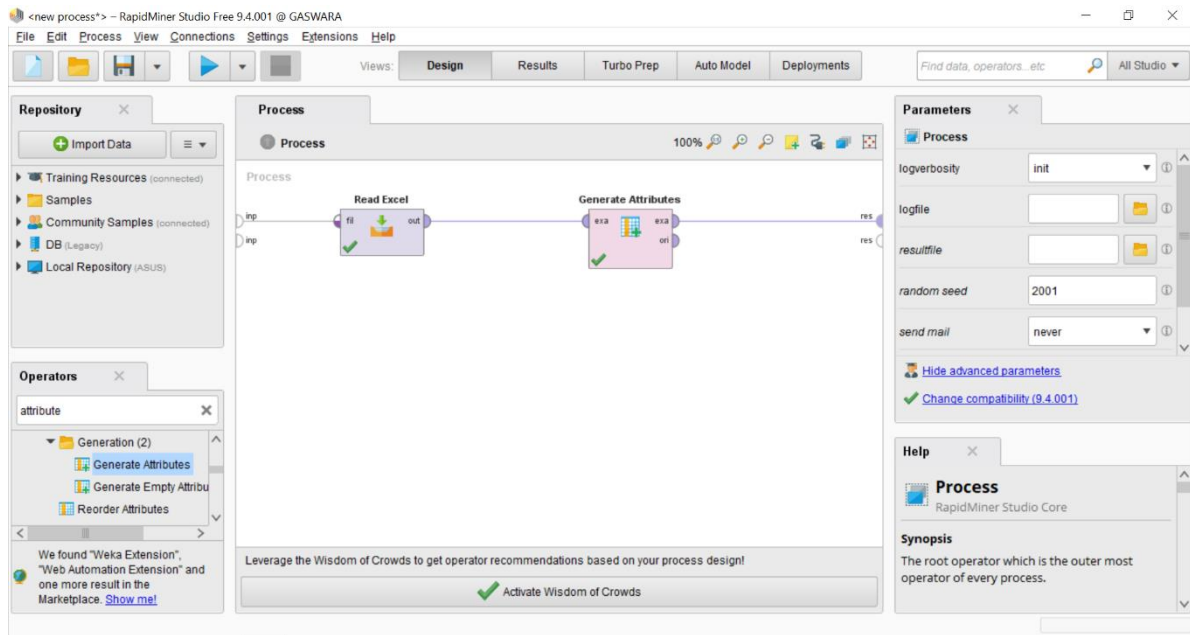
Gambar 3.17 Cara penggunaan *clustering* dengan RapidMiner

Langkah-langkah proses penggunaan *clustering* dengan RapidMiner dijelaskan sebagai berikut:

- a. *Import file* xlsx ke RapidMiner dengan operator *Read Excel*.
- b. *Drag and drop* operator *k-Means clustering*.
- c. Atur parameter sesuai dengan yang diinginkan. Seperti berapa jumlah klaster yang ingin dikelompokkan atau jumlah iterasi maksimal yang dijalankan.
- d. Jalankan program.

Attribute/Feature Construction

- a. Tujuan: Menambahkan atribut baru untuk meningkatkan ketelitian pada atribut lainnya. Misalnya, menambahkan atribut kategori luas yang diambil dari perhitungan atribut panjang dan atribut lebar.
- b. Syarat Data: Atribut atau kolom dapat dalam berbagai tipe data kecuali *id foreign key*.
- c. Keluaran: Atribut baru yang telah dibuat pada tabel disertai dengan nilainya



