

PREDIKSI JUMLAH PENUMPANG DAN PENAMBAHAN GERBONG KERETA API MENGGUNAKAN METODE *SUPPORT VECTOR REGRESSION* (SVR)

PREDICTION OF TOTAL PASSENGERS AND ADDITION OF RAILWAY TRAIN USING SUPPORT VECTOR REGRESSION (SVR) METHOD

Ufiq Nur ¹Aeni, Anggunmeka Luhur Prasasti², Meta Kallista³
^{1,2,3}Prodi S1 Teknik Komputer, Fakultas Teknik, Universitas Telkom
¹ufiqnuraeni@student.telkomuniversity.ac.id, ²anggunmeka@telkomuniversity.ac.id,
³metakallista@telkomuniversity.ac.id

Abstrak

Transportasi kereta api dapat dibagi menjadi transportasi penumpang dan transportasi kargo. Oleh karena itu, proses perencanaan dan operasional terkait dengan kapasitas penumpang adalah bidang yang lebih menarik untuk dikaji lebih dalam dari masalah prediksi penambahan gerbong. *Support Vector Regression* (SVR) merupakan pengembangan dari metode *Support Vector Machine* untuk kasus regresi. Metode ini mampu mengatasi *overfitting*, dalam pemilihan parameter SVR menggunakan algoritma *Grid Search*. Data yang digunakan pada Tugas Akhir ini menggunakan data jumlah penumpang kereta api Argo Parahyangan periode 2019 dan jenis kelas kereta api dibagi menjadi 2, yaitu kelas Ekonomi dan kelas Eksekutif. Rata-rata dari hasil pengujian SVR menggunakan kernel RBF menghasilkan nilai performansi MAE (*Mean Absolute Error*) sebesar 0.1276, MSE (*Mean Square Error*) sebesar 0.1796 dan MAPE (*Mean Absolute Percentage Error*) sebesar 0,00376.

Kata kunci : Algoritma *Grid Search*, Kereta Api, Prediksi, *Support Vector Regression*.

Abstract

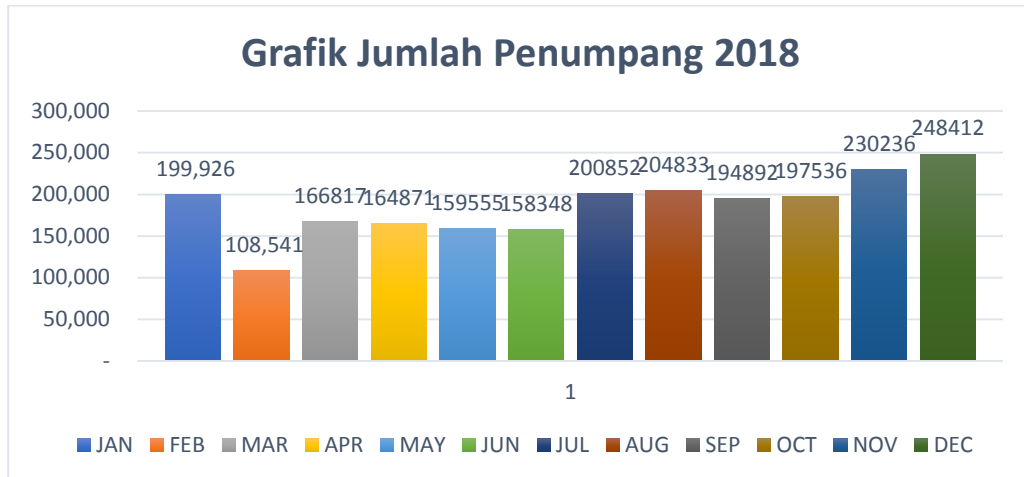
Rail transportation can be divided into passenger transportation and cargo transportation. Therefore, the planning and operational process related to passenger capacity is a more interesting field to study more deeply than the prediction problem of the addition of railway. *Support Vector Regression* (SVR) is a development of the *Support Vector Machine* method for regression cases. This method is able to overcome *overfitting*, in selecting SVR parameters using the *Grid Search* algorithm. The data used in this Final Project uses data on the total of Argo Parahyangan train passengers in the 2019 period and the types of train classes are divided into 2, namely Economy and Executive classes. The average SVR test results using RBF kernel produce MAE (*Mean Absolute Error*) performance values of 0.1276, MSE (*Mean Square Error*) of 0.1796 and MAPE (*Mean Absolute Percentage Error*) of 0.00376.

