

Implementasi Metode *Support Vector Regression (SVR)* Dalam Peramalan Penjualan Roti (Studi Kasus: Harum Bakery)

Noval Dini Maulana¹, Budi Darma Setiawan², Candra Dewi³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹novaldinimaulana@gmail.com, ²s.budidarma@ub.ac.id, ³dewi_candra@ub.ac.id

Abstrak

Roti merupakan salah satu jenis makanan yang digemari oleh Masyarakat Indonesia. Salah satu bukti pentingnya roti bagi masyarakat Indonesia adalah impor tepung terigu yang terus meningkat. Salah satu perusahaan pembuat roti yang saat ini sedang berkembang adalah Harum Bakery. Kendala yang sering dihadapi Harum Bakery adalah sistem peramalan permintaan pelanggan yang masih manual dan terkesan mengira-ngira. Proses peramalan tersebut sangat jelas berpengaruh besar terhadap proses penjualan. Dengan adanya peramalan penjualan roti, diharapkan dapat membantu toko roti Harum Bakery dalam mempersiapkan bahan baku dan segala sesuatu yang diperlukan untuk pembuatan roti. *Support Vector Regression (SVR)* adalah salah satu metode yang bisa digunakan dalam melakukan peramalan. Data yang digunakan adalah data penjualan roti manis, cake dan tawar dengan tipe data time series dan menggunakan 4 fitur. Pada penelitian ini metode SVR yang digunakan untuk meramal hasil penjualan menghasilkan nilai evaluasi RMSE untuk roti manis sebesar 0,00176, roti cake sebesar 0,00019, dan roti tawar sebesar 0,00010.

Kata kunci: Roti, *Support Vector Regression*, Peramalan, *time series*

Abstract

Bread is one of the favorite foods of the Indonesian people, the proof is the increasing import of wheat flour. One of the bakery companies that is currently developing is Harum Bakery. Constraints that are often faced by Harum Bakery are customer demand forecasting systems that are still manual and seem to be guessing. The forecasting process give a big impact on the sales process. With the forecasting of bread sales, it is hoped that Harum Bakery can help bakeries in preparing raw materials and everything needed for bread making. Support Vector Regression (SVR) is one method that can be used in forecasting. The data used is data on sales of sweet bread, cake and white bread with time series data types and uses 4 features. In this study the SVR method used to predict the results of the sale resulted in an evaluation value of RMSE for sweet breads is 0.00176, bread cake is 0.00019, and large breads is 0.00010.

Keyword: *Support Vector Regression, Forecasting, Time Series*

1. PENDAHULUAN

Roti merupakan salah satu jenis makanan yang digemari oleh Masyarakat Indonesia. Salah satu bukti pentingnya roti bagi masyarakat Indonesia adalah impor tepung terigu yang terus meningkat. Salah satu perusahaan pembuat roti yang saat ini sedang berkembang adalah Harum Bakery.

Kendala yang sering dihadapi Harum Bakery adalah sistem peramalan permintaan pelanggan yang masih manual dan terkesan mengira-ngira. Proses peramalan tersebut sangat

jelas berpengaruh besar terhadap proses penjualan, dimana perusahaan harus cepat dalam menyiapkan bahan baku dan proses pembuatan roti apabila ada permintaan pelanggan dalam jumlah besar. Oleh karena hal itu, dibutuhkan proses peramalan permintaan pelanggan yang lebih ilmiah dan akurat, sehingga dapat berdampak efektif terhadap stok bahan baku dan dapat memaksimalkan hasil penjualan.

Ada banyak macam dan jenis metode yang bisa digunakan untuk proses peramalan seperti *Support Vector Regression* yang digunakan oleh (M. Raabith Rifqi, 2018) dengan menggunakan

data permintaan darah pada PMI Kota Malang. Kemudian penelitian yang dilakukan oleh (Mimin Putri Raharyani, 2017) yang juga menggunakan metode SVR untuk meramal jumlah pengunjung pariwisata. Selanjutnya penelitian dari (Karuniawan Susanto, 2018) yang menggunakan metode SVR dan *Ant Colony Optimization* untuk meramalkan produksi susu segar. Dan yang terakhir ada penelitian dari (Listiya Surtiningsih, 2017) yang menggunakan metode SVR di optimasi menggunakan algoritma genetika untuk meramalkan jumlah kunjungan wisata mancanegara ke Bali.

Pada penelitian ini akan menggunakan metode *Support Vector Regression* (SVR) yang menggunakan data *time series* dengan 4 fitur yang bertujuan untuk mendapatkan parameter terbaik dan menghasilkan nilai evaluasi menggunakan RMSE dengan nilai yang baik.

