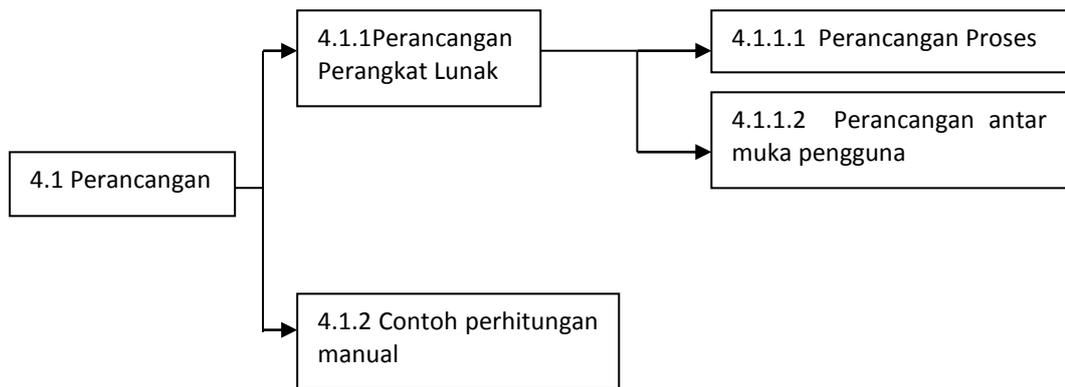


## BAB 4 PERANCANGAN

### 1.1. Perancangan

Bab ini membahas mengenai perancangan aplikasi Prediksi Nilai Tukar Rupiah dengan metode *Ensemble kNN*. Perancangan yang dilakukan meliputi perancangan perangkat lunak dan contoh perhitungan manual sistem. Proses perancangan perangkat lunak mempunyai dua tahap yaitu perancangan proses sistem Prediksi Nilai Tukar Rupiah menggunakan metode *Ensemble kNN* dan perancangan antarmuka pengguna. Tahap-tahap perancangan yang akan dilakukan dapat diilustrasikan pada Gambar 4.1.



Gambar 4.1 Diagram Blok Perancangan

#### 1.1.1. Perancangan Perangkat Lunak

Perancangan perangkat lunak terdiri dari dua tahap, yaitu perancangan proses dan perancangan antarmuka pengguna.

##### 1.1.1.1. Perancangan Proses

Perancangan proses merupakan perancangan tahap atau urutan sistem saat melakukan proses Prediksi menggunakan *Ensemble kNN*. Gambar 4.2 mengilustrasikan urutan proses yang akan dilakukan oleh sistem Prediksi Nilai Tukar Rupiah ini.



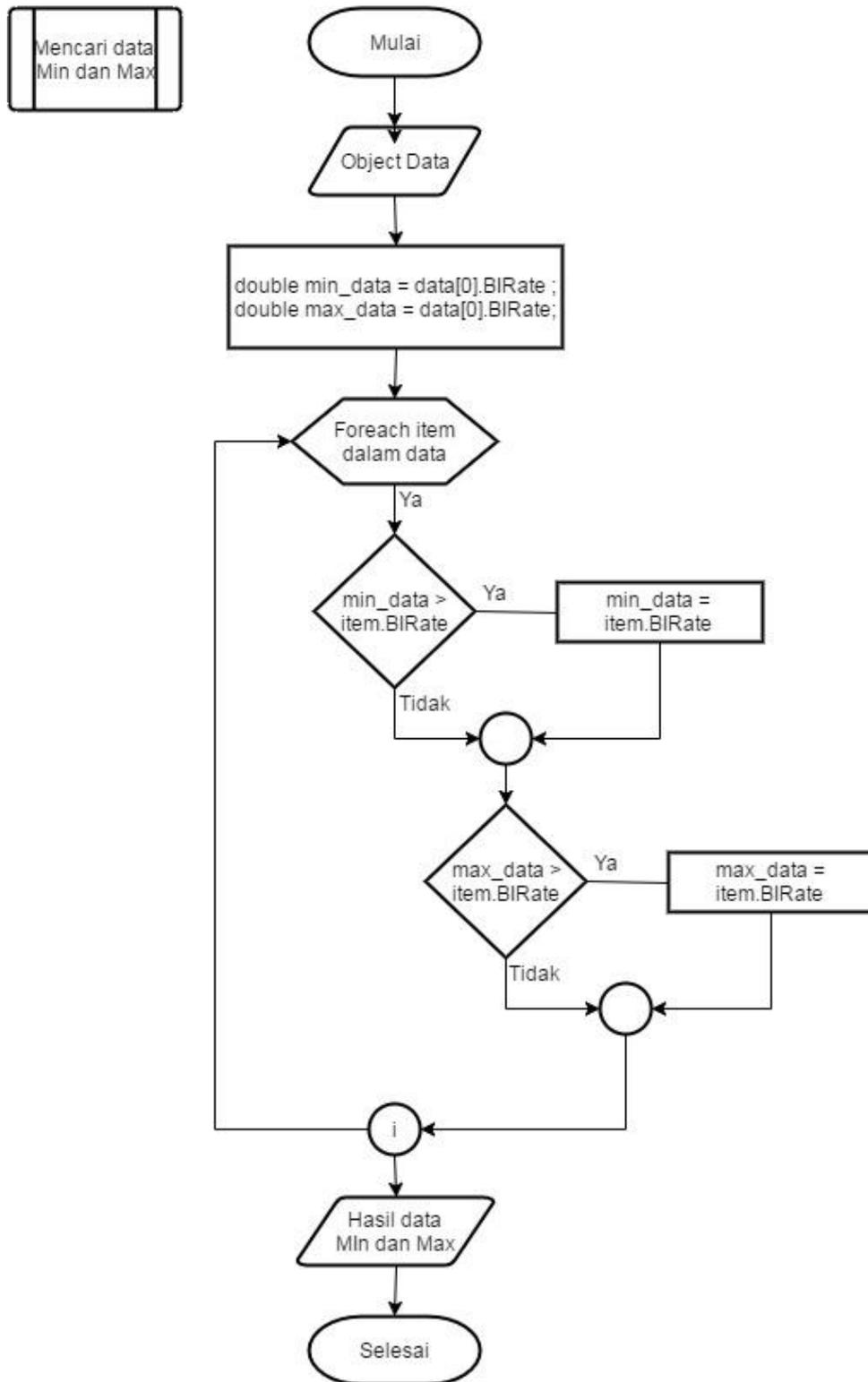
Gambar 4.2 Urutan Proses Sistem

Berdasarkan Gambar 4.2 pada tahap *preprocessing* data training dilakukan normalisasi data menggunakan persamaan MinMax (2-1) seperti yang dijelaskan pada diagram alir sistem pada Gambar 4.3.