Keywords: *Grid Search Algorithm*, Prediction, Railway Train, *Support Vector Regression*.

1. Pendahuluan

Pembangunan ekonomi membutuhkan jasa transportasi yang cukup serta memadai. Tanpa adanya transportasi sebagai sarana pendukung, tidak dapat diharapkan tercapainya hasil yang memuaskan dalam usaha pengembangan ekonomi dari suatu negara. Untuk tiap tingkatan perkembangan atau pertumbuhan ekonomi dari suatu negara, diperlukan kapasitas angkutan yang *optimal*. Namun perlu diperhatikan bahwa penentuan kapasitas transportasi dan tingkatan investasinya tidak merupakan hal yang mudah untuk dilaksanakan.

PT Kereta Api Indonesia (PT. KAI) adalah salah satu BUMN yang bergerak di bidang pelayanan jasa transportasi darat, mulai tahun 1998 perusahaan ini berbentuk PT (Persero). Keberhasilan PT KAI dalam menghasilkan jasanya tidak hanya tergantung pada keunggulan teknologi, sarana dan prasarana maupun dana yang tersedia, melainkan juga tergantung kepada kepercayaan publik terhadap kinerja manajemen yang mempunyai peranan penting untuk menjamin kelangsungan aktivitas bisnis perusahaan sesuai yang diharapkan[1].



Gambar 1 Grafik Jumlah Penumpang Kereta Api Tahun 2018

Salah satu cara agar tingkat kepuasan dan kenyamanan penumpang dapat diprediksi, yaitu dengan membangun aplikasi untuk memprediksi gerbong dan jumlah penumpang di KA Argo Parahyangan, sehingga dapat membantu PT. KAI Stasiun Bandung dalam mengambil kebijakan/keputusan yang tepat. Salah satu metode yang dapat digunakan adalah metode *Support Vector Regression* (SVR). Metode ini adalah penerapan dari metode SVM (*Support Vector Machine*) dalam kasus regresi dan dapat dikatakan sebagai perbaikan dari metode Regresi Linier / Analisis Regresi.

Support Vector Regression (SVR) adalah algoritma yang memasukkan semua data menjadi satu *cluster* dengan tingkat kepastian yang akurat dengan tetap meminimalisasi nilai *error*[2]. Teknik ini dapat menghasilkan sebuah fungsi dengan hasil yang bergelombang mengikuti jalur data yang terbentuk dengan menggunakan fungsi *kernel* atau disebut *kernel trick*. Kelebihan dari SVR dapat mengatasi masalah *overfitting*. Sehingga dengan data yang ada, *Support Vector Regression* bisa menghasilkan performansi yang lebih bagus dan dapat digunakan untuk memprediksi tingkat kepuasan dan kenyamanan penumpang kereta api yang ada di Stasiun Bandung ke depannya.

2. Metode

2.1 Prediksi Jumlah Penumpang

Pada dasarnya, cara kerja sistem prediksi jumlah penumpang dan penambahan gerbong mempelajari dari data-data *real-time*, didapatkan dari PT. KAI DAOP 2 Jawa Barat. Tujuannya adalah untuk mendapatkan data-data hasil prediksi. Rancangan Algoritma pada Tugas Akhir ini berupa implementasi dari akurasi metode *Support Vector Regression* menggunakan beberapa indikator untuk digunakan pada sebuah aplikasi berbasis web.