2. DASAR TEORI

2.1 Kajian Pustaka

Kajian pustaka yang pertama adalah penelitian yang dilakukan oleh M. Raabith Rifqi pada tahun 2018 dengan judul *Support Vector Regression Untuk Peramalan Permintaan Darah: Studi Kasus Unit Transfusi Darah Cabang – PMI Kota Malang*. Masukan (*input*) yang digunakan adalah Data permintaan darah bulan September 2013 – Agustus 2014. Evaluasi kinerja algoritme SVR pada penulisan ini menggunakan MAPE (Mean Absolute Percentage Error), dan dihasilkan nilai MAPE terbaik yaitu 3,899%, dengan $\lambda = 10$, $\sigma = 0,5$, $cLR = 0,01$, $\epsilon = 0,01$, jumlah fitur data = 4, jumlah iterasi = 5000. Pada penelitian ini digunakan data testing sebanyak 12 data. Penelitian selanjutnya dilakukan oleh Karuniawan Susanto pada tahun 2018 dengan judul *Normalisasi SVR dengan Ant Colony Optimization untuk Peramalan Tingkat Produksi Susu Segar (Studi Kasus pada koperasi susu Sae Pujon, Malang)*. *Input* data yang digunakan adalah Data penjualan susu segar mulai dari bulan Januari 2014 sampai dengan Desember 2015. Sistem peramalan produksi susu segar koperasi susu Sae Pujon menggunakan metode SVR dan *Ant Colony Optimization* menghasilkan nilai evaluasi menggunakan *Mean Absolute Square Error* (MAPE) 3,30425%. Hasil evaluasi tersebut bisa disimpulkan mempunyai akurasi yang baik karena nilai *Mean Absolute Square Error* kurang dari 10%. Selanjutnya penelitian

yang dilakukan oleh Listiya Surtiningsih pada tahun 2017, dengan topik peramalan jumlah kunjungan wisatawan mancanegara ke Bali dengan menggunakan metode *Support Vector Regression* dengan Algoritme Genetika. *Input* dari penelitian ini adalah data jumlah kunjungan wisatawan mancanegara ke Bali mulai bulan Januari 2001 sampai dengan Desember 2001. Berdasarkan pengujian yang sudah dilakukan, didapatkan parameter terbaik adalah λ sebesar 1 – 10, kompleksitas sebesar 1 – 100, Epsilon sebesar 0,00001 – 0,001, Gamma sebesar 0,00001 – 0,001, Sigma sebesar 0,01 – 3,5, Iterasi SVR sebanyak 1250, generasi GA sebanyak 90, Populasi sebanyak 70, Kombinasi Crossover Rate dan Mutation Rate sebesar 0,6 dan 0,4, jumlah fitur 2, dan jumlah periode peramalan 1 bulan. Nilai evaluasi yang didapatkan menggunakan MAPE yaitu sebesar 2,513%. Terakhir penelitian yang dilakukan oleh Mimin Putri Raharyani pada tahun 2017 dengan topik penelitian Implementasi Algoritme SVR pada Peramalan Jumlah kunjungan pariwisata. *Input* yang digunakan yaitu Data jumlah pengunjung pada bulan Januari 2013 sampai dengan Desember 2013. Nilai evaluasi MAPE yang dihasilkan sebesar kurang dari 10% dan bisa dikategorikan baik untuk memperamalan jumlah pengunjung pariwisata. Berdasarkan nilai MAPE tersebut rata-rata selisih antara hasil peramalan dengan data aktual adalah sebesar 115 jumlah pengunjung. Berdasarkan hasil uji data normalisasi dan data non normalisasi pada algoritme SVR, rata-rata nilai MAPE minimum yang diperoleh adalah 5,93% dengan data yang dinormalisasi dan rata-rata nilai MAPE minimum 3,98% untuk data yang tidak dinormalisasi. Nilai MAPE tersebut dapat disimpulkan bahwa normalisasi data belum tentu menghasilkan hasil peramalan yang lebih baik.

2.2 Dasar Teori

2.2.1 Peramalan

Menurut Makridakis (1991), peramalan (*forecasting*) adalah peramalan nilai-nilai dari sebuah peubah berdasarkan terhadap nilai yang diketahui dari peubah tersebut atau dari sebuah peubah yang berhubungan. Meramal atau memperamalan juga bisa berdasarkan kepada keahlian keputusan, yang juga didasarkan kepada data yang sudah ada sebelumnya. Sedangkan menurut Heizer dan Render (2009), peramalan atau peramalan adalah sebuah seni dan ilmu untuk memperamalkan kejadian di masa mendatang. Peramalan atau peramalan dapat dilakukan dengan melibatkan data masa lalu

(*historis*) lalu memproyeksikannya ke masa depan dengan suatu bentuk model yang matematis.

2.2.2 Support Vector Regression (SVR)

Algoritme SVR adalah teori yang diadaptasi dari teori *machine learning* yang sudah digunakan untuk memecahkan masalah klasifikasi, yaitu *Support Vector Machine (SVM)*. SVR ini adalah penerapan algoritme SVM dalam kasus regresi. Pada metode SVM adalah penerapan dari teori *machine learning* kasus klasifikasi yang menghasilkan nilai bulat, sedangkan pada algoritme *Support Vector Regression (SVR)* yaitu untuk penerapan kasus regresi yang menghasilkan keluaran berupa bilangan riil (Furi, Jordi, & Saepudin, 2015). Konsep algoritme SVR dapat menghasilkan nilai peramalam yang bagus karena SVR mempunyai kemampuan menyelesaikan masalah *overfitting* (Furi, Jordi, & Saepudin, 2015). *Overfitting* adalah perilaku data saat fase pelatihan atau *training* menghasilkan akurasi prediksi hampir sempurna (Yasin, Prahutama, & Utami, 2014). Tujuan algoritme SVR adalah menemukan garis pemisah atau bisa disebut dengan *Hyperplane* terbaik. *Hyperplane* terbaik dapat ditemukan dengan cara mengukur margin dengan *hyperplane* tersebut. Margin sendiri adalah jarak dari *hyperplane* dengan data yang terdekat. Data yang paling dekat dari margin disebut dengan *support vector* (Furi, Jordi, & Saepudin, 2015).