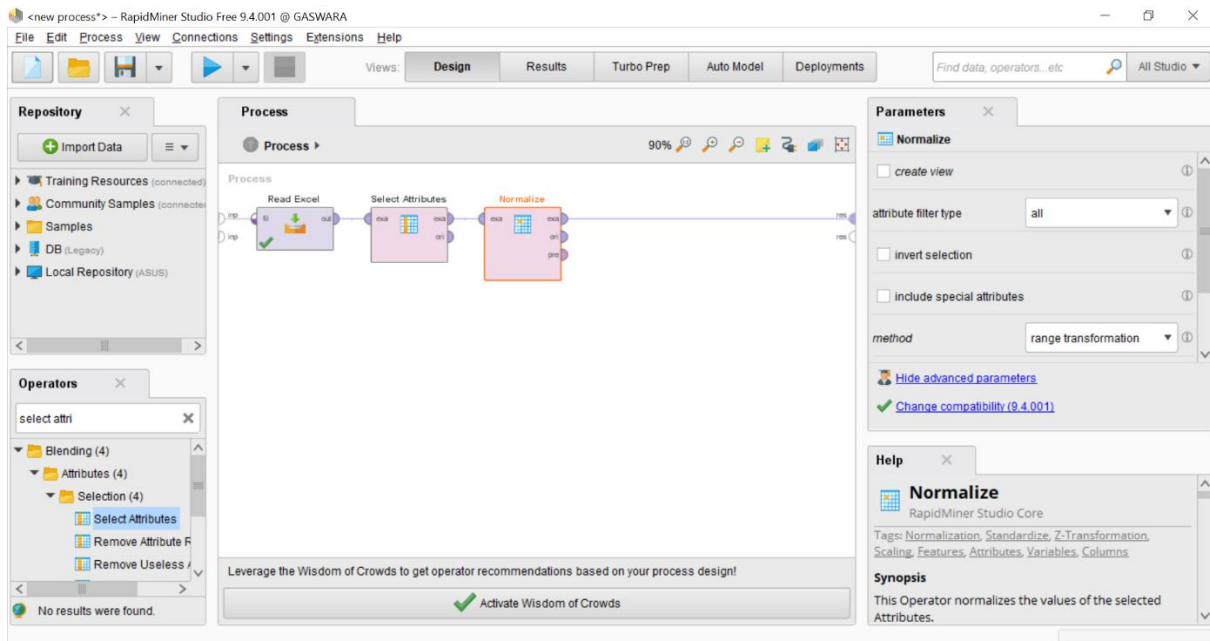
Gambar 3.18 Cara penggunaan *attribute/feature construction* dengan RapidMiner

Langkah-langkah proses penggunaan *attribute/feature construction* dengan RapidMiner dijelaskan sebagai berikut:

- a. *Import file* xlsx ke RapidMiner dengan operator *Read Excel*.
- b. *Drag and drop* operator *Generate Attributes*.
- c. Atur *function descriptions* pada parameter *generate attribute* dengan memberi nama atribut baru beserta fungsi untuk pemberian nilai.
- d. Jalankan program.

Min-Max Normalization

- a. Tujuan: Membuat setiap titik data memiliki skala yang sama sehingga setiap atribut sama pentingnya. Metode ini mentransformasi sebuah variabel numerik menjadi skala yang lebih kecil. *Min-max normalization* menjamin semua atribut memiliki skala yang tepat, tetapi tidak dengan mengatasi *outliers* dengan baik.
- b. Syarat Data: Atribut atau kolom memiliki tipe data *integer* atau *float*. Atribut yang digunakan merupakan sifat dari data itu sendiri/bukan *id foreign key* (misal. umur, gaji, dan harga) dan saling memiliki hubungan/memiliki keterkaitan.
- c. Keluaran: Variabel yang sudah ditransformasi dengan skala 0 sampai 1