Pada penelitian ini, kelas kereta api yang digunakan adalah kelas bisnis dan kelas eksekutif pada Kereta Api Argo Parahyangan. Berikut adalah sedikit penjelasan dari masing-masing kelas pada kereta api:

- Kelas Ekonomi Premium: kenyamanan dan elegan seperti kelas eksekutif, dengan fasilitas AC, televisi, dan tempat duduk standar kelas ekonomi yang memiliki sebanyak 80 tempat duduk. Kelas ini menggunakan model kursi *private seat* yang dipisahkan dengan *arm rest*.
- Kelas Eksekutif: kelas yang hanya memiliki kapasitas per gerbong 50 tempat duduk. Jarak antar penumpang yang lebih lapang, busa bangku yang lebih empuk, dapat diubah untuk arah hadap, dan dapat direbahkan sampai 50 derajat.

2.2 Klasifikasi

Klasifikasi adalah proses menemukan model dari satu *set* data. Tujuan klasifikasi adalah untuk membuat keputusan dengan memprediksi suatu kasus berdasarkan hasil klasifikasi yang diperoleh[3]. Metode klasifikasi yang digunakan adalah klasifikasi *Support Vector Regression*. SVR memiliki keunggulan teoretis dibandingkan algoritma yang lain, seperti tidak adanya *minimum* lokal dalam fase optimisasi model dan SVR adalah algoritma yang efisien yang dapat digunakan sebagai alternatif dari algoritma regresi linier.

Untuk mengetahui seberapa bagus model *Learning* diperlukan untuk mengukur performanya. Untuk mengukur nilai keakuratan model regresi yang digunakan, diperlukan perhitungan nilai *error* antara prediksi dan data sebenarnya. Data sebenarnya dinyatakan dengan simbol γ , nilai prediksinya adalah \hat{y} dan banyaknya data adalah m . Maka beberapa nilai *error* yang sering dipakai diantaranya dinyatakan sebagai berikut:

1. *Mean Square Error* (MSE)

$$\text{MSE} = \frac{1}{m} \sum_{i=1}^m ((\hat{y}_i - y_i)^2)$$

2. *Mean Absolute Percentage Error* (MAPE)

$$\text{MAPE} = \frac{\sum_i^m \text{APE}}{m} \text{ dimana } \text{APE} = \frac{\sum_i^m |\hat{y}_i - y_i|}{y_i} 100$$

3. *Mean Absolute Error* (MAE)

$$\text{MAE} = \frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|$$

4. *Root Mean Square Error* (RMSE)

$$\sqrt{\frac{1}{m \times n} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} [f(i, j) - g(i, j)]^2}$$

2.3 Support Vector Regression (SVR)

Dalam Tugas Akhir ini, SVR diselidiki sebagai teknik alternatif untuk mendekati analisis teknik yang kompleks. Parameter ditentukan dengan mencobakan beberapa nilai rentang tertentu untuk membangun *hyperplane*. Kinerja SVR dipengaruhi oleh tiga parameter, yaitu:

