

Prediksi *Financial Time Series* Menggunakan *Independent Component Analysis* dan *Support Vector Regression* Studi Kasus : IHSG dan JII

Ratih Puspita Furi¹, Jondri, M.Si.², Deni Saepudin, Dr³

Fakultas Informatika Telkom University, Bandung

Abstrak

Data indeks harga saham merupakan salah satu contoh data *financial time series* yang sifatnya cenderung berubah-ubah dan mengandung *noise*. Sifat *noise* ini terjadi saat data mengandung sedikit informasi yang dibutuhkan (*less information*). Hal ini tentunya akan mempengaruhi prediksi nilai indeks harga saham. Oleh karena itu dibutuhkan identifikasi dan penghapusan *noise* menggunakan metode *Independent Component Analysis* (ICA), sebelum membangun sistem untuk memprediksi nilai indeks harga saham menggunakan *Support Vector Regression* (SVR).

ICA merupakan teknik baru untuk pemrosesan sinyal statistik. Dengan menggunakan metode ICA, identifikasi *independent component* (ICs) dan penghapusan ICs yang mengandung *noise* dapat dilakukan. ICs yang mengandung banyak informasi (*most information*) akan digunakan sebagai input pada SVR untuk membangun sistem untuk memprediksi nilai *closing* indeks harga saham.

Data *financial time series* yang digunakan adalah data Indeks Harga Saham Gabungan (IHSG) dan Jakarta Islamic Index (JII).

Kata Kunci: *financial time series, indeks harga saham, Independent Component Analysis, Support Vector Regression.*

Abstract

Stock price index data is an example of financial time series data which tends to change and contain noise. Such noise happens when the data contains less needed information. Thus can influence the stock price index prediction. Therefore, noise identification and noise removal using Independent Component Analysis (ICA) method are needed, before constructing model for stock price index prediction using Support Vector Regression (SVR).

ICA is a new technique to process statistical signal. Independent Components (ICs) with less information can be identified and removed by using ICA method. ICs with most information will be used as inputs in SVR to build sistem for predicting the closing price of stock index.

Financial time series data used in this project are taken from Indeks Harga Saham Gabungan (IHSG) and Jakarta Islamic Index (JII).

Keywords : *financial time series, stock price index, Independent Component, Support Vector Regression.*

I. Pendahuluan

1.1 Latar Belakang

Di bursa efek terdapat indeks harga saham yang dijadikan tolak ukur suatu harga saham yang diperdagangkan, seperti IHSG dan JII. Indeks harga saham dapat digunakan sebagai indikator *trend* pasar yang menggambarkan kondisi pasar perekonomian yang sedang berlangsung. Indeks harga saham juga digunakan investor sebagai indikator penting untuk mengambil keputusan dalam berinvestasi. Oleh karena itu, mengetahui prediksi nilai indeks harga saham menjadi sesuatu yang sangat penting.

Data indeks harga saham termasuk data *financial time series*. Data *financial time series* merupakan data yang sifatnya berubah-ubah dan *noisy*. Sifat *noise* ini disebabkan oleh tidak lengkapnya ketersediaan informasi yang dibutuhkan pada data tersebut untuk mengetahui hubungan antara *past price* dan *future price* [1]. Hal ini dikarenakan perilaku dari pasar keuangan, seperti pembelian atau penjualan dalam jumlah yang besar, hingga pengaruh kondisi suatu negara. Sehingga menyebabkan anomali pada data indeks saham.

Support Vector Machine (SVM) merupakan algoritma terbaru dari Neural Network, yang berdasar pada teori statistik. Dalam pengaplikasiannya, SVM memiliki potensi yang besar dan performansi yang sangat bagus. SVM memiliki prinsip *risk minimization* yang memiliki kemampuan generalisasi yang cukup baik. SVM membangun sebuah *hyperplane* yang memisahkan dua kelas, yang dapat diperluas untuk kasus *multiclass*. Pemisahan kelas dengan margin yang besar dapat meminimalkan *error* yang terjadi saat melakukan generalisasi solusi, sehingga SVM mampu menghasilkan solusi yang unik. [1].

Support Vector Regression (SVR) merupakan model regresi dari SVM. Metode SVR digunakan untuk memecahkan permasalahan non-linear. SVR sukses diterapkan di beberapa permasalahan dalam prediksi *time series*. Hal ini yang mendorong penelitian ini menggunakan SVR untuk memprediksi nilai dari indeks harga saham [1].

Sifat *noise* dari data *financial time series* merupakan salah satu permasalahan saat menggunakan SVR untuk memprediksi nilai indeks harga saham, karena dapat menurunkan kemampuan dari SVR untuk melakukan generalisasi solusi. Sehingga poses identifikasi dan penghapusan *noise* menjadi hal yang penting [1].

Independent Component Analysis (ICA) merupakan teknik baru untuk pemrosesan sinyal statistik. ICA bertujuan untuk memisahkan sinyal sumber (*source signals*) yang tersembunyi dari sinyal campuran (*mixture signals*). ICA diterapkan pada data *financial time series* untuk melakukan identifikasi *independent components* (ICs) dan penghapusan ICs yang mengandung *noise*. Selanjutnya ICs yang tidak mengandung *noise* akan digunakan sebagai variabel prediksi, yang kemudian akan digunakan sebagai input untuk SVR [1].

1.2 Tujuan

Beberapa tujuan yang ingin dicapai antara lain :

- a. Melakukan prediksi nilai *closing* IHSG dan JII dengan mengimplementasikan metode *Independent Component Analysis* dan *Support Vector Regression*.
- b. Mengetahui perbandingan data asli dan ICs *non-noisy* dari data IHSG dan JII.
- c. Mengetahui pengaruh parameter C terhadap performansi dari sistem prediksi yang dibangun.
- d. Mengetahui pengaruh parameter *loss function* (ϵ) terhadap performansi dari sistem prediksi yang dibangun.
- e. Mengetahui pengaruh parameter σ^2 dari kernel *Radial Basis Function* (RBF) terhadap performansi dari sistem prediksi yang dibangun.
- f. Mengetahui pengaruh penggunaan metode ICA terhadap performansi dari sistem prediksi yang dibangun.

2. Landasan Teori

2.1 Independent Component Analysis (ICA)

Independent Component Analysis (ICA) merupakan sebuah teknik baru yang digunakan untuk melakukan pemrosesan sinyal statistik. ICA bertujuan untuk menemukan *independent source* dari sekumpulan data. Dalam studi ini, ICA digunakan untuk melakukan proses identifikasi dan melakukan penghapusan terhