

Penerapan Stochastic Gradient Descent Support Vector Regression pada Data Laju Pertumbuhan Produk Domestik Bruto di Indonesia

Khamid Muhammad Arrazaq^{1, a)}, Dewi Retno Sari Saputro^{2, b)}, Ririn Setiyowati^{3, c)}

^{1,2,3}Universitas Sebelas Maret, Surakarta.

^{a)}Penulis korespondensi: khamid.arrazaq@student.uns.ac.id

^{b)}dewiretnoss@staff.uns.ac.id

^{c)}ririnsetiyowati@staff.uns.ac.id

Abstrak. Pertumbuhan ekonomi didefinisikan sebagai peningkatan pendapatan suatu negara yang diukur dengan menggunakan data produk domestik bruto (PDB). Laju pertumbuhan PDB cenderung mengalami trend naik meskipun pada setiap periode waktunya mengalami fluktuasi. Fluktuasi ini akan memengaruhi keputusan investor dalam menanam atau menarik modal. Pada penelitian ini, dilakukan penerapan model regresi pada data laju pertumbuhan PDB untuk membantu investor dalam memahami pola pertumbuhan PDB di masa mendatang. Salah satu model regresi yang dapat digunakan adalah support vector regression dengan algoritme optimasi stochastic gradient descent (SGD-SVR). Hasil penelitian menunjukkan bahwa model SGD-SVR mampu diterapkan pada data laju pertumbuhan PDB di Indonesia. Pada tahap pelatihan, MSE yang dihasilkan sebesar 0.2805 dengan jumlah iterasi sebanyak 360 iterasi. Sedangkan, pada tahap pengujian menghasilkan MSE 0.0325

Kata Kunci : PDB, model regresi, support vector regression, SGD

PENDAHULUAN

Secara umum, pertumbuhan ekonomi didefinisikan sebagai peningkatan kemampuan dari suatu perekonomian dalam memproduksi barang dan jasa. Menurut [1] pertumbuhan ekonomi merujuk pada perubahan yang bersifat kuantitatif dan biasanya diukur dengan menggunakan data produk domestik bruto (PDB). Laju pertumbuhan ekonomi menyatakan persentase kenaikan atau penurunan pendapatan daerah yang diukur dengan cara membandingkan data PDB pada periode tertentu dengan data pada periode sebelumnya.

Produk domestik bruto (PDB) adalah jumlah nilai tambah yang dihasilkan oleh seluruh unit usaha, atau jumlah nilai barang dan jasa akhir yang dihasilkan oleh seluruh unit ekonomi pada suatu negara ² (BPS, 2023). Saat ini terdapat 9 sektor lapangan usaha yang menentukan nilai PDB tiap periodenya. Laju pertumbuhan PDB cenderung mengalami trend naik meskipun pada setiap periode waktunya mengalami fluktuasi. Fluktuasi ini akan memengaruhi keputusan investor dalam menanam atau menarik modal [3]. Oleh karena itu, perlu dilakukan penerapan model regresi pada data laju pertumbuhan PDB untuk membantu investor dalam memahami pola pertumbuhan PDB di masa mendatang.

Model regresi adalah suatu metode statistik yang digunakan untuk memodelkan hubungan antara satu atau lebih variabel bebas dan variabel terikat. Tujuan utama dari model regresi adalah untuk memahami dan menganalisis bagaimana perubahan dalam variabel bebas dapat memengaruhi variabel terikat. Dalam konteks kecerdasan buatan, model regresi dilatih untuk menyesuaikan variabel pada data melalui proses optimasi parameter menggunakan algoritme optimasi tertentu dan penentuan hyperparameter optimal [4].

Data laju pertumbuhan PDB termasuk dalam kelompok data multivariate time series berpola nonlinear. Salah satu model regresi yang dapat digunakan pada kelompok data ini adalah support vector regression (SVR). Namun, SVR memiliki kekurangan yakni diperlukannya penggunaan algoritme optimasi untuk mengestimasi parameter pada SVR [5]. Pada penelitian ini, algoritme stochastic gradient descent (SGD) digunakan untuk menyelesaikan masalah optimasi tersebut.