1. Parameter *Complexity* (C) menyajikan *trade-off* antara kompleksitas model dan jumlah sehingga penyimpangan yang lebih besar dari C dapat ditoleransi. Untuk nilai C yang besar, pengoptimalan akan memilih *hyperplane* dengan margin lebih kecil jika *hyperplane* tersebut melakukan pekerjaan yang lebih baik untuk mengklasifikasikan semua poin pelatihan dengan benar. Sebaliknya, nilai C yang sangat kecil akan menyebabkan pengoptimal mencari *hyperplane* pemisah margin yang lebih besar, bahkan jika *hyperplane* tersebut salah mengklasifikasikan lebih banyak poin[8].
2. Epsilon (ϵ) mengontrol lebar Zona sensitif, ϵ digunakan untuk menyesuaikan data pelatihan. Epsilon juga digunakan untuk mengatur batas kesalahan fungsi $f(x)$.
3. Gamma (γ) merupakan parameter fungsi *kernel*[4]. Untuk mendapatkan nilai gamma harus memakai nilai parameter *cLR* (*coefisien Learning Rate*). Parameter *cLR* merupakan konstanta laju pembelajaran. Parameter Gamma menentukan seberapa jauh pengaruh dari satu sampel training dataset dengan nilai rendah berarti “jauh”, dan nilai tinggi berarti “dekat”. Dengan gamma yang rendah, titik yang berada jauh dari garis pemisah yang masuk akal dipertimbangkan dalam perhitungan untuk garis pemisah. Ketika gamma tinggi berarti titik – titik berada di sekitar garis yang masuk akal akan dipertimbangkan dalam perhitungan[7].

Penggunaan *kernel* yang berbeda sudah pasti mempunyai kelebihan dan kekurangan. Berikut merupakan fungsi *kernel* yang populer dan sering digunakan:

1. *Linear Kernel*

Linear kernel merupakan fungsi *kernel* yang paling sederhana. *Linear kernel* digunakan ketika data yang dianalisis sudah terpisah secara *linear*. *Linear kernel* cocok ketika terdapat banyak fitur dikarenakan pemetaan ke ruang dimensi yang lebih tinggi tidak benar – benar meningkatkan kinerja seperti pada klasifikasi teks. Dalam klasifikasi teks, baik jumlah *instances* (dokumen) maupun jumlah fitur (kata) sama sama besar.

2. *Sigmoid Kernel*

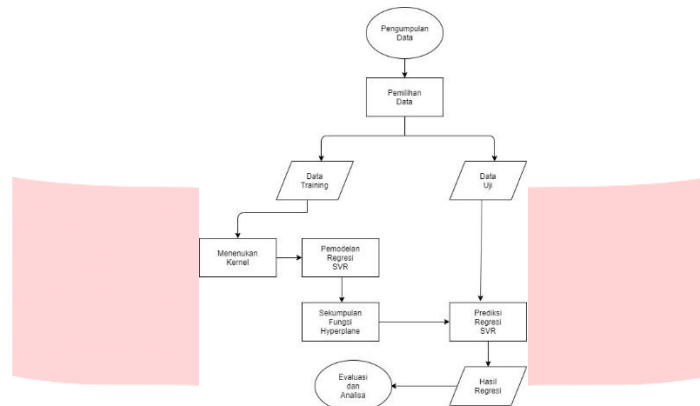
Kernel sigmoid bekerja seperti fungsi *basis radial* (RBF), juga dikenal sebagai *kernel Gaussian*, dalam kasus tertentu dan digunakan untuk fleksibilitasnya dalam menyesuaikan data. Ini membuatnya cocok untuk klasifikasi nonlinier dalam aplikasi praktis. *Kernel sigmoid* berperilaku seperti koefisien korelasi yang menekankan kesamaan dalam orientasi.

3. *RBF Kernel*

RBF kernel merupakan fungsi *kernel* yang biasa digunakan dalam analisis ketika data tidak terpisah secara linear. RBF mempunyai sejumlah fungsi menarik yang mampu menghubungkan pendekatan *non-linear* dan melakukan interpolasi Vapnik. Menurut penelitian Schölkopf yang sudah dibuktikan[5], mengatakan bahwa *kernel* RBF tidak hanya jauh lebih mudah untuk diterapkan, tetapi juga dapat diaplikasikan dalam kasus *non-linear* pemetaan data pelatihan

menjadi berhingga pada dimensi ruang. Dengan demikian, RBF sangat cocok untuk menangani masalah hubungan *non-linear*.