Dasar dari ide algoritme *Support Vector* untuk melakukan estimasi regresi adalah menghitung nilai fungsi linier, dimana α_i, α_i^* adalah *Lagrange non-negative multiplier*. Solusi dari masalah ini secara tradisional diperoleh dengan menggunakan paket pemrograman kuadratik. Permukaan aproksimasi optimal menggunakan formulasi yang telah dimodifikasi, setelah memperpanjang *Support Vector Regression* menjadi *non linier* ditunjukkan dengan persamaan dibawah berikut ini:

$$F(x) = \sum_{i=1}^1 (\alpha_j^* - \alpha_j) (K(x_i, x) + \lambda^2) \quad (2.1)$$

Seperti halnya kasus klasifikasi, hanya beberapa koefisien ($\alpha_j^* - \alpha_j$) yang tidak memiliki hasil nol, titik data yang sesuai tersebut disebut *Support Vector* (Vijayakumar & Wu, 1999).

2.2.3 Algoritme Sequential Learning

Menurut Vijayakumar & Wu (1999), proses *Sequential Learning* merupakan proses yang ada

di setiap proses perhitungan fungsi *Support Vector Regression*, dimana proses ini berguna untuk mendapatkan garis pemisah / *hyperplane* yang optimal. Berikut ini merupakan langkah-langkah dari proses *sequential learning*: (Vijayakumar & Wu, 1999).

1. Inisialisasi $\alpha_i = 0, \alpha_i^* = 0$, lalu hitung matrik R_{ij}

$$R_{ij} = (K(x_i, x) + \lambda^2) \text{ untuk } i, j = 1, \dots, n \quad (2.2)$$

Keterangan:

R_{ij} = matriks *hessian*

K = fungsi kernel

X_i = data ke - i

X_j = data ke - j

λ = Variabel Skalar

2. Untuk setiap data latih, $i = 1$ sampai n dihitung:

- a. $E_i = y_i - \sum_{j=1}^1 (\alpha_j^* - \alpha_j) R_{ij} \quad (2.3)$

- b. $\delta\alpha_i^* = \min\{\max[\gamma (E_i - \epsilon), -\alpha_i^*], C - \alpha_i^*\}$

$$\delta\alpha_i = \min\{\max[\gamma (-E_i - \epsilon), -\alpha_i], C - \alpha_i\} \quad (2.4)$$

- c. $\alpha_i^* = \alpha_i^* + \delta\alpha_i^*$

$$\alpha_i = \alpha_i + \delta\alpha_i \quad (2.5)$$

Keterangan:

E_i = Nilai Error

y_i = Nilai data normalisasi

α_i^* = *Lagrange multiplier*

α_i = *Lagrange multiplier*

R_{ij} = Matriks *Hessian*

$\delta\alpha_i^*$ = Variabel tunggal, bukan perkalian dari δ dengan α_i^*

$\delta\alpha_i$ = Variabel tunggal, bukan perkalian dari δ dengan α_i

γ = Nilai *Learning rate*

ϵ = Parameter *epsilon*

C = Parameter kompleksitas

$$\gamma = \frac{cLR}{\text{Max (Matrik Hessian)}} \quad (2.6)$$

Keterangan:

γ = Nilai *Learning rate*

cLR = Nilai *coefisien Learning rate*

3. Setelah itu kembali ke langkah kedua sampai kondisi iterasi maksimum atau $\max(|\delta\alpha_i|) < \epsilon$ dan $\max(|\delta\alpha_i^*|) < \epsilon$

4. Fungsi regresinya adalah

$$f(x) = \sum_{j=1}^n (\alpha_j^* - \alpha_j) (K(x_i, x_j) + \lambda^2) \quad (2.7)$$

Keterangan:

X_i = data ke - i

X_j = data ke - j

λ = Variabel Skalar

Selesai.

2.2.4 Fungsi Kernel

Dalam mendukung menyelesaikan permasalahan non-linier dengan algoritme *Support Vector Regression*, maka digunakanlah fungsi kernel. Dalam penyelesaian masalah linear dengan ruang dimensi yang tinggi, yang perlu dilakukan yaitu mengganti *inner product* (x_i dan x_j) dengan fungsi kernel. Kelebihan dari penggunaan fungsi kernal ini yaitu mampu berhubungan dengan ruang fitur berdimensi yang lebih tinggi tanpa perlunya perhitungan pemetaan eksplisit (Furi, Jodi, & Saepudin, 2015). Kerja algoritme *Support Vector regression* ditentukan oleh jenis fungsi kernel yang akan dipakai dan pengaturan parameter kernel (Che, & Wang, 2014). Fungsi kernel yang sering dipakai adalah *Radial Basis Function* (RBF) Kernel dengan persamaan yang ditunjukkan pada persamaan 2.8 berikut (Furi, Jodi, & Saepudin, 2015):

$$K(x_i, x_j) = \exp\left(-\frac{1}{2\sigma^2}\right)(x_i - x_j) \quad (2.8)$$

Keterangan:

X_i = Data ke - i

X_j = Data ke - j

σ = Standart deviasi

2.2.5 Normalisasi dan Denormalisasi

Normalisasi data adalah cara untuk mengubah data sehingga menjadi nilai yang memiliki kekuatan sama besar (Patel & Mehta, 2011). Normalisasi memiliki tujuan untuk mendapatkan data dengan ukuran yang lebih kecil yang dapat mewakili data yang asli tanpa kehilangan karakteristiknya sendiri. Rumus normalisasi seperti ditunjukkan dalam persamaan 2.9 (Parto & Sahu, 2015):

$$\text{Normalisasi} = \frac{x - \min}{\max - \min} \quad (2.9)$$

Keterangan:

x = data ke-i

Min = Nilai minimum dari data yang digunakan

Max = data maksimum dari data yang digunakan

Denormalisasi adalah proses pengembalian data ke nilai awal yang sebelumnya telah dilakukan proses normalisasi untuk mendapatkan data asli. Denormalisasi dilakukan untuk hasil akhir atau *output* dari *training* peramalan. Rumus denormalisasi seperti ditunjukkan pada persamaan 2.10 berikut:

$$\text{Denormalisasi} = y(\max - \min) + \min \quad (2.10)$$

Keterangan:

y = hasil *output* pelatihan

Min = data minimal

Max = data maksimal

2.2.6 Root Mean Square Error (RMSE)

Root Mean Square Error merupakan algoritme yang sering digunakan untuk menilai algoritma *machine learning* atau mesin pembelajaran, termasuk algoritme yang jauh lebih canggih daripada regresi linier (Conway & White, 2012). Nilai RMSE digunakan untuk membedakan kinerja model dalam periode kalibrasi dengan periode validasi serta digunakan untuk oerbandingan kinerja antara model individual dengan model prediksi yang lain (Hosseini, Javaherian, & Movahed, 2014).

$$RMSE = \sqrt{\frac{\sum_{j=i}^n (y' - y)^2}{n}} \quad (2.11)$$

Keterangan:

n = Jumlah keseluruhan data.

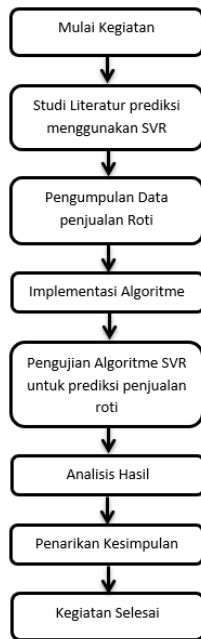
e = *Error rate*.

y' = Nilai keluaran atau *output* (prediksi).

y = Nilai aktual atau sebenarnya.

3. PERANCANGAN DAN IMPLEMENTASI

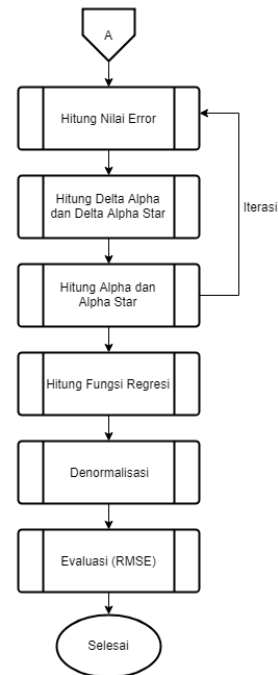
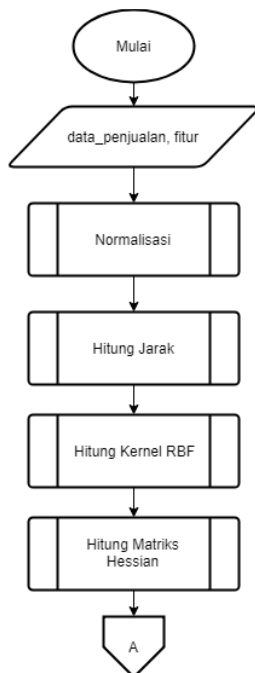
3.1. Tahapan Penelitian



Gambar 3.1 Blok Diagram Tahapan Penelitian

Gambar 3.1 mendeskripsikan langkah dalam penyusunan penulisan penelitian ini meliputi studi literatur, analisis kebutuhan sistem, pengumpulan data, perancangan sistem, implementasi sistem, pengujian sistem, dan penarikan kesimpulan.

3.2 Perancangan Algoritme



Gambar 3.2 Diagram Alir Proses SVR

Gambar 3.2 merupakan tahapan yang nantinya akan dilakukan pada algoritme SVR untuk meramalkan penjualan roti. Tahapan-tahapan tersebut adalah:

1. Data yang dibutuhkan untuk proses perhitungan adalah data penjualan dan juga jumlah fitur yang akan digunakan.
2. Lalu data tersebut akan dilakukan proses normalisasi.
3. Setelah normalisasi, selanjutnya akan dilakukan proses perhitungan jarak.
4. Setelah menghitung jarak, maka selanjutnya menghitung kernel, kernel yang digunakan adalah kernel RBF.
5. Setelah mendapatkan nilai kernel, lalu menghitung nilai matriks hessian.
6. Setelah itu melakukan hitung nilai *error*.
7. Setelah mendapat nilai *error*, selanjutnya menghitung nilai *delta alpha star* dan *delta alpha*.
8. Setelah itu mencari nilai *alpha star* dan *alpha*.
9. Setelah mendapat nilai *alpha star* dan *alpha*, maka dilakukan proses iterasi untuk mencari nilai *error* lagi, begitu seterusnya hingga mendapat nilai *alpha star* dan *alpha* yang optimal.
10. Setelah mendapat nilai *alpha star* dan *alpha* yang optimal, maka selanjutnya mencari nilai regresi.
11. Setelah didapatkan nilai regresi, dilakukan proses denormalisasi.