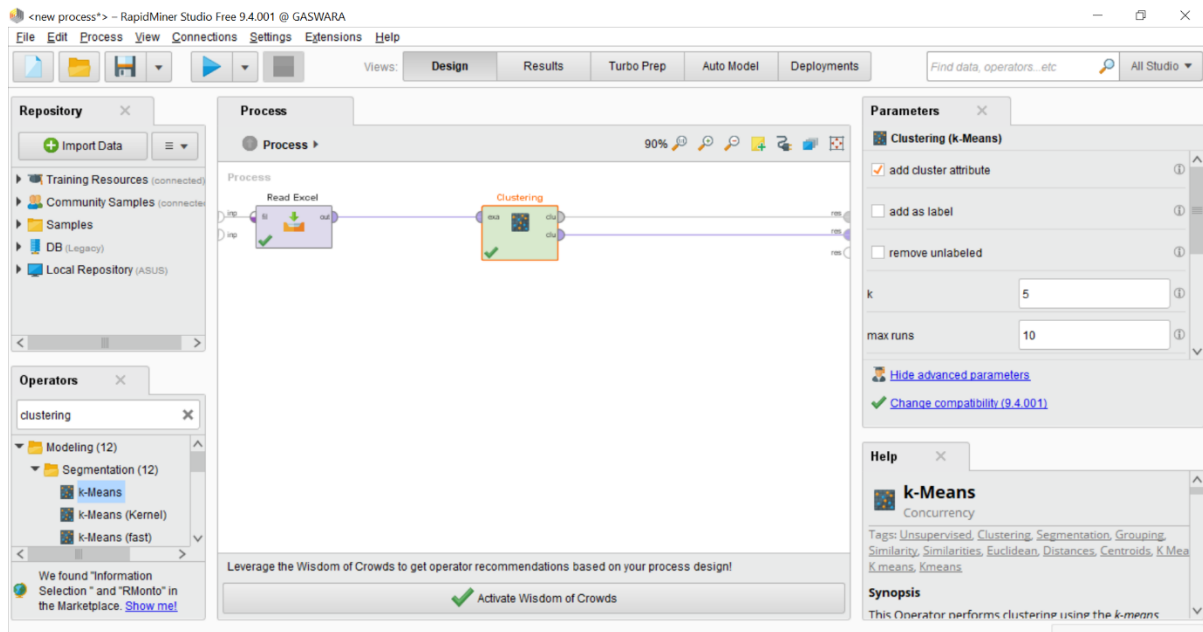
Gambar 3.19 Cara penggunaan *min-max normalization* dengan RapidMiner

Langkah-langkah proses penggunaan *min-max normalization* dengan RapidMiner dijelaskan sebagai berikut:

- a. *Import file* *xlsx* ke RapidMiner dengan operator *Read Excel*.
- b. *Drag and drop* operator *Select Attribute*.
- c. Pilih *attribute filter type* menjadi *subset* lalu pilih *attributes* sesuai keperluan.
- d. Setelah itu, *drag and drop* operator *normalize*.
- e. Atur *method* pada parameternya menjadi *range transformation*.
- f. Ubahlah nilai *min* dan nilai *max* sesuai dengan kebutuhan.
- g. Jalankan program.

Z-Score Normalization

- a. Tujuan: Membuat setiap titik data memiliki skala yang sama sehingga setiap atribut sama pentingnya dengan menjamin terhindar dari *outliers*. Metode ini mentransformasi sebuah variabel numerik menjadi skala yang lebih kecil.
- b. Syarat Data: Atribut atau kolom memiliki tipe data *integer* atau *float*. Atribut yang digunakan merupakan sifat dari data itu sendiri/bukan *id foreign key* (misal. umur, gaji, dan harga) dan saling memiliki hubungan/memiliki keterkaitan.
- c. Keluaran: Variabel yang sudah ditransformasi dengan skala variabel *X* antara -1,5 sampai 1,5 dan skala variabel *Y* antara -2 sampai 2