2.4 Algoritma Grid Search Cross Validation

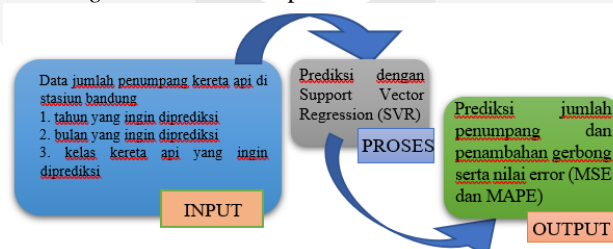


Gambar 2. Diagram Alir Algoritma *Grid Search Cross Validation*[6]

Gambar 2 menjelaskan, untuk memilih nilai optimal dari parameter ϵ , C , dan g yang terlibat dalam model SVR, algoritma pencarian *grid* digunakan dalam penelitian ini. Ide dasar dari algoritma pencarian *grid* adalah untuk mencoba setiap nilai yang mungkin dari parameter dalam ruang tertentu dengan jarak langkah tertentu, dan parameter yang mengoptimalkan kinerja model SVR dengan akurasi terbaik dapat diturunkan berdasarkan persilangan. validasi. Proses algoritma diilustrasikan melalui diagram alir, seperti yang ditunjukkan pada Gambar 2. Data eksperimen yang dikumpulkan dari koefisien redistribusi momen dibagi menjadi dua set (60% dan 45% untuk pelatihan dan 40% dan 55% untuk pengujian) untuk SVR. Berdasarkan algoritma pencarian *grid* dan validasi silang, nilai parameter untuk SVR ditentukan sebagai berikut: $\epsilon = 0.01$, $C = 1000$, dan $g = 1.3$ untuk kelas Ekonomi dan $\epsilon = 0.1$, $C = 10000$, dan $g = 3.2$ untuk kelas Eksekutif.

3. Implementasi Sistem

Prediksi jumlah penumpang dan penambahan gerbong ini diimplementasikan dengan Pembagian data yang dilakukan dengan cara membagi data menjadi dua, yaitu *Data Training* dan *Data Testing* dengan persentase 40:60 untuk Kelas Ekonomi dan 55:45 untuk Kelas Eksekutif. Pembagian data meliputi *Data Testing* 40% dan 55% dari 356 data, diperoleh 142 data dan 195 data dan untuk *Data Training* 60% dan 45% diperoleh 214 data dan 161 data.



Gambar 3. Diagram Blok Sistem

Pada Gambar 3 adalah gambaran umum sistem. Pada proses *input* sistem akan dimasukkan data berupa *file excel* dengan parameter berupa tahun, bulan dan 2 jenis kelas kereta api Argo Parahyangan, yaitu Kelas Ekonomi dan Kelas Eksekutif. Kemudian pada *input user* akan memasukkan tanggal, bulan, dan tahun yang akan diprediksi.

Selanjutnya *input* sistem yang berupa *file excel* akan diproses dengan metode SVR atau GPR untuk membuat model data nya. Setelah model data sudah jadi, maka dapat memprediksi masukkan dari *user* berupa tahun, bulan dan hari atau kelas kereta api Argo Parahyangan yang diprediksi. Selanjutnya hasil dari metode SVR atau GPR ini akan diketahui berapa tingkat atau jumlah penumpang dari 2 jenis kelas kereta api Argo Parahyangan. Kemudian setelah hasil keluar akan dilakukan analisis dari segi akurasi yaitu dari nilai *error* guna mendapatkan analisa yang

diharapkan tepat dengan sasaran. Disertakan juga prediksi penambahan jumlah penumpang kereta api Argo Parahyangan.

3.1 Hasil Pengujian Nilai *Test Size*

Pada bagian ini, pengujian nilai *test size* menggunakan algoritma *Grid Search* untuk menghasilkan nilai MAPE yang terbaik. MAPE terbaik adalah nilai yang mendekati 1. Pengujian dilakukan dengan mengubah nilai *test size* sebagai pada pembagian *Data Training* dan *Data Testing*. Pengujian nilai *test size* dibagi menjadi dua, yaitu pengujian pada kereta Kelas Ekonomi dan kereta Kelas Eksekutif.