**Gambar 4.3 Diagram Alir *Preprocessing* Data**

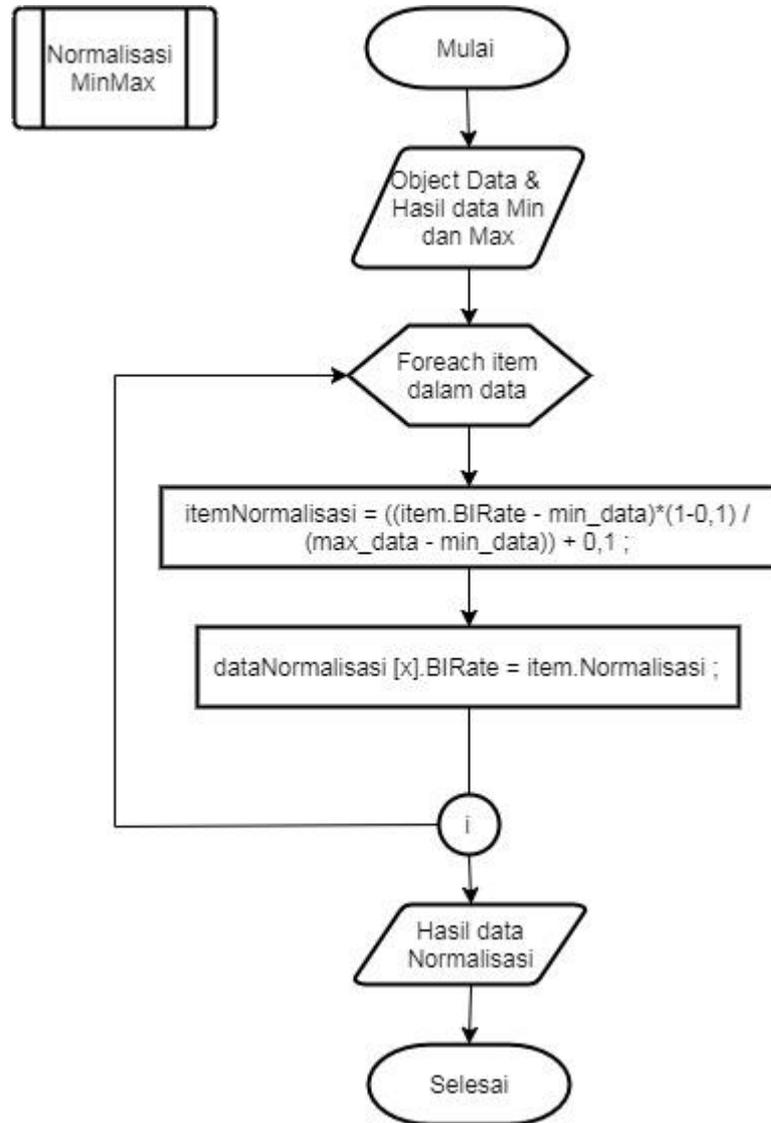
Proses pertama adalah melakukan normalisasi pada parameter BI Rate, inflasi, ekspor, dan impor. Terdapat masing-masing 36 data BI Rate, inflasi, ekspor, impor dan utang negara yang nantinya akan dinormalisasi dengan menggunakan persamaan *MinMax* (2-1). Proses normalisasi bertujuan untuk memberikan skala nilai atribut/parameter dari data pada range tertentu sehingga pada parameter BI Rate, inflasi, ekspor, impor, dan utang negara memiliki range yang nilainya seimbang. Sebelum mendapatkan hasil normalisasi *MinMax*, terlebih dahulu mencari nilai *minimum* dan *maksimum* dari tiap parameter. Salah satu diagram alir sistem untuk menentukan nilai *minimum* dan *maksimum* pada parameter BI rate dapat dilihat pada Gambar 4.4 .



**Gambar 4.4 Diagram Alir Proses Menentukan Nilai *Min* dan *Max* pada Data BI Rate**

Setelah nilai *minimum* dan *maksimum* pada parameter BI Rate sudah didapatkan, maka selanjutnya dilakukan proses normalisasi pada tiap parameter menggunakan persamaan *MinMax* (2-1). Salah satu diagram alir sistem untuk

menormalisasi data pada parameter BI Rate dapat ditunjukkan pada Gambar 4.5



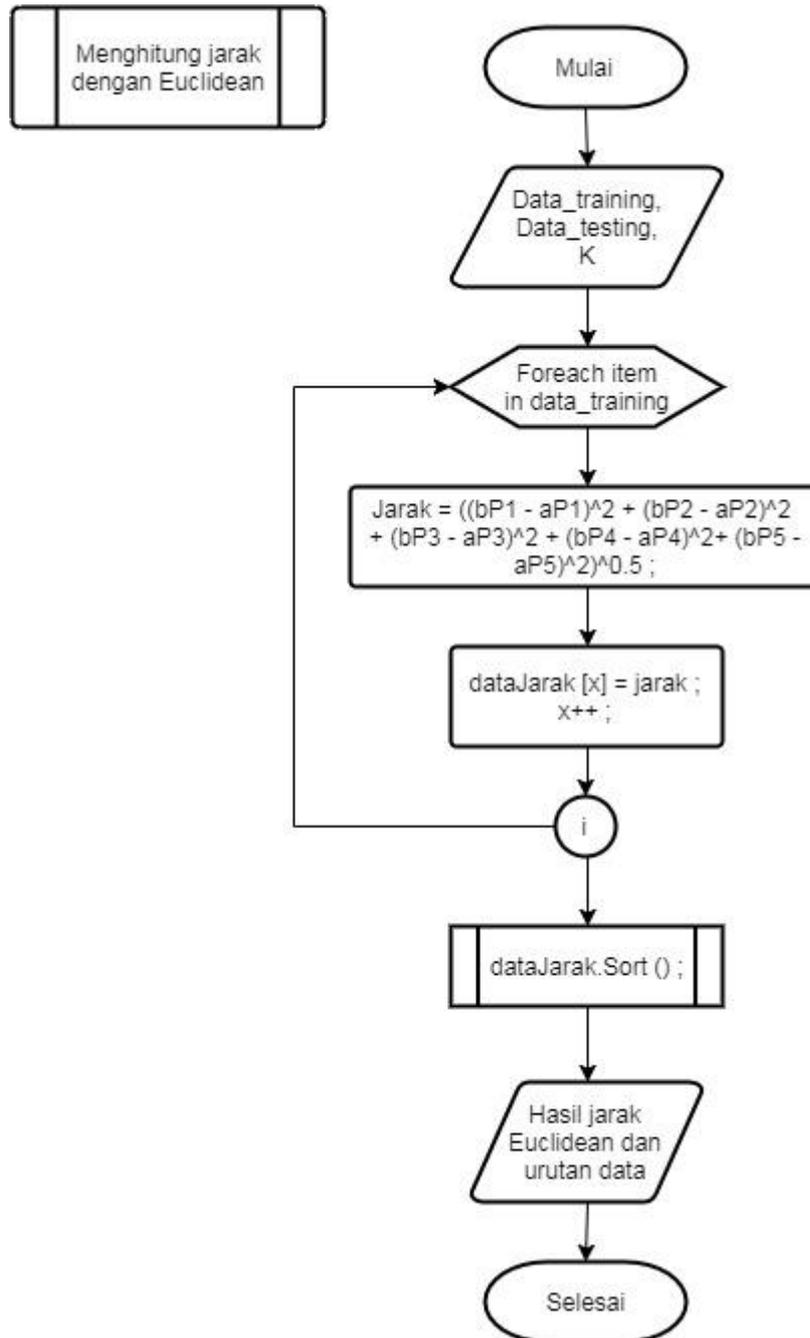
**Gambar 4.5 Diagram Alir Proses Normalisasi Data**

Proses mencari nilai *minimum* dan *maksimum*, serta proses normalisasi juga dilakukan pada data dengan parameter inflasi, ekspor, impor dan utang Negara. Selanjutnya menghitung prediksi nilai tukar rupiah dari Januari - Desember 2016 dengan metode *kNN*. Pada tahap ini dilakukan prediksi terhadap metode menggunakan jumlah tetangga terdekat (*k*) yang berbeda. Diagram alir sistem dalam menghitung prediksi nilai tukar dengan metode *kNN* dapat dilihat pada Gambar 4.6 .



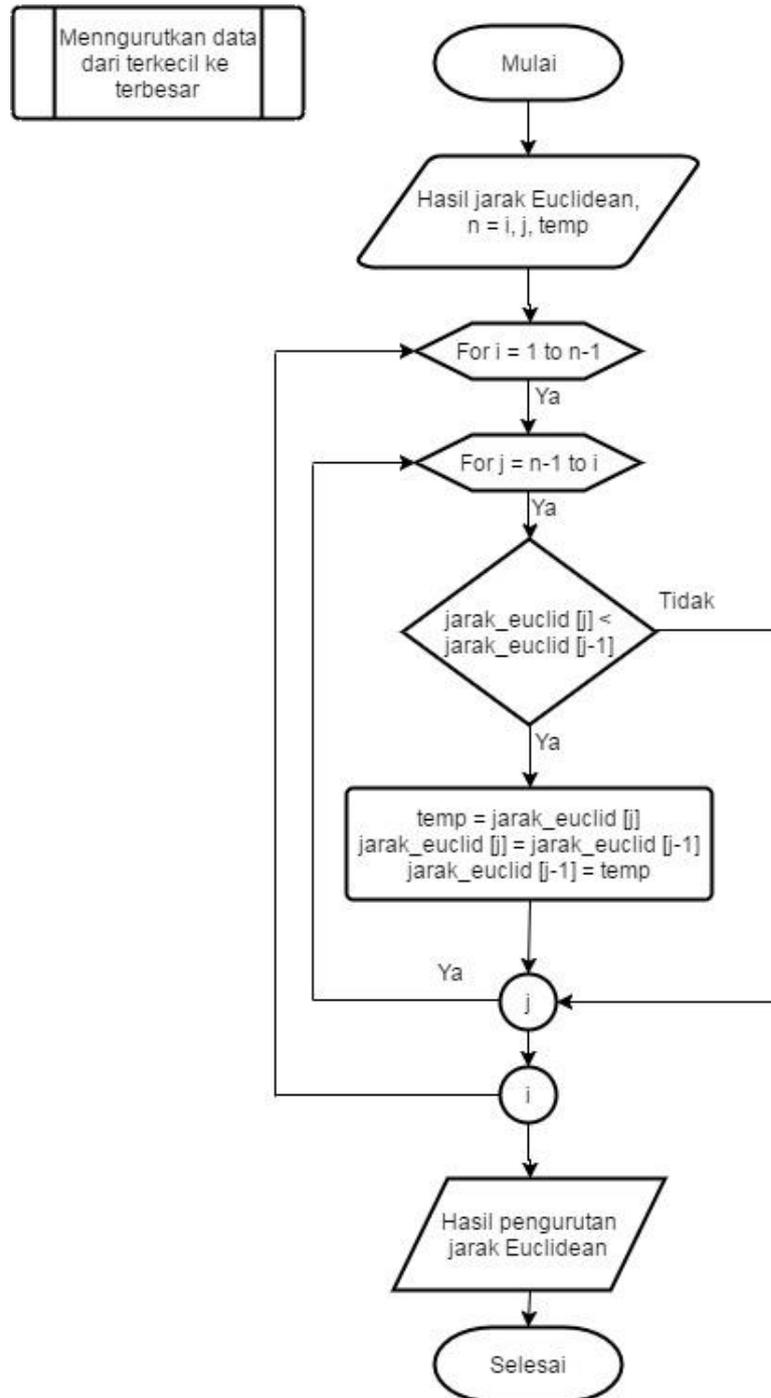
**Gambar 4.6 Diagram Alir Proses Menghitung Prediksi Menggunakan Metode *k*NN**

Sebelum mendapatkan hasil prediksi nilai tukar dengan metode *k*NN terlebih dahulu menghitung jarak *Euclidean* pada data *training* 1,2, ... ,*n* terhadap data *testing* berdasarkan parameter menggunakan persamaan (2-4). Diagram alir sistem untuk menghitung jarak *Euclidean* dapat dilihat pada Gambar 4.7 .



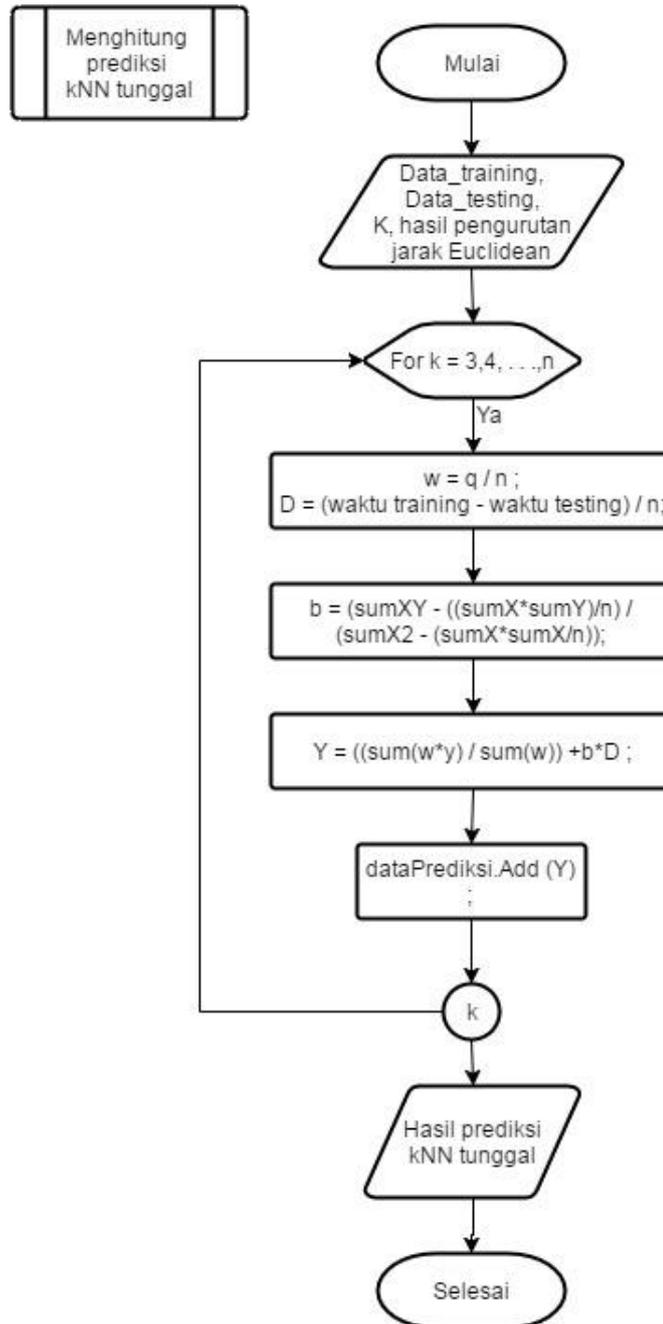
**Gambar 4.7 Diagram Alir Proses Menghitung Jarak Euclidean**

Dimana ( $aP1$ ) merupakan parameter BI rate pada data *training*, ( $bP1$ ) parameter BI rate pada data *testing*, ( $P2$ ) parameter inflasi, ( $P3$ ) parameter ekspor, ( $P4$ ) parameter impor dan ( $P5$ ) parameter utang negara. Selanjutnya mengurutkan jarak *Euclidean* tersebut dari jarak yang terkecil hingga terbesar, kemudian pilih  $k$  data *training* yang terdekat dengan data *testing* tersebut, yaitu data *training* yang memiliki jarak *Euclidean* terkecil dengan data *testing*. Diagram alir sistem untuk mengurutkan jarak *Euclidean* dari terkecil ke terbesar dapat dilihat pada Gambar 4.8.



**Gambar 4.8 Diagram Alir Proses Mengurutkan Data**

Selanjutnya menghitung prediksi nilai tukar menggunakan persamaan (2-7). Proses menghitung prediksi nilai tukar menggunakan persamaan (2-7) memerlukan perhitungan bobot, rata-rata selisih nomor urut antara data *testing* dengan data *training*, dan *slope* terlebih dahulu. Diagram alir sistem untuk menghitung prediksi nilai tukar menggunakan persamaan (2-7) dapat dilihat pada Gambar 4.9 .