Penerapan model regresi SVR dengan algoritme SGD (SGD-SVR) telah digunakan pada beberapa penelitian. Xie dan Li (2019) dalam penelitiannya membandingkan pemodelan SGD-SVR, budgeted SGD (BSGD) dan sequential minimal optimization SVR (SMO-SVR) pada dataset regresi milik LIBSVM. Hasil penelitian mereka menunjukkan bahwa algoritme SGD-SVR memiliki nilai mean squared error (MSE) paling rendah. Selanjutnya, [6] dalam penelitiannya membandingkan pemodelan SGD-SVR dengan accurate online SVR (AO-SVR) pada data historis penerbangan. Hasil penelitiannya menunjukkan bahwa algoritme SGD-SVR memiliki nilai mean absolute error (MAE) paling rendah.

Berdasarkan uraian tersebut, pada penelitian ini dilakukan penerapan model SGD-SVR pada data laju pertumbuhan PDB di Indonesia. Selain itu, penelitian ini juga bermanfaat untuk bahan evaluasi bagi pemerintah atau pihak lain yang ingin melakukan penelitian lebih lanjut.

METODOLOGI PENELITIAN

Pengumpulan Data

Data pada penelitian ini adalah data sekunder dari [2] berupa data kuartal dari kuartal 2 tahun 2010 hingga kuartal 2 tahun 2023 yang berjumlah 53 data. Data yang digunakan dalam penelitian ini terdiri dari 9 variabel bebas dan 1 variabel terikat yang mengacu pada perhitungan PDB berdasarkan 9 sektor lapangan usaha utama (BPS, 2023). Variabel-variabel yang digunakan pada data ini ditunjukkan oleh Tabel 1.

Tabel 1: Variabel penelitian

Variabel	Keterangan (kenaikan pendapatan dalam %)
X_1	Pertanian, Kehutanan, dan Perikanan
X_2	Pertambangan dan Penggalian
X_3	Industri Pengolahan
X_4	Listrik, Gas, dan Air
X_5	Konstruksi
X_6	Perdagangan, dan Akomodasi
X_7	Pengangkutan, dan Komunikasi
X_8	Keuangan, Persewaan, dan Jasa Perusahaan
X_9	Jasa Pelayanan Pemerintah dan Jasa-jasa lainnya
Y	Produk Domestik Bruto

Sumber: Badan Pusat Statistik

Adapun statistika deskriptif untuk setiap variabel pada data ditunjukkan oleh Tabel 2.

Tabel 2: Statistika deskriptif data

Variabel	Rata-rata	Variansi	Minimal	Median	Maksimal
X_1	2.04	188.18	-22.95	8.50	16.36
X_2	0.52	6.17	-5.93	0.56	4.45
X_3	0.97	3.23	-6.49	0.75	5.22
X_4	1.19	10.71	-7.69	1.60	7.73
X_5	1.28	16.47	-7.37	3.37	5.72
X_6	1.17	7.28	-9.67	1.32	7.14
X_7	2.04	6.52	-10.21	2.01	10.16
X_8	1.38	3.24	-7.89	1.75	4.01
X_9	1.60	38.02	-10.61	2.52	13.28

Y	1.22	6.33	-4.19	1.05	5.05
---	------	------	-------	------	------

Berdasarkan Tabel 2 diketahui bahwa 9 variabel bebas X_1 hingga X_9 memiliki variansi yang tidak jauh berbeda, yang berarti tidak diperlukan proses normalisasi data. Diketahui pula data tidak memiliki *missing values* serta *outlier* sehingga tidak diperlukan proses pembersihan data. Selanjutnya, data dibagi menjadi data pelatihan dan data pengujian pada perbandingan 70%:30%, 80%:20%, dan 90%:10%.