Tabel 1. Hasil Perbandinga Nilai *Test Size* Ekonomi

<i>Testing</i>	<i>Training</i>	MAPE
10%	90%	0.0003365
15%	85%	0.0003347
20%	80%	0.0003335
25%	75%	0.0003333
30%	70%	0.0003305
35%	65%	0.0003296
40%	60%	0.0003291
45%	55%	0.000332

Tabel 1 di atas ini, menjelaskan hasil pengujian nilai MAPE terkecil yaitu dengan nilai *test size* 40% dengan MAPE sebesar 0.0003291. Maka nilai *test size* tersebut yang digunakan untuk prediksi jumlah penumpang kereta api kelas Ekonomi.

Tabel 2. Hasil Perbandinga Nilai *Test Size* Eksekutif

<i>Testing</i>	<i>Training</i>	MAPE
10%	90%	0.16058
15%	85%	0.1079
20%	80%	0.12131
25%	75%	0.12925
30%	70%	0.10792
35%	65%	0.09275
40%	60%	0.08139
45%	55%	0.07259
50%	50%	0.0659
55%	45%	0.0601
60%	40%	0.17811
65%	35%	0.17537

Tabel 2 di atas, dapat disimpulkan bahwa hasil pengujian nilai MAPE terkecil yaitu dengan nilai *test size* 55% dengan MAPE sebesar 0.0601. Maka nilai *test size* tersebut yang digunakan untuk prediksi jumlah penumpang kereta api kelas Eksekutif.

3.2. Pengujian Nilai Parameter

Pengujian nilai parameter menggunakan algoritma *Grid Search*. Parameter yang dicari yaitu kombinasi nilai K, C, Epsilon dan gamma untuk menghasilkan model yang terbaik. Model terbaik adalah model yang memiliki nilai RMSE yang mendekati 0. Pengujian nilai parameter dibagi menjadi dua, yaitu pengujian kereta api kelas Ekonomi dan kelas Eksekutif. Berikut ini hasil pengujian nilai parameter (K, C, Gamma dan Epsilon) dari 2 jenis kelas kereta api Argo Parahyangan.

3.2.1 Pengujian Nilai cLR

Pengujian menggunakan target prediksi pada hari Kamis tanggal 26 Desember 2019. Pengujian dilakukan untuk mendapatkan nilai Gamma terbaik sehingga bisa menghasilkan nilai evaluasi yang terbaik. Ada 9 parameter Gamma yang akan diuji, dan nilai parameter lain akan ditentukan terlebih dahulu. Untuk nilai K, C dan Epsilon akan menggunakan nilai terbaik yang sudah didapatkan, yaitu untuk nilai K=15, C=1000, dan Epsilon=0.01.

Berdasarkan pengujian yang sudah ditunjukkan pada Tabel 3, nilai cLR terbaik ada pada nilai 1.3 yang mempunyai nilai RMSE 0.1. Pada Gambar 4 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.

Tabel 3. Hasil Perbandingan Nilai cLR

No.	Nilai cLR	Nilai RMSE
1	0.1	189.822
2	0.3	178.512
3	0.5	133.682
4	0.7	63.969
5	0.9	19.678
6	1	10.103
7	1.2	0.591
8	1.3	0.1
9	1.5	0.10002



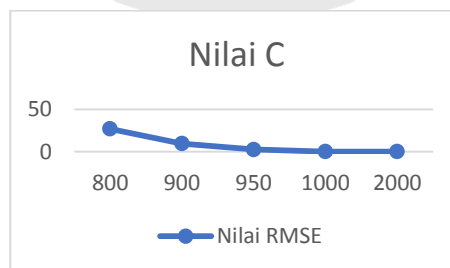
Gambar 4. Grafik Pengujian Nilai cLR

3.2.2 Pengujian Nilai C

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 5 parameter nilai. Untuk nilai K, Gamma, dan Epsilon akan menggunakan nilai terbaik yang sudah didapatkan, yaitu untuk nilai K=15, Gamma=1.3, dan Epsilon=0.01.