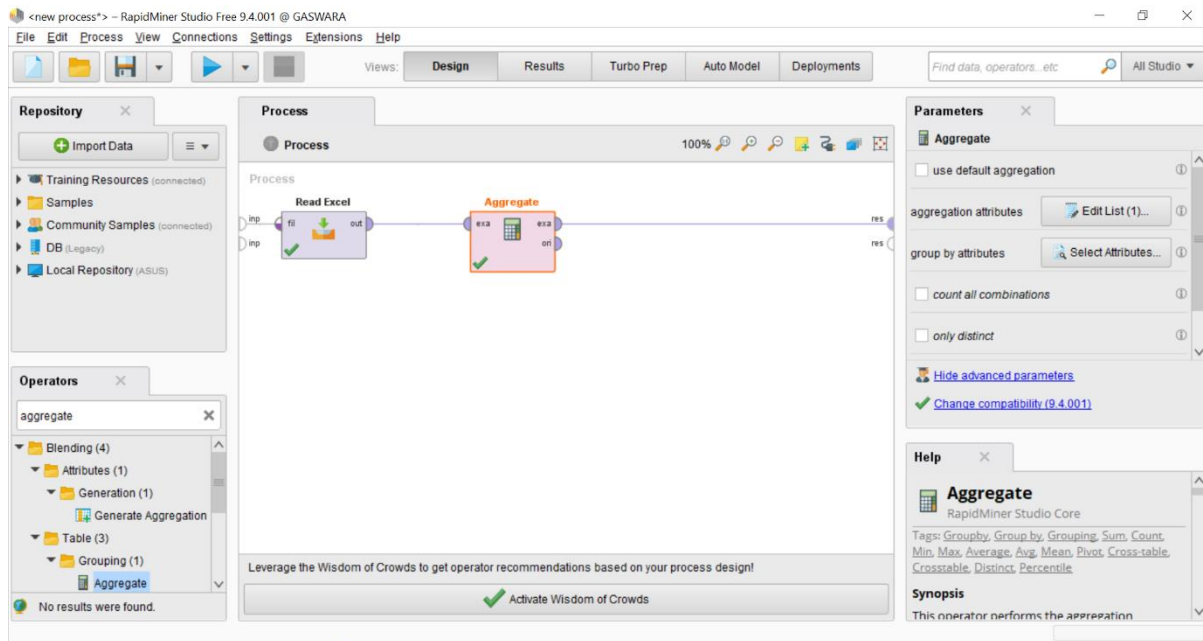
Gambar 3.20 Cara penggunaan *z-score normalization* dengan RapidMiner

Langkah-langkah proses penggunaan *z-score normalization* dengan RapidMiner dijelaskan sebagai berikut:

- a. *Import file* *xlsx* ke RapidMiner dengan operator *Read Excel*.
- b. *Drag and drop* operator *Select Attribute*.
- c. Pilih *attribute filter type* menjadi *subset* lalu pilih *attributes* sesuai keperluan.
- d. Setelah itu, *drag and drop* operator *normalize*.
- e. Atur *method* pada parameteranya menjadi *z transformation*.
- f. Jalankan program.

Aggregation

- a. Tujuan: Menyajikan data dalam format yang diinginkan untuk mencapai tujuan tertentu. Metode ini meringkas data dengan menjumlahkan, menghitung rata-rata, atau menghitung nilai yang sering muncul pada data.
- b. Syarat Data: Atribut atau kolom dapat dalam berbagai tipe data termasuk *id foreign key*.
- c. Keluaran: Hasil ringkasan data berupa variabel numerik



Gambar 3.21 Cara penggunaan *aggregation* dengan RapidMiner

Langkah-langkah proses penggunaan *aggregation* dengan RapidMiner dijelaskan sebagai berikut:

- a. *Import file* xlsx ke RapidMiner dengan operator *Read Excel*.
- b. *Drag and drop* operator *Aggregate*.
- c. Pada parameter *Aggregate*, pilihlah fungsi *aggregate* yang ingin digunakan pada *aggregate attributes*.
- d. Pilih juga pengelompokan pada hasilnya berdasarkan atribut dengan memilih pada *group by attributes*.
- e. Jalankan program.