Metode yang Digunakan

Support Vector Regression (SVR)

SVR merupakan suatu model regresi pada *supervised machine learning* (SML), yakni suatu pendekatan kecerdasan buatan yang melatih model untuk mengenali pola pada data berlabel [4]. Tujuan dari SVR yakni menemukan *hyperplane* (garis pemisah) berupa fungsi regresi $f(x)$ yang menyesuaikan titik data semirip mungkin. Dalam SVR, dikenal juga istilah *hyperparameter* yakni suatu parameter yang nilainya perlu ditentukan sebelum tahap pelatihan dimulai untuk mempercepat waktu pelatihan dan meningkatkan akurasi peramalan. SVR bekerja dengan meminimalkan permasalahan optimasi yang ditulis sebagai

$$\min \left\{ \frac{1}{2} \sum_{i,j=1}^n (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x_i, x_j) + \varepsilon \sum_{i=1}^n (\alpha_i + \alpha_i^*) - \sum_{i=1}^n y_i(\alpha_i - \alpha_i^*) \right\} \quad (1)$$

dengan batasan

$$\sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0 \text{ dan } 0 \leq \alpha_i, \alpha_i^* \leq C$$

dengan:

- ε : *hyperparameter* yang menentukan lebar batas margin
- C : *hyperparameter* yang mengatur keseimbangan antara penyesuaian data (*data fitting*) dan kesederhanaan model.
- α_i, α_i^* : pengali lagrange (*lagrange multiplier*) yang mengatur besarnya bobot pada titik di dalam margin (*support vector*)
- $K(x_i, x_j)$: fungsi kernel berdefinit positif
- y_i : variabel terikat pada data

Dari persamaan (1) dibentuk fungsi regresi SVR yang ditulis sebagai

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*)K(x_i, x) \quad (2)$$

Fungsi kernel K pada persamaan (1) digunakan untuk mempermudah perhitungan *inner product* antar pasangan titik data dalam ruang fitur (*feature space*) tanpa perlu menghitung koordinat titik dalam ruang tersebut [7]. Pada penelitian ini, digunakan kernel *radial basis function* (RBF) yang ditulis sebagai

$$K(x, x') = \exp(-\gamma \|x - x'\|^2) \quad (3)$$

dengan $\|x - x'\|^2$ adalah jarak *euclidean* kuadrat antara vektor input $x, x' \in R^d$ dan γ merupakan *hyperparameter* yang nilainya ditentukan dengan persamaan

$$\gamma = \frac{1}{n(x) \times \text{var}(x)} \quad (4)$$

dengan $n(x)$ adalah jumlah variabel bebas pada data dan $\text{var}(x)$ adalah nilai variansi dari data.

Stochastic Gradient Descent (SGD)

Masalah optimasi pada persamaan (1) dapat diselesaikan dengan menggunakan algoritme optimasi. Salah satu algoritme optimasi yang mudah diimplementasikan yakni *stochastic gradient descent* (SGD). SGD merupakan suatu

metode iterasi yang sering digunakan untuk mencari titik minimum fungsi [8]. Algoritme SGD diterapkan dalam SVR dengan langkah-langkah sebagai berikut.

1. Inisiasi nilai pengali lagrange α , nilai *hyperparameter learning rate* λ , dan derajat toleransi *tol*
2. Menghitung nilai gradien (turunan) fungsi objektif (L) pada persamaan (1) terhadap pengali lagrange α yang ditulis sebagai

$$\begin{aligned}\frac{\partial L}{\partial \alpha_i} &= \sum_{j=1}^n (\alpha_j - \alpha_j^*) K(x_i, x_j) - y_i + \varepsilon \\ \frac{\partial L}{\partial \alpha_i^*} &= y_i - \sum_{j=1}^n (\alpha_j - \alpha_j^*) K(x_j, x_i) + \varepsilon\end{aligned}\quad (5)$$