Tabel 4. Hasil Pengujian Nilai C

No.	C	Nilai RMSE
1	800	27.051
2	900	9.621
3	950	2.517
4	1000	0.100022
5	2000	0.100024



Gambar 5. Grafik Pengujian Nilai C

Parameter *Complexity* merupakan batas penalti toleransi terhadap kesalahan sebuah peramalan. Berdasarkan hasil pengujian parameter nilai C yang sudah dilakukan, ditunjukkan pada Tabel 4

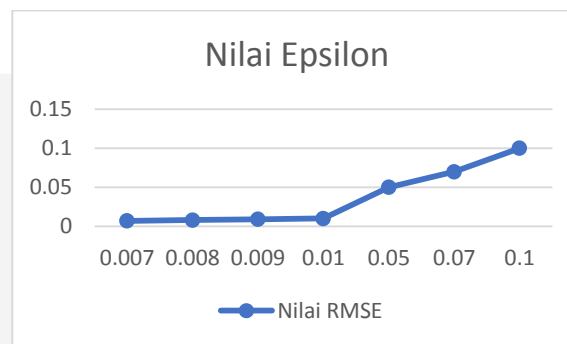
bahwa nilai C paling optimal adalah 1000 yang mempunyai nilai evaluasi RMSE sebesar 0,100022. Gambar 5 menunjukkan bahwa pada nilai C 800 cenderung mempunyai nilai RMSE yang tinggi, lalu grafik menurun dan terus mengalami penurunan sampai pada nilai C 1000. Pada nilai C 1000 grafik terlihat konvergen dan tidak mengalami perubahan lagi. Hal ini sesuai dengan pernyataan yang menyatakan bahwa semakin besar nilai *Complexity* maka semakin menjadikan model peramalan semakin tidak mentoleransi kesalahan, sehingga memberikan nilai peramalan yang baik[9].

3.2.3 Pengujian 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 7 parameter nilai. Untuk jumlah iterasi menggunakan iterasi 15, sedangkan untuk nilai lainnya menggunakan nilai terbaik yang sudah didapatkan.

Tabel 5. Hasil Pengujian Nilai Epsilon

No.	Epsilon	Nilai RMSE
1	0.007	0.00702
2	0.008	0.008071
3	0.009	0.00898
4	0.01	0.01001
5	0.05	0.05
6	0.07	0.07003
7	0.1	0.10002



Gambar 6. Grafik Pengujian 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. Berdasarkan pengujian nilai Epsilon yang sudah dilakukan, terlihat pada Tabel 5, bahwa nilai epsilon terbaik didapatkan pada nilai Epsilon 0,01 yang memiliki nilai evaluasi sebesar 0,01001. Gambar 6 menunjukkan bahwa nilai Epsilon kecil cenderung konstan, lalu pada nilai epsilon = 0,05 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.

3.3 Hasil Akurasi Menggunakan SVR

Penjelasan pada tabel 6 yaitu perbandingan nilai akurasi pada program atau aplikasi. Nilai MSE dan MAPE untuk mengukur nilai seberapa bagus metode SVR, dari tabel tersebut memiliki nilai *error* yang kecil. Arti kecil karena nilai *error* di bawah nilai 0 dari range 100. Secara umum, menunjukkan performa model yang bagus untuk prediksi.