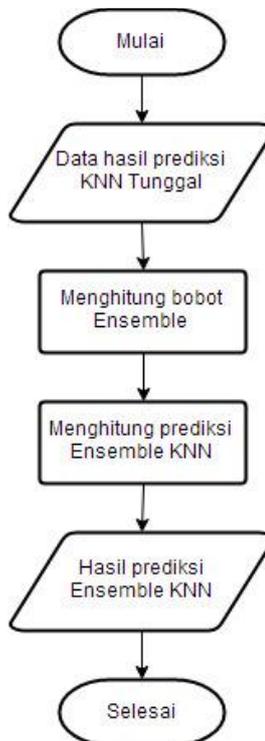


**Gambar 4.9 Diagram Alir Proses Perhitungan KNN**

Menghitung nilai pembobot untuk setiap data *training* yang terpilih sebagai tetangga terdekat, oleh karena data yang dianalisis merupakan data deret waktu yang memperhatikan urutan data maka data *training* yang nomor urutnya jauh dari data *testing* akan diboboti kecil, sedangkan data *training* yang dekat dengan data *testing* akan diboboti besar. Sehingga persamaan pembobot yang digunakan adalah  $w_j = \frac{q}{n}$  dengan  $w_j$  = pembobot *k*NN untuk tetangga ke-*n* yang terpilih, (*q*) = urutan waktu, dan (*n*) = banyaknya amatan pada data *training*. Nilai pembobot ini digunakan untuk memboboti peubah *output* yang terpilih menjadi tetangga terdekat.

Kemudian menghitung prediksi peubah *output* dengan metode *kNN*. Prediksi bukan hanya menggunakan rata-rata *k*-tetangga terdekat dari peubah *output* seperti halnya *kNN*. Agar nilai prediksi yang dihasilkan tetap mengikuti pola data yang memiliki *trend*, maka dalam penelitian ini prediksi menggunakan modifikasi metode *kNN* yaitu dengan menambahkan faktor koreksi *trend* dan perubahan waktu, yaitu menggunakan persamaan (2-7)

Nilai prediksi yang diperoleh pada langkah 1-6 dijadikan sebagai data *training* untuk memprediksi peubah *output* data *testing* berikutnya. Banyaknya data *training* yang digunakan dalam analisis selalu sama yaitu sebanyak 24 data, sehingga untuk prediksi peubah *output* yang kedua, data dengan nomor urut pertama tidak diikutsertakan dalam analisis. Untuk memprediksi data yang ketiga, data dengan nomor urut pertama dan kedua tidak diikutsertakan. Hal ini dilakukan juga pada prediksi selanjutnya.



**Gambar 4.10 Diagram Alir Proses Perhitungan Ensemble KNN**

Berdasarkan Gambar 4.10, langkah-langkah menghitung prediksi nilai tukar rupiah dengan metode *ensemble KNN* sebagai berikut:

1. Menghitung prediksi menggunakan metode *kNN* dengan jumlah tetangga terdekat (*k*) berbeda-beda yaitu 3, 4, 6, 9, 10, 12, 15 dan 24. Nilai-nilai *k* ini digunakan untuk melihat bagaimana pengaruh nilai *k* terhadap hasil prediksi.
2. Menggabungkan kedelapan hasil prediksi tersebut menggunakan persamaan (2-9), pembobot *ensemble* ( $w_h$ ) merupakan korelasi antara data kurs dengan prediksi metode *kNN* ke-*h* ( $r_h$ ) menggunakan persamaan (2-10). Korelasi menggambarkan besarnya hubungan antara data dengan

prediksi metode *kNN*. Prediksi yang baik akan mengikuti pola data yang sebenarnya, jika prediksi mengikuti pola *trend* yang searah dengan pola data sebenarnya maka korelasi yang dihasilkan besar, sehingga pembobot  $w_h$  juga besar.

### 1.1.1.2. Perancangan Antarmuka Pengguna

Perancangan antarmuka bertujuan untuk mewakili keadaan sebenarnya dari sistem yang akan dibangun. Sistem Prediksi Nilai Tukar Rupiah dengan metode *Ensemble kNN* terdiri dari tiga halaman yaitu halaman Normalisasi Data, halaman *kNN* dan *Ensemble kNN*, dan halaman pengujian MAPE, MAE, RMSEP.

#### 1. Perancangan antaruka pengguna halaman Normalisasi Data



**Gambar 4.11 Perancangan Antarmuka Halaman Normalisasi Data**

Halaman Normalisasi menampilkan proses normalisasi yang dilakukan pada aplikasi Prediksi Nilai Tukar Rupiah terhadap Dollar Amerika dengan metode *Ensemble kNN*. Ilustrasi halaman Normalisasi Data dapat dilihat pada Gambar 4.11. Halaman ini terdiri dari beberapa bagian antara lain :

- Tombol ComboBox untuk memilih Bulan yang akan diprediksi. Serta untuk proses menentukan data *training* dengan rentang waktu 2 tahun sebelumnya.
- Table yang akan menampilkan data mentah atau data awal sebelum di normalisasi.
- Tombol Proses Normalisasi untuk menormalisasi data mentah/awal.
- Table yang akan menampilkan data yang telah di normalisasi.

## 2. Perancangan antarmuka pengguna halaman *kNN* dan *Ensemble kNN*

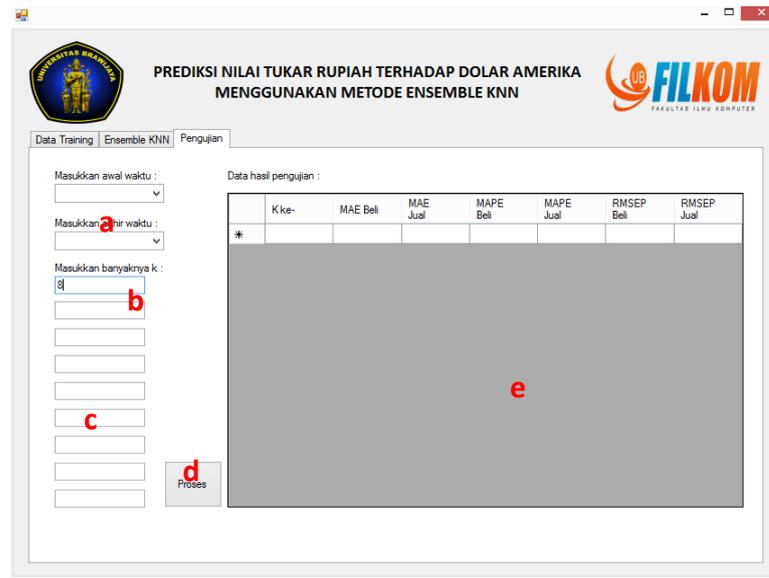
K ke-	Prediksi Kurs Jual	Prediksi Kurs Beli
*		