3. Memperbarui nilai α dengan mengurangi nilai α lama terhadap perkalian antara *learning rate* λ dan gradien fungsi objektif L (Admojo & Sulistya, 2022) yang ditulis sebagai

$$\begin{aligned}\alpha_i(\text{baru}) &= \alpha_i(\text{lama}) - \lambda \frac{\partial L}{\partial \alpha_i} \\ \alpha_i^*(\text{baru}) &= \alpha_i^*(\text{lama}) - \lambda \frac{\partial L}{\partial \alpha_i^*}\end{aligned}\quad (6)$$

4. Apabila kondisi pemberhentian belum terpenuhi, maka lakukan langkah 2 dan 3. Kondisi pemberhentian terpenuhi jika $0 \leq \sum_{i=1}^n (\alpha_i - \alpha_i^*) \leq$ derajat toleransi (*tol*) sudah tercapai. Apabila kondisi pemberhentian sudah terpenuhi maka langkah iterasi berhenti.

Dari langkah iterasi tersebut, diperoleh pengali lagrange α optimal yang digunakan untuk menghitung nilai peramalan pada tahap pengujian.

Metode yang Digunakan

Support Vector Regression (SVR)

SVR merupakan suatu model regresi pada *supervised machine learning* (SML), yakni suatu pendekatan kecerdasan buatan yang melatih model untuk mengenali pola pada data berlabel [4]. Tujuan dari SVR yakni menemukan *hyperplane* (garis pemisah) berupa fungsi regresi $f(x)$ yang menyesuaikan titik data semirip mungkin. Dalam SVR, dikenal juga istilah *hyperparameter* yakni suatu parameter yang nilainya perlu ditentukan sebelum tahap pelatihan dimulai untuk mempercepat waktu pelatihan dan meningkatkan akurasi peramalan. SVR bekerja dengan meminimalkan permasalahan optimasi yang ditulis sebagai

$$\min \left\{ \frac{1}{2} \sum_{i,j=1}^n (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j) + \varepsilon \sum_{i=1}^n (\alpha_i + \alpha_i^*) - \sum_{i=1}^n y_i (\alpha_i - \alpha_i^*) \right\} \quad (1)$$

dengan batasan

$$\sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0 \text{ dan } 0 \leq \alpha_i, \alpha_i^* \leq C$$

dengan:

- ε : *hyperparameter* yang menentukan lebar batas margin
- C : *hyperparameter* yang mengatur keseimbangan antara penyesuaian data (*data fitting*) dan kesederhanaan model.
- α_i, α_i^* : pengali lagrange (*lagrange multiplier*) yang mengatur besarnya bobot pada titik di dalam margin (*support vector*)
- $K(x_i, x_j)$: fungsi kernel berdefinit positif
- y_i : variabel terikat pada data

Dari persamaan (1) dibentuk fungsi regresi SVR yang ditulis sebagai

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(x_i, x) \quad (2)$$

Fungsi kernel K pada persamaan (1) digunakan untuk mempermudah perhitungan *inner product* antar pasangan titik data dalam ruang fitur (*feature space*) tanpa perlu menghitung koordinat titik dalam ruang tersebut (Xie & Li, 2019). Pada penelitian ini, digunakan kernel *radial basis function* (RBF) yang ditulis sebagai

$$K(x, x') = \exp(-\gamma \|x - x'\|^2) \quad (3)$$

dengan $\|x - x'\|^2$ adalah jarak *euclidean* kuadrat antara vektor input $x, x' \in R^d$ dan γ merupakan *hyperparameter* yang nilainya ditentukan dengan persamaan

$$\gamma = \frac{1}{n(x) \times \text{var}(x)} \quad (4)$$

dengan $n(x)$ adalah jumlah variabel bebas pada data dan $\text{var}(x)$ adalah nilai variansi dari data.