Berikut ini adalah hasil nilai prediksi pada 6 hari jika diberikan input tahun prediksi 2019 dan bulan prediksi di bulan Desember akhir dengan performansi nilai MSE dan MAPE dengan metode SVR sebagai berikut:

Tabel 6. Hasil Akurasi Menggunakan SVR

Data Aktual	Ekonomi			Data Aktual	Eksekutif		
	Data Prediksi	MSE	MAPE		Data Prediksi	MSE	MAPE
2762	2826	0.000100	0.00033	3975	3927	248.40	0.0601
2576	2525	58	2	3658	3705	3	0.0601
3180	3147			4043	4060		
2543	2509			3210	3283		
2416	2370			3449	3456		
2882	2827			4088	4003		

4. Kesimpulan

Berdasarkan hasil pengujian yang sudah dilakukan dapat ditarik kesimpulan bahwa pada kasus Prediksi Jumlah Penumpang dan Penambahan Gerbong menghasilkan nilai *error* 0.0003291 dan 0.0601 dengan parameter nilai $cLR = 1.3$, $complexity = 1000$, $epsilon = 0.01$ untuk kereta kelas Ekonomi dan nilai $cLR=3.2$, $complexity=10000$, $epsilon 0.1$ untuk kereta kelas Eksekutif, serta jumlah iterasi sebanyak 15 menggunakan *Kernel* RBF. Hasil ini terbilang sangat baik karena hasil evaluasi sangat dekat dengan angka 0.

5. Saran

Berdasarkan hasil penelitian Tugas Akhir ini, maka penulis menyarankan untuk pengembangan penelitian selanjutnya yaitu:

1. Menambahkan lebih banyak *Data Training* / *Data Testing* yang terdapat dari PT. KAI DAOP 2 Bandung.
2. Dapat dikembangkan lagi dengan menghubungkan data seluruh kereta api keberangkatan dari Stasiun Bandung

Daftar Pustaka:

- [1] I. Purnamasari, "Hubungan Struktur Sistem Pengendalian Manajemen dan Proses Sistem Pengendalian Manajemen dengan Kinerja Keuangan Perusahaan pada PT . Kereta Api Indonesia (Persero) (*The Correlation between the Structure of Management Control System and Process of Mana*," *Fokus Ekon.*, vol. 4, no. 1, pp. 27–43, 2009.
- [2] Q. Huang, J. Mao, and Y. Liu, "An Improved Grid Search Algorithm of SVR Parameters Optimization," *Int. Conf. Commun. Technol. Proceedings, ICCT*, no. 2, pp. 1022–1026, 2012.
- [3] R. Primartha, *Belajar Machine Learning Teori dan Praktik*. Bandung: Informatika, 2018.
- [4] S. Shataee, S. Kalbi, A. Fallah, and D. Pelz, "Forest Attribute Imputation Using Machine-Learning Methods and ASTER Data: Comparison of k-NN, SVR and Random Forest Regression Algorithms," *Int. J. Remote Sens.*, vol. 33, no. 19, pp. 6254–6280, 2012.
- [5] D. Nguyen, "Learning Data Science - Predict Stock Price with Support Vector Regression (SVR)," *IT Next Organizer*, 1 Maret 2016.
- [6] L. Septiningrum, J. Statistika, F. Sains, D. A. N. Matematika, and U. Diponegoro, "Prediksi Indeks Harga Saham Gabungan Menggunakan Support Vector Regression Dengan Algoritma Grid Search", 2015.
- [7] H. Yasin, R. E. Caraka, Tarno, and A. Hoyyi, "Prediction of Crude Oil Prices Using Support Vector Regression (SVR) With Grid Search - Cross Validation Algorithm," *Glob. J. Pure Appl. Math.*, vol. 12, no. 4, pp. 3009–3020, 2016.
- [8] R. E. Caraka, "Peramalan Crude Palm Oil (CPO) Menggunakan Support Vector Regression Kernel Radial Basis," vol. 7, no. 1, pp. 43–57, 2017.
- [9] Furi, R. P., Jondri & Saepudin, D., "Peramalan Financial Time Series Menggunakan Independent Component Analysis dan Support Vector Regression (Studi Kasus: IHSG dan JII), 2015.