12. Setelah nilai di denormalisasi, maka langkah terakhir adalah melakukan proses evaluasi.
13. Setelah itu sistem akan mengeluarkan hasil berupa nilai evaluasi menggunakan RMSE.

4. PENGUJIAN DAN ANALISIS

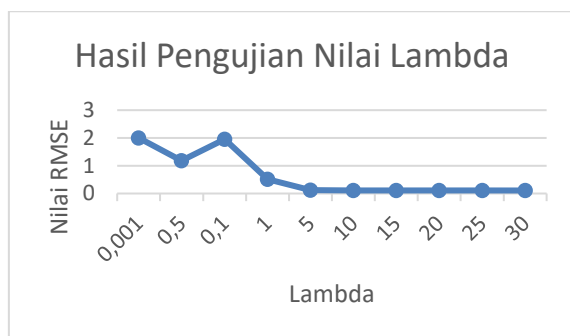
4.1 Pengujian Parameter Nilai λ

Pengujian menggunakan target prediksi pada hari Senin tanggal 29 Januari 2018. Roti yang diprediksi adalah penjualan roti manis. Ada 10 nilai λ yang akan diuji, dan nilai parameter lain akan ditentukan dahulu. Berikut nilai parameter lain yang akan digunakan untuk mendapatkan nilai λ terbaik:

- a. *Coefisien Learning Rate* (cLR) = 0,01
- b. *Complexity* = 5
- c. *Epsilon* = 0,00001
- d. *Sigma* = 0,5
- e. Jumlah Iterasi = 10

Tabel 4.1 Hasil Pengujian Nilai λ

No.	Nilai λ	Hasil RMSE
1	0,001	2,0035
2	0,5	1,1818
3	0,1	1,9534
4	1	0,5091
5	5	0,1226
6	10	0,1117
7	15	0,1097
8	20	0,1090
9	25	0,1087
10	30	0,1086



Gambar 4.1 Grafik Hasil Uji Nilai λ

Parameter λ berfungsi untuk menunjukkan ukuran skalar untuk pemetaan ruang pada kernel SVR (Vijayakumar, & Wu, 1999). Berdasarkan hasil pengujian parameter λ yang dilakukan, Roti Manis memiliki hasil RMSE paling optimal pada parameter

λ 5. Seperti ditunjukkan pada gambar grafik 6.1 terlihat nilai RMSE terhadap nilai λ cenderung menurun, tetapi terjadi peningkatan nilai RMSE pada nilai λ 0,1 yang kemudian nilai RMSE kembali menurun dan memberikan penurunan yang signifikan sampai nilai λ 5, karena untuk nilai λ selanjutnya cenderung konvergen atau tidak mengalami perubahan yang signifikan. Hal ini menunjukkan bahwa nilai λ yang terlalu kecil dapat menyebabkan nilai skala ruang pemetaan kernel tidak stabil dan menyebabkan nilai *error rate* semakin meningkat.

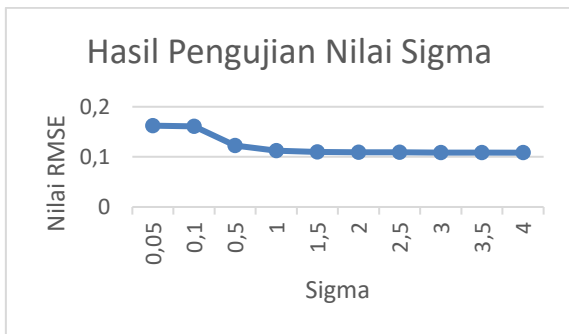
4.2 Pengujian Parameter Nilai σ

Pengujian selanjutnya adalah menguji parameter nilai σ . Pengujian dilakukan untuk mendapatkan nilai σ terbaik sehingga dapat menghasilkan solusi peramalan terbaik. Ada 10 nilai σ yang akan diuji, dan nilai parameter lain akan ditentukan terlebih dahulu, namun untuk nilai λ akan digunakan nilai 5 yang merupakan hasil terbaik dari pengujian sebelumnya. Berikut nilai parameter lain yang akan digunakan untuk mendapatkan nilai σ terbaik:

- a. *Coefisien Learning Rate* (cLR) = 0,01
- b. *Complexity* = 5
- c. *Epsilon* = 0,00001
- d. λ = 5
- e. Jumlah Iterasi = 10

Tabel 4.2 Hasil Pengujian Nilai σ

No.	Nilai σ	Hasil RMSE
1	0,05	0,1623
2	0,1	0,1609
3	0,5	0,1226
4	1	0,1124
5	1,5	0,1101
6	2	0,1093
7	2,5	0,1089
8	3	0,1087
9	3,5	0,1085
10	4	0,1084



Gambar 4.2 Grafik Hasil Uji Nilai Sigma

Berdasarkan pengujian *sigma* yang dilakukan terhadap roti manis, didapatkan hasil RMSE paling optimal yaitu 0,1101 dengan nilai *sigma* = 1,5. Pada grafik 6.2 terlihat pada nilai *sigma* 0,1 grafik mengalami penurunan lalu mulai konvergen pada nilai *sigma* 1,5 sampai sampai nilai *sigma* 4. Hal ini menunjukkan bahwa nilai *sigma* yang terlalu kecil dapat menyebabkan persebaran data yang tidak sesuai, sehingga menyebabkan nilai error rate menjadi semakin meningkat. Sebagaimana pengertiannya, bahwa nilai *sigma* adalah konstanta dari fungsi *Kernel RBF* untuk mengatur persebaran data kedalam dimensi fitur yang lebih tinggi. (Furi, Jordi, & Saepudin, 2015).