Stochastic Gradient Descent (SGD)

Masalah optimasi pada persamaan (1) dapat diselesaikan dengan menggunakan algoritme optimasi. Salah satu algoritme optimasi yang mudah diimplementasikan yakni *stochastic gradient descent* (SGD). SGD merupakan suatu metode iterasi yang sering digunakan untuk mencari titik minimum fungsi [8]. Algoritme SGD diterapkan dalam SVR dengan langkah-langkah sebagai berikut.

1. Inisiasi nilai pengali lagrange α , nilai *hyperparameter learning rate* λ , dan derajat toleransi *tol*
2. Menghitung nilai gradien (turunan) fungsi objektif (L) pada persamaan (1) terhadap pengali lagrange α yang ditulis sebagai

$$\frac{\partial L}{\partial \alpha_i} = \sum_{j=1}^n (\alpha_j - \alpha_j^*) K(x_i, x_j) - y_i + \varepsilon$$

$$\frac{\partial L}{\partial \alpha_i^*} = y_i - \sum_{j=1}^n (\alpha_j - \alpha_j^*) K(x_j, x_i) + \varepsilon \quad (5)$$

3. Memperbarui nilai α dengan mengurangi nilai α lama terhadap perkalian antara *learning rate* λ dan gradien fungsi objektif L (Admojo & Sulistya, 2022) yang ditulis sebagai

$$\alpha_i(\text{baru}) = \alpha_i(\text{lama}) - \lambda \frac{\partial L}{\partial \alpha_i}$$

$$\alpha_i^*(\text{baru}) = \alpha_i^*(\text{lama}) - \lambda \frac{\partial L}{\partial \alpha_i^*} \quad (6)$$

4. Apabila kondisi pemberhentian belum terpenuhi, maka lakukan langkah 2 dan 3. Kondisi pemberhentian terpenuhi jika $0 \leq \sum_{i=1}^n (\alpha_i - \alpha_i^*) \leq$ derajat toleransi (*tol*) sudah tercapai. Apabila kondisi pemberhentian sudah terpenuhi maka langkah iterasi berhenti.

Dari langkah iterasi tersebut, diperoleh pengali lagrange α optimal yang digunakan untuk menghitung nilai peramalan pada tahap pengujian.

Mean Squared Error (MSE)

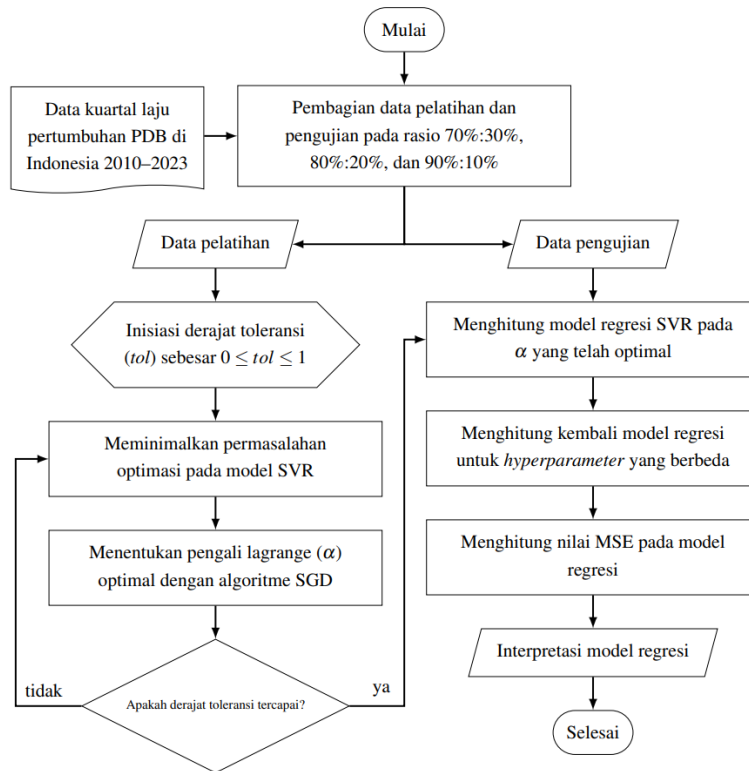
Pada penelitian ini, digunakan metrik *mean squared error* (MSE) untuk memeriksa seberapa dekat nilai peramalan dengan nilai aktualnya. Semakin rendah MSE, semakin dekat nilai peramalan dengan nilai aktualnya. MSE dituliskan sebagai

$$MSE = \sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n} \quad (7)$$

dengan y_i merupakan nilai aktual pada data ke- i , \hat{y}_i merupakan nilai peramalan pada data ke- i , dan n merupakan jumlah data.

Prosedur dan Implementasi

Langkah penelitian yang dilakukan untuk mencapai tujuan penelitian ditunjukkan oleh Gambar 1.



Gambar 1: Diagram alir penelitian

HASIL DAN PEMBAHASAN

Di sini kami memberikan beberapa saran dasar untuk memformat matematika Anda, tetapi kami tidak mencoba untuk mendefinisikan gaya atau spesifikasi rinci untuk pengaturan huruf matematika. Anda harus menggunakan gaya, simbol, dan konvensi standar untuk bidang/disiplin yang Anda tulis.

Penerapan Model SGD-SVR

Ditentukan nilai hyperparameter $\varepsilon=0.1$, $C=1$, $\gamma=0.0036$, learning rate (λ)=0.01, dan derajat toleransi (tol)=0.001. Kemudian, dilakukan penerapan model SGD-SVR pada data pelatihan dengan rasio pembagian sebesar 80% untuk data pelatihan dan 20% untuk data pengujian. Nilai fungsi objektif dan batasan pada persamaan (1) setiap iterasinya ditunjukkan oleh Tabel 3.

Tabel 3: Nilai fungsi objektif dan batasan setiap iterasi

Iterasi	Fungsi Objektif	Batasan
1	-5.7401	1.1528

50	-28.0557	0.1975
100	-30.2872	-0.0530
150	-30.7830	0.0281
199	-30.9134	0.0006

Berdasarkan Tabel 3, didapatkan nilai fungsi objektif minimal sebesar -30.9134 dengan pengulangan sebanyak 199 iterasi.

Eksplorasi Hyperparameter Model Regresi

Pada tahap ini, dilakukan eksplorasi *hyperparameter* pada tahap pelatihan model regresi. Proses ini bertujuan untuk mencari nilai MSE minimal serta jumlah iterasi terkecil untuk menyingkat waktu pelatihan model. Data penelitian yang berjumlah 53 data dibagi menjadi data pelatihan dan data pengujian. Hasil kombinasi pembagian data dan penentuan *hyperparameter* terbaik ditunjukkan oleh Tabel 4.

Tabel 4: Hasil kombinasi pembagian data dan *hyperparameter*

Pembagian data	Hyperparameter		MSE
	C	ϵ	
70% dan 30%	1	0.05	0.5159
		0.1	0.5083
		0.5	0.5433
	3	0.05	0.3094
		0.1	0.3006
		0.5	0.3483
80% dan 20%	1	0.05	0.1551
		0.1	0.1049
		0.5	0.1629
	3	0.05	0.0867
		0.1	0.0836
		0.5	0.0848
90% dan 10%	1	0.05	0.0604
		0.1	0.0766
		0.5	0.1887
	3	0.05	0.0413
		0.1	0.0382
		0.5	0.0761

Berdasarkan Tabel 4, pembagian data yang menghasilkan nilai MSE minimal yakni 90% atau 47 data untuk pelatihan dan 10% atau 6 data untuk pengujian. Dari Tabel 4.8 diketahui 3 kombinasi *hyperparameter* C dan ϵ terbaik yakni [1, 0.05], [3, 0.05], dan [3, 0.1]. Kemudian dilakukan penentuan *hyperparameter learning rate* (λ) terbaik yang menghasilkan nilai MSE minimal. Nilai λ yang diujikan adalah 0.005, 0.01, dan 0.02. Hasil pengujian *hyperparameter* ini ditunjukkan oleh Tabel 5.