**Gambar 4.12 Perancangan Antarmuka Halaman *Ensemble kNN***

Halaman *kNN* tunggal dan *Ensemble kNN* menampilkan proses *kNN* serta *Ensemble kNN* dari data yang telah dinormalisasi. Ilustrasi halaman *kNN* dan *Ensemble kNN* dapat dilihat pada Gambar 4.12. Halaman ini terdiri dari beberapa bagian antara lain :

- TextBox untuk menginputkan berapa jumlah  $K$  (kostanta) yang akan di tentukan.
- TextBox untuk menginputkan nilai  $K$  (kostanta) yang akan di tentukan.
- Label untuk menampilkan nilai dari *Slope* pada parameter Kurs Beli.
- Label untuk menampilkan nilai dari *Slope* pada parameter Kurs Jual.
- Tombol untuk untuk proses *kNN* serta *Ensemble kNN*.
- Table untuk menampilkan hasil dari proses *kNN* serta *Ensemble kNN*.

### 3. Perancangan antarmuka pengguna halaman pengujian MAPE, MAE, RMSEP



**Gambar 4.13 Perancangan Antarmuka Halaman Pengujian**

Halaman pengujian MAPE, MAE, RMSEP menampilkan hasil proses dari pengujian MAPE, MAE, dan RMSEP dari data *kNN* dan *Ensemble kNN*. Ilustrasi halaman *kNN* tunggal dan *Ensemble kNN* dapat dilihat pada Gambar 4.13. Halaman ini terdiri dari beberapa bagian antara lain :

- TombolComboBox untuk memilih atau menentukan awal waktu dan TombolComboBox untuk memilih atau menentukan akhir waktu
- TextBox untuk menginputkan berapa jumlah K (kostanta) yang akan di tentukan.
- TextBox untuk menginputkan nilai K (kostanta) yang akan di tentukan.
- Tombol untuk melakukan proses pengujian MAPE, MAE , dan RMSEP.
- Table untuk menampilkan hasil dari proses pengujian MAPE, MAE , dan RMSEP terhadap data *kNN* dan *Ensemble kNN*.

#### 4.1.2. Perhitungan Manual

Perhitungan manual berfungsi untuk memberikan gambaran umum perancangan sistem yang akan dibangun. Sistem Prediksi Nilai Tukar Rupiah terhadap Dollar Amerika pada penelitian ini menggunakan metode *Ensemble kNN*. Data *testing* terdiri dari 6 parameter yaitu tanggal, data BI Rate, data inflasi, ekspor, impor dan utang negara. Sedangkan data *training* terdiri dari 8 parameter yaitu tanggal, BI Rate, inflasi, ekspor, impor, utang negar, nilai jual rupiah( kurs jual), dan nilai beli rupiah (kurs beli). Data *testing* dan *training* berupa data sebenarnya dari bulan Januari 2014 sampai Desember 2016.

Pada manualisasi ini, proses yang dilakukan pertamakali yaitu menormalisasi data training dan testing pada parameter ekspor dan impor. Kedua, menghitung jarak euclidian dan dilakukan sorting dari nilai terkecil ke terbesar. Ketiga, menghitung bobot(weight) pada tiap K (kostanta). Keempat, menghitung *kNN*.

Kelima, menghitung *Ensemble kNN* dan selanjutnya dilakukan pengujian dengan MAPE, MAE, RMSEP.

#### 4.1.2.1. Tahap Normalisasi Data Training dan Testing

Tahap pertama yang akan dilakukan pada proses Prediksi Nilai Tukar Rupiah yaitu menormalisasi data *training* dan data *testing*. Proses normalisasi data *training* dan data *testing* dilakukan dengan mencari nilai tertinggi(max) dan terendah(min). Nilai tertinggi dan terendah akan di traformasi menggunakan persamaan (2-1).

Contoh data yang terdiri dari 36 data *training* dan 12 testing yang akan di normalisasikan pada manualisasi ini tampak pada Tabel 4.1

**Tabel 4.1 Data Awal**

Inflansi
0.0457
0.0531
0.059
0.0557
0.0547
0.059
0.0861
0.0879
0.084
0.0832
0.0837
0.838
0.0822
0.0775
0.0732
0.0725
0.0732
0.067
0.0453
0.0399
0.0453
0.0483
0.0623
0.0836
0.0696
0.0629
0.0638
0.0679
0.0715

0.0726
0.0726
0.0718
0.0683
0.0625
0.0489
0.0335
0.0414
0.0442
0.0445
0.036
0.0333
0.0345
0.0321
0.0279
0.0307
0.0331
0.0358
0.0302

Salah satu contoh perhitungan normalisasi dari data *training* pada Table 4.1 berdasarkan persamaan (2-1) adalah sebagai berikut :

$$newdata = \frac{(0.0457 - 0.0279) * (1 - 0.1)}{(0.838 - 0.0279)} + 0.1 = 0.11977$$

Jadi, nilai normalisasi dari data *training* dengan parameter inflasi pada bulan Januari 2014 yang sebelumnya (0.0457) ditransformasi nilainya menjadi (0.11977).

#### 4.1.2.2. Tahap Menghitung Jarak Euclidian

Tahap kedua yang akan dilakukan yaitu menghitung jarak *Euclidian* serta mengurutkan dari terkecil ke terbesar berdasarkan nilai jarak. Proses menghitung jarak Euclidian data *training* terhadap data *testing* dilakukan dengan persamaan(2-4). Contoh data training dan testing setelah di normalisasi bisa lihat pada Tabel 4.2

**Tabel 4.2 Contoh Data *Training* dan *Testing***

Data testing Januari					
January-16	0.85	0.114998	0.352872	0.448542	0.79704
Data Training					
January-14	0.925	0.160326	0.729005	0.913679	0.338184
February-14	0.925	0.155104	0.685713	0.671166	0.364435

Salah satu contoh perhitungan jarak *Euclidian* dari data *training* dan *testing* pada Tabel 4.2 berdasarkan persamaan (2-4) adalah sebagai berikut :

$$d_{x_{train,1}x_{test,1}} = \sqrt{\left( (0.85 - 0.925)^2 + (0.114998 - 0.160326)^2 + (0.352872 - 0.729005)^2 \right) + (0.448542 - 0.913679)^2 + (0.79704 - 0.338184)^2}$$

$$= 0.758984 = 0.76$$

Jadi, nilai dari jarak Euclidian data *training* bulan Januari 2014 terhadap data *testing* Bulan Januari 2016 sebesar 0.76. Selanjutnya dilakukan perhitungan sebanyak data *training* maka di dapatkan hasil seperti Tabel 4.3 yang kemudian di urutkan dari yang terkecil ke terbesar berdasarkan nilai jarak *Euclidian*.

**Tabel 4.3 Hasil Urutan Jarak Euclidian**

nomer urut	Euclidian	nomer urut	Euclidian
1	0.758984	14	0.189058
2	0.595588	19	0.196908
3	0.580894	23	0.213464
4	0.714422	22	0.284328
5	0.538417	17	0.313032
6	0.780098	24	0.334602
7	0.404568	13	0.347009
8	0.516896	16	0.399648
9	0.726457	7	0.404568
10	0.716274	11	0.465959
11	0.465959	15	0.507787
12	0.657678	8	0.516896
13	0.347009	20	0.532063
14	0.189058	5	0.538417
15	0.507787	18	0.564259
16	0.399648	3	0.580894
17	0.313032	21	0.582662
18	0.564259	2	0.595588
19	0.196908	12	0.657678
20	0.532063	4	0.714422
21	0.582662	10	0.716274
22	0.284328	9	0.726457
23	0.213464	1	0.758984
24	0.334602	6	0.780098

#### 4.1.2.3. Tahap Menghitung Bobot

Tahap ketiga yang akan dilakukan yaitu menghitung nilai pembobot untuk setiap data *training* yang terpilih sebagai tetangga terdekat, oleh karena data yang dianalisis merupakan data deret waktu yang memperhatikan urutan data. Data *training* yang nomor urutnya jauh dari data *testing* akan diboboti kecil, sedangkan data *training* yang dekat dengan data *testing* akan diboboti besar, sehingga formula pembobot yang digunakan adalah  $w_j = \frac{q}{n}$  dengan  $w_j =$

pembobot  $kNN$  untuk tetangga ke- $j$  yang terpilih,  $q$  = jarak urutan waktu, dan  $n$  = banyaknya amatan pada data *training*. Nilai pembobot ini digunakan untuk memboboti peubah *output* yang terpilih menjadi tetangga terdekat.

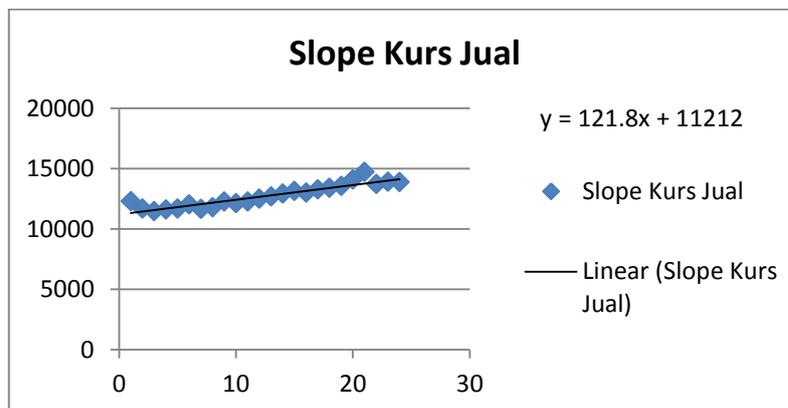
Jarak urutan waktu merupakan jarak antara tanggal sekarang dengan tanggal selanjutnya. Misal jarak bulan Januari 2015 terhadap bulan Maret 2015 yaitu 2. Salah satu contoh perhitungan untuk menentukan nilai bobot setiap data *training* berdasarkan Tabel 4.3 terhadap data *testing* berdasarkan Tabel 4.2 adalah sebagai berikut :

$$w_{14,1} = \frac{(\text{jarak urutan waktu dari Februari 2015 sampai Januari 2016})}{\text{banyak data training}} = \frac{11}{24} = 0.48533$$

#### 4.1.2.4. Tahap Menghitung $kNN$

Tahap keempat yang akan dilakukan yaitu menghitung prediksi peubah *output* dengan metode  $kNN$ . Prediksi bukan hanya menggunakan rata-rata  $k$ -tetangga terdekat dari peubah *output* seperti halnya  $kNN$ . Agar nilai prediksi yang dihasilkan tetap mengikuti pola data yang memiliki *trend*, maka dalam penelitian ini prediksi menggunakan modifikasi metode  $kNN$  yaitu dengan menambahkan faktor koreksi *trend* dan perubahan waktu berdasarkan persamaan (2-7).

Tetapi sebelum menghitung  $kNN$ , terlebih dahulu mencari nilai *slope* yang merupakan koefisien regresi antara  $T$  (waktu) terhadap  $Y$ (peubah output) dan menghitung rata-rata selisih antara nomor urut data *testing* dengan data *training* yang terpilih menjadi  $k$  tetangga terdekat. Pada penelitian ini ( $T$ ) merupakan deret waktu dari Bulan ke-  $i$  sampai  $n$  dan peubah output berupa nilai tukar rupiah (kurs). Maka *slope* bisa di ilustrasikan pada Gambar 4.14 berikut :



**Gambar 4.14 Slope dari Kurs Jual**

Dari Gambar 4.14 dapat diketahui nilai *slope* untuk Kurs Jual sebesar 121.8. Sedangkan untuk menghitung nilai rata-rata selisih deret waktu ( $D$ ) berdasarkan banyak  $k$  tetangga. Contoh, di ketahui dengan nilai  $k=3$  maka berdasarkan Tabel 4.3 diperoleh data ke 14 (Februari 2015), data ke 19 (Juli 2015), dan data ke 23 (November 2015) dari data *training*, sehingga dapat ditulis berdasarkan persamaan berikut :

$$D_{k=3} = \frac{(\text{jarak Feb15 sampai Jan16})+(\text{jarak Jul15 sampai Jan16})+(\text{jarNov15 samapi Jan16})}{\text{banyak } k \text{ tetangga}}$$

$$= \frac{11+6+2}{3} = 6.333$$

Setelah mendapatkan nilai bobot, nilai slope dan nilai rata-rata dari selisih waktu berdasarkan  $k$  tetangga maka dapat menghitung regresi kNN berdasarkan persamaan (2-7) adalah sebagai berikut :

$$y_1 = \frac{(0.4583 * 12927) + (0.25 * 13548) + (0.083 * 13909)}{(0.4583 + 0.25 + 0.083)} + (121.8 * 6.33) = 13997.88$$

Dari perhitungan di atas menggunakan persamaan (2-7), di peroleh hasil 13997.88 untuk  $k=3$  dengan data kurs jual.

#### 4.1.2.5. Tahap Menghitung Ensemble KNN

Tahap kelima yang akan dilakukan yaitu menggabungkan dari kedelapan hasil prediksi kNN tersebut berdasarkan persamaan (2-9) dengan  $s$  merupakan banyaknya metode kNN yang digunakan, pembobot *ensemble* ( $w_h$ ) merupakan korelasi antara data dengan prediksi metode kNN ke- $h$  ( $r_h$ ) menggunakan persamaan (2-10).

Contoh data hasil dari perhitungan kNN untuk *kurs* jual bisa dilihat pada Tabel 4.4 :

**Tabel 4.4 Contoh Hasil Prediksi Kurs Jual kNN**

K ke-	KNN Jual
k=3	13997.88
k=4	13961.9
k=6	13935.89
k=9	13682.48
k=10	13678.58
k=12	13691.88
k=15	13695.42
k=24	13789.94

$$W_{k=3} = \frac{13997.88}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.126753$$

$$W_{k=4} = \frac{13961.9}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.126428$$

$$W_{k=6} = \frac{13935.89}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.126192$$

$$W_{k=9} = \frac{13682.48}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.123897$$

$$W_{k=10} = \frac{13678.58}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.123862$$

$$W_{k=12} = \frac{13691.88}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.123983$$

$$W_{k=15} = \frac{13695.42}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.124015$$

$$W_{k=24} = \frac{13789.94}{13997.88+13961.9+13935.89+13682.48+13678.58+13691.88+13695.42+13789.94} = 0.12487$$

Menghitung *Ensemble kNN* dengan persamaan (2-9) sebagai berikut :

$$\begin{aligned} \mu &= \frac{(13997.88 * 0.126753) + (13961.9 * 0.126428) + (13935.89 * 0.126192) + (13682.48 * 0.123897) + \\ &= \frac{(13678.58 * 0.123862) + (13691.88 * 0.123983) + (13695.42 * 0.124015) + (13789.94 * 0.12487)}{0.126753 + 0.126428 + 0.126192 + 0.123897 + 0.123862 + 0.123983 + 0.124015 + 0.12487} \\ &= 13805.47 \end{aligned}$$

Maka dari menggunakan persamaan (2-9) diperoleh hasil prediksi menggunakan metode *Ensemble kNN* yaitu 13805.47 untuk kurs jual pada bulan Januari 2016.

#### 4.1.2.6. Menghitung MAE/MAD, MAPE, RMSEP

Tahap terakhir mengitung dan evaluasi terhadap hasil prediksi akhir dengan nilai tukar rupiah pada data *testing* berdasarkan nilai MAPE, MAE dan RMSEP. Sebagai salah satu contoh data hasil dari prediksi *kNN* dan *Ensemble kNN* bisa dilihat pada Tabel 4.5

**Tabel 4.5 Tabel Hasil Prediksi Kurs Jual Tahun 2016 pada K=3**

bulan	Aktual	k=3
Januari 2016	13915	13997.87
Februari 2016	13462	14051.75
Maret 2016	13342	13741.54
Apr-16	13270	13893.7
Mei 2016	13683	13591.8
Juni 2016	13246	13560.23
Juli 2016	13159	13999.53
Agustus 2016	13367	13645.1
Sep-16	13063	13488.03
Oktober 2016	13116	13344.03
Nov-16	13631	13271.03

Desember 2016	13503	13366.13
------------------	-------	----------

Dari tabel 4.5, maka dapat dihitung untuk nilai MAE, MAPE, RMSEP dengan k=3 menggunakan persamaan (2-11) untuk MAE/MAD, (2-12) untuk MAPE, dan (2-13) untuk RMSEP

$$MAE = \frac{|13915-13997.87|+|13462-14051.75|+|13342-13741.54|+|13270-13893.7|+|13683-13591.8|+|13246.-13560.233|+|13159-13999.533|+|13367-13645.1|+|13063-13488.03|+|13116-13344.02857|+|13631.-13271.033|+|13503-13366.133|}{12} = 364.15$$

$$MAPE = \frac{\left(\frac{82.87}{13915}\right) + \left(\frac{589.75}{13462}\right) + \left(\frac{399.54}{13342}\right) + \left(\frac{623.7}{13270}\right) + \left(\frac{91.2}{13683}\right) + \left(\frac{314.23}{13246}\right) + \left(\frac{840.53}{13159}\right) + \left(\frac{278.1}{13367}\right) + \left(\frac{425.03}{13063}\right) + \left(\frac{228.02}{13116}\right) + \left(\frac{359.96}{13631}\right) + \left(\frac{136.86}{13503}\right)}{12} * 100\% = 2.735\%$$

$$RMSEP = \sqrt{\frac{(82.87)^2 + (589.75)^2 + (399.54)^2 + (623.7)^2 + (91.2)^2 + (314.23)^2 + (840.53)^2 + (278.1)^2 + (425.03)^2 + (228.02)^2 + (359.96)^2 + (136.86)^2}{12}} = 425075$$