4.3 Pengujian Parameter Nilai cLR

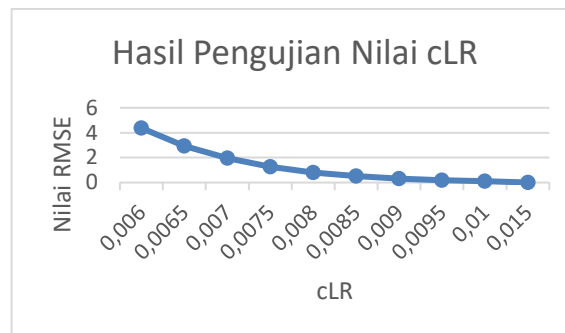
Selanjutnya dilakukan pengujian parameter nilai *Coefisien Learning Rate (cLR)*. Pengujian dilakukan untuk mendapatkan nilai *cLR* terbaik sehingga bisa menghasilkan nilai evaluasi yang terbaik. Ada 10 parameter *cLR* yang akan diuji, dan nilai parameter lain akan ditentukan terlebih dahulu. Untuk nilai *lambda* dan *sigma* akan menggunakan nilai terbaik yang sudah didapatkan pada pengujian sebelumnya, yaitu untuk nilai *lambda* = 5 dan nilai *sigma* = 1. Berikut nilai parameter lain yang akan digunakan untuk mendapatkan nilai *cLR* terbaik:

- a. *Lambda* = 5
- b. *Sigma* = 1,5
- c. *Complexity* = 5
- d. *Epsilon* = 0,00001
- e. Jumlah Iterasi = 10

Tabel 4.3 Hasil Pengujian Nilai cLR

No.	Nilai cLR	Hasil RMSE
1	0,006	4,3759
2	0,0065	2,9480
3	0,007	1,9544
4	0,0075	1,2734

5	0,008	0,8143
6	0,0085	0,5104
7	0,009	0,3131
8	0,0095	0,1878
9	0,01	0,1101
10	0,015	0,0035



Gambar 4.2 Grafik Hasil Uji Nilai cLR

Berdasarkan pengujian yang sudah ditunjukkan pada Tabel 6.3, nilai *cLR* terbaik ada pada 0,015 yang mempunyai nilai RMSE paling kecil yaitu 0,0035. Pada grafik 6.3 terlihat bahwa nilai RMSE cenderung mengalami penurunan seiring dengan nilai *cLR* yang semakin besar. Hal ini menunjukkan bahwa nilai *cLR* yang terlalu kecil dapat menyebabkan peningkatan nilai *error rate* dan menghasilkan nilai peramalan yang buruk. Hal ini dikarenakan nilai *cLR* mempengaruhi nilai *gamma*, dan apabila nilai *gamma* keluar dari batas solusi maka nilai *alpha* dan *alpha star* yang didapatkan tidak pas dan menyebabkan nilai evaluasinya sangat besar. Parameter *Coefisien Learning Rate (cLR)* sendiri merupakan konstanta laju pembelajaran (Vijayakumar, & Wu, 1999).

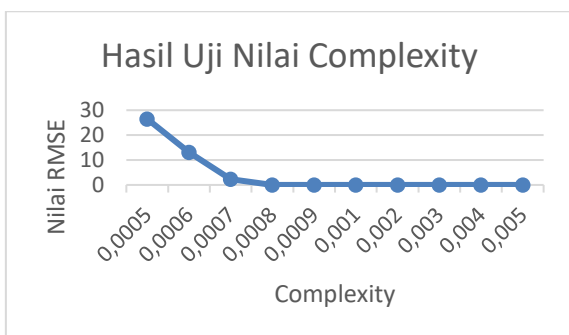
4.4 Pengujian Parameter Nilai Complexity

Setelah menguji nilai *cLR* maka selanjutnya dilakukan pengujian nilai *complexity*. Pengujian dilakukan untuk mendapatkan nilai *complexity* terbaik sehingga didapatkan nilai hasil evaluasi terkecil. Untuk pengujian nilai *complexity* digunakan 10 parameter nilai. Untuk nilai *lambda* menggunakan nilai 5, nilai *sigma* 1, dan nilai *cLR* 0,015. Untuk nilai lainnya akan ditentukan seperti berikut:

- a. *Lambda* = 5
- b. *Sigma* = 1,5
- c. *cLR* = 0,015
- d. *Epsilon* = 0,00001
- e. Jumlah Iterasi = 10

Tabel 4.4 Hasil Pengujian Nilai Complexity

No.	Nilai Complexity	Hasil RMSE
1	0,0005	26,4875
2	0,0006	13,0248
3	0,0007	2,2285
4	0,0008	0,0035
No.	Nilai Complexity	Hasil RMSE
5	0,0009	0,0035
6	0,001	0,0035
7	0,002	0,0035
8	0,003	0,0035
9	0,004	0,0035
10	0,005	0,0035