Tabel 5: Hasil kombinasi *hyperparameter* C , ϵ , dan λ

Hyperparameter			MSE
C	ϵ	λ	
1	0.05	0.005	0.0644
		0.01	0.0604
		0.02	0.1052
3	0.05	0.005	0.0351

	0.01	0.0413
	0.02	0.0413
	0.005	0.0383
0.1	0.01	0.0382
	0.02	0.0325

Dari Tabel 5, 3 kombinasi *hyperparameter* C , ε , dan *learning rate* (λ) terbaik yakni $[3, 0.05, 0.005]$, $[3, 0.1, 0.01]$, dan $[3, 0.1, 0.02]$. Selanjutnya dilakukan penentuan derajat toleransi (*tol*) terbaik. Nilai *tol* yang diujikan adalah 0.01, 0.001, dan 0.0001. Hasil pengujian *tol* ini ditunjukkan oleh Tabel 6.

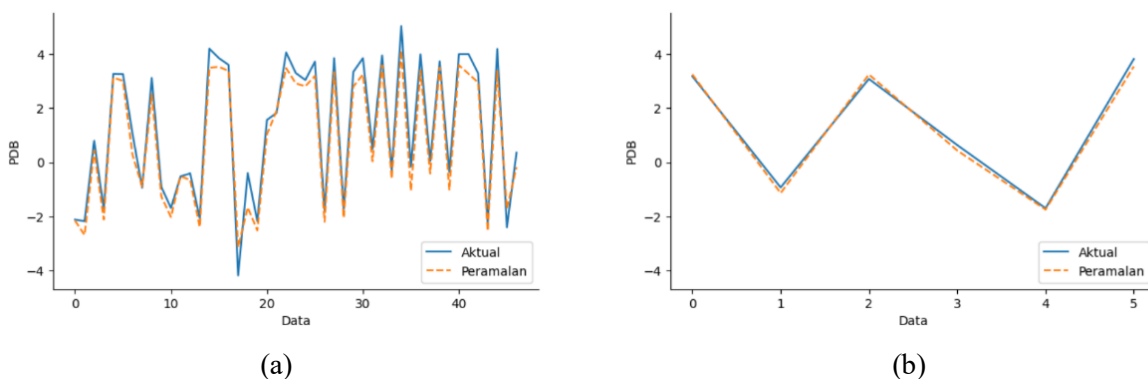
Tabel 6: Hasil kombinasi *hyperparameter* dan *tol*

Hyperparamete			<i>tol</i>	epoch	MSE
C	ε	λ			
3	0.05	0.005	0.01	64	0.0536
			0.001	1000	0.0351
			0.0001	1000	0.0351
	0.1	0.01	0.01	76	0.0796
			0.001	239	0.0382
			0.0001	880	0.0353
3	0.1	0.02	0.01	113	0.0385
			0.001	360	0.0325
			0.0001	439	0.0353

Berdasarkan Tabel 6, *hyperparameter* C , ε , λ , dan derajat toleransi (*tol*) terbaik adalah $[3, 0.1, 0.02, 0.001]$ dengan nilai MSE 0.0325 dan jumlah iterasi sebanyak 360 iterasi.

Interpretasi Model Regresi

Model yang sudah dilatih dengan *hyperparameter* terbaik kemudian diuji performanya. Uji performa pada model dilakukan dengan membandingkan nilai MSE pada data pelatihan dan data pengujian untuk menguji kinerja model regresi dalam mengatasi overfitting. Menurut [9], overfitting adalah suatu kondisi yang terjadi dalam tahap pelatihan model ML dimana model tidak mampu melakukan generalisasi dengan baik pada data pengujian. Ciri-ciri dari overfitting pada model ML yakni, nilai MSE yang sangat rendah pada data pelatihan, namun sangat tinggi pada data pengujian. Beberapa faktor yang menyebabkan terjadinya overfitting adalah data pelatihan yang terlalu sedikit, serta jumlah data dengan nilai yang salah (noise) yang banyak pada data [9]. Perbandingan nilai aktual dan nilai peramalan pada data ditunjukkan oleh Gambar 2.



Gambar 2: Perbandingan nilai aktual dan peramalan pada (a) tahap pelatihan, dan (b) tahap pengujian

Diperoleh nilai MSE model SGD-SVR pada data pelatihan sebesar 0.2805, dan pada data pengujian sebesar 0.0325, menyebabkan penerapan model SGD-SVR tidak termasuk kategori *overfitting*. Kemudian,, berdasarkan Gambar 2a dan Gambar 2b, diketahui bahwa model regresi mampu mempelajari data dengan cukup baik. Dengan demikian, model SGD-SVR mampu diterapkan pada data laju pertumbuhan PDB di Indonesia.

KESIMPULAN

Berdasarkan hasil dan pembahasan, diperoleh kesimpulan yakni model regresi SVR dengan algoritme SGD dapat diterapkan pada data laju pertumbuhan PDB di Indonesia. Pada tahap pelatihan data menghasilkan MSE 0.2805 dengan jumlah iterasi sebanyak 360 iterasi. Sedangkan, pada tahap pengujian data menghasilkan MSE 0.0325, menyebabkan penerapan model pada data penelitian tidak termasuk kategori *overfitting*.

REFERENSI

-
- [1] Weran, B. A. (2017). Pengaruh Pertumbuhan Ekonomi dan Pertumbuhan Penduduk Terhadap Tingkat Kemiskinan Kabupaten Flores Timur. *Economics Bosowa*, 3(3), 11-23
- [2] Badan Pusat Statistik. (2023). "Produk Domestik Bruto Berdasarkan Lapangan Usaha". Retrieved from <https://www.bps.go.id/subject/11/produk-domestik-bruto--lapangan-usaha.html>
- [3] Tambunan, R. S., Yusuf, Y., & Mayes, A. (2015). "Pengaruh kurs, inflasi, libor dan PDB terhadap foreign direct investment (FDI) di Indonesia". *Jurnal Ekonomi*, 23(1).
- [4] IBM Cloud Education. (2023). "What is Supervised Learning?". Retrieved from <https://www.ibm.com/id-en/topics/supervised-learning>
- [5] Jiang, Pingan & Zhou, Xuelin & Liu, Tonglai & Guo, Xiaohu & Ma, Deying & Zhang, Cong & Li, Yan & Liu, Shuangyin. (2023). "Prediction Dynamics in Cotton Aphid Using Unmanned Aerial Vehicle Multispectral Images and Vegetation Indices". *IEEE Access*. pp.1-1. 10.1109/ACCESS.2023.3235912
- [6] Ren, L.-H., Ye, Z.-F., & Zhao, Y.-P. (2020). "A modeling method for aero-engine by combining stochastic gradient descent with support vector regression". *Aerospace Science and Technology*, 99, 105775
- [7] Xie, Z., & Li, Y. (2019). "Large-scale support vector regression with budgeted stochastic gradient descent". *International Journal of Machine Learning and Cybernetics*, 10, 1529–1541.
- [8] Admojo, F. T., & Sulistya, Y. I. (2022). "Analisis performa algoritma Stochastic Gradient Descent (SGD) dalam mengklasifikasi tahu berformalin". *Indonesian Journal of Data and Science*, 3(1), 1-8.
- [9] Ying, X. (2019). "An overview of overfitting and its solutions". *Journal of Physics: Conference Series*. IOP Publishing