Gambar 4.4 Grafik Hasil Uji Nilai Complexity

Parameter *Complexity* merupakan batas penalti toleransi terhadap kesalahan sebuah peramalan (Furi, Jordi, & Saepudin, 2015). Berdasarkan hasil pengujian parameter nilai C yang sudah dilakukan, ditunjukkan pada Tabel 6.4 ditunjukkan bahwa nilai C paling optimal adalah 0,0008 yang mempunyai nilai evaluasi RMSE sebesar 0,0035. Grafik 6.4 menunjukkan bahwa pada nilai C 0,0005 cenderung mempunyai nilai RMSE yang tinggi, lalu grafik menurun dan terus mengalami penurunan sampai pada nilai C 0,0008. Pada nilai C 0,0008 grafik terlihat konvergen dan tidak mengalami perubahan lagi. Hal ini sesuai dengan pernyataan Furi, Jordi, & Saepudin (2015) yang menyatakan bahwa semakin besar nilai *Complexity* maka semakin menjadikan model peramalan semakin tidak mentoleransi kesalahan, sehingga memberikan nilai peramalan yang baik.

4.5 Pengujian Parameter Nilai Epsilon

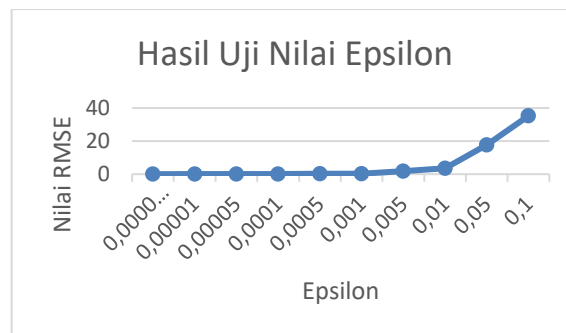
Selanjutnya dilakukan pengujian terhadap nilai *epsilon*. Pengujian dimaksudkan agar mendapat nilai *epsilon* terbaik sehingga didapatkan nilai evaluasi yang kecil. Untuk pengujian nilai *epsilon* akan digunakan 10 parameter nilai. Untuk jumlah iterasi menggunakan iterasi 10, sedangkan untuk nilai

lainnya menggunakan nilai terbaik yang sudah didapatkan dari pengujian sebelumnya. Berikut nilai lain yang digunakan:

- f. $\Lambda = 5$
- g. $\Sigma = 1,5$
- h. $cLR = 0,015$
- i. $Complexity = 0,0008$
- j. Jumlah Iterasi = 10

Tabel 4.5 Hasil Pengujian Nilai Epsilon

No.	Nilai Epsilon	Hasil RMSE
1	0,000005	0,0017
2	0,00001	0,0035
3	0,00005	0,0176
4	0,0001	0,0352
5	0,0005	0,1763
6	0,001	0,3526
7	0,005	1,7631
8	0,01	3,5262
9	0,05	17,6313
10	0,1	35,2625



Gambar 4.5 Grafik Hasil Uji Nilai Epsilon

Parameter *epsilon* digunakan untuk mengatur batas kesalahan fungsi regresi $f(x)$. Nilai *epsilon* tersebut menyelubungi nilai dari fungsi $f(x)$ sehingga akan membentuk daerah yang disebut daerah *error zone*, dan jika nilai $f(x)$ melebihi *error zone* yang terbentuk maka akan dikenakan penalti sebesar nilai C yang sudah ditentukan (Furi, Jordi, & Saepudin, 2015). Berdasarkan pengujian nilai *epsilon* yang sudah dilakukan, terlihat pada Tabel 6.5 bahwa nilai *epsilon* terbaik didapatkan pada nilai *epsilon* 0,000005 yang memiliki nilai evaluasi sebesar 0,0017. Grafik 6.5 menunjukkan bahwa nilai *epsilon* kecil cenderung konstan, lalu pada nilai *epsilon* = 0,0005 grafik mulai terlihat naik dan terus meningkat seiring dengan nilai *epsilon* yang semakin besar, sehingga menunjukkan nilai *epsilon* yang besar akan melakukan proses pembelajaran yang terlalu cepat sehingga hasil yang didapat tidak maksimal. Nilai *epsilon* yang

terlalu besar juga dapat menyebabkan pencarian solusi menjadi keluar batas.

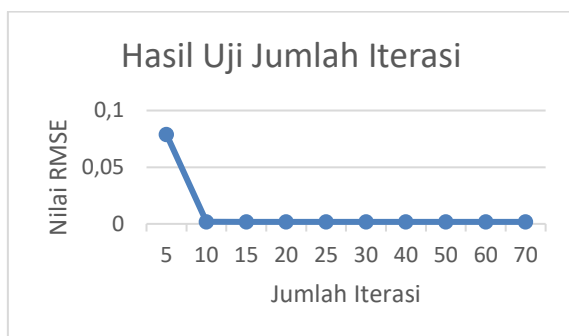
4.6 Pengujian Jumlah Iterasi

Untuk pengujian terakhir adalah melakukan uji jumlah iterasi. Pengujian ini dilakukan agar didapatkan nilai evaluasi terkecil sehingga proses peramalan semakin baik. Ada 10 parameter jumlah iterasi yang digunakan, sedangkan untuk nilai lainnya menggunakan nilai terbaik yang sudah didapatkan pada pengujian sebelumnya, berikut nilai lain yang digunakan:

- k. $\lambda = 5$
- l. $\sigma = 1,5$
- m. $cLR = 0,015$
- n. $Complexity = 0,0008$
- o. $Epsilon = 0,000005$

Tabel 4.6 Hasil Pengujian Jumlah Iterasi

No.	Jumlah Iterasi	Hasil RMSE
1	5	0,078939534300966
2	10	0,0017916568074186
3	15	0,0017631386324354
4	20	0,0017631275855779
5	25	0,0017631275813001
6	30	0,0017631275812996
7	40	0,0017631275812985
8	50	0,0017631275812969
9	60	0,0017631275813012
10	70	0,0017631275812969



Gambar 4.6 Grafik Hasil Uji Jumlah Iterasi

Berdasarkan pengujian jumlah iterasi diatas, dapat disimpulkan bahwa jumlah iterasi sangat berpengaruh besar terhadap proses *sequential learning*. Pada pengujian jumlah iterasi ditunjukkan pada Tabel 6.6 bahwa jumlah iterasi terbaik didapatkan pada jumlah iterasi 50 yang menghasilkan nilai evaluasi sebesar 0,0017631275812969. Terlihat pada grafik 6.6 bahwa grafik mengalami penurunan pada iterasi

ke 10 dan mulai konvergen setelahnya. Pengujian ini menunjukkan bahwa semakin besar jumlah iterasi maka algoritme SVR akan semakin teliti dan observasi terhadap pola data menjadi semakin meningkat.

4.7 Hasil pengujian

Berikut perbandingan nilai hasil prediksi dan nilai aktual pada roti manis menggunakan algoritme SVR dengan parameter diatas:

Tabel 4.7 Perbandingan Nilai Prediksi dan Nilai Aktual

Penjualan Roti Manis Bulan Januari 2018	
Nilai Prediksi	Nilai Aktual
129,9982	130
355,9982	356
166,9982	167
234,9982	235
109,9982	110

5. KESIMPULAN

Dari hasil pengujian yang sudah dilakukan dapat ditarik kesimpulan bahwa pada kasus penjualan roti manis pada Toko Roti Harum Bakery menghasilkan nilai evaluasi menggunakan RMSE sebesar 0,0017 dengan parameter nilai $\lambda = 5$, $\sigma = 1,5$, $cLR = 0,015$, $complexity = 0,0008$, $epsilon = 0,000005$ dan jumlah iterasi sebanyak 50. Hasil ini terbilang sangat baik karena hasil evaluasi sangat dekat dengan angka 0.

6. DAFTAR PUSTAKA

Che, J., Wang, J., 2014. Short Term Load Forecasting Using Support Vector Regression Combination Model. *Applied Energy* 132, pp. 602 - 609.

Conway, D., & White, J. M. (2012). *Machine Learning for Hackers*. (J. Steele, Ed.).

Furi, R. P., Jondri & Saepudin, D., 2015. Peramalan Financial Time Series Menggunakan *Independent Component Analysis* dan *Support Vector Regression* (Studi Kasus: IHSG dan JII). S1. Telkom University.

Heizer, Jay dan Barry Render., 2009. *Manajemen Operasi Buku 1 Edisi 9*. Jakarta: Salemba Empat.

- Hosseini, M., Javaherian, A., & Movahed, B. (2014). Determination of permeability index using Stoneley slowness analysis, NMR models, and formation evaluations: a case study from a gas reservoir, south of Iran. *Journal of Applied Geophysics*, 109, 80–87.
- Makridakis, Spyros dkk., 1991. *Metode dan Aplikasi Peramalan*. Edisi Kedua. Jakarta: Erlangga.
- Parto, S. & Sahu, K., 2015. *Normalization: A Preprocessing Stage*. Burla, Odisha, India: Departmen of CSE & IT.
- Patel, V, R. & Mehta, R. G., 2011. *Impact of Outlier Removal and normalization Approach in Modifid K-means Clustering Algorithm*. *IJCSI International Journal of Computer Science*.
- Raharyani, M. P., 2017. Implementasi Algoritme Support Vector Regression pada Peramalan Jumlah pengunjung Pariwisata. S1. Universitas Brawijaya.
- Rangkuti, Freddy., 2005. Analisis SWOT : Teknik Membedah Kasus Bisnis. Jakarta: PT. Gramedia.
- Rifqi, M. R., 2018. Support Vector Regression Untuk Peramalan Permintaan Darah: Studi Kasus Unit Transfusi Darah Cabang – PMI Kota Malang. S1. Universitas Brawijaya.
- Surtiningsih, L., 2017. Peramalan Jumlah Kunjungan Wisatawan Mancanegara ke Bali Menggunakan Support Vector Regression dengan Algoritme Genetika. S1. Universitas Brawijaya.
- Susanto, K., 2018. Normalisasi SVR dengan Ant Colony Optimization untuk peramalan tingkat produksi susu segar (Studi Kasus pada koperasi susu Sae Pujon, Malang). S1. Universitas Brawijaya.
- Vijayakumar, S. Wu, Si, 1999. *Sequential Support Vector Classifiers and Regression*. Genoa: International Conference on Soft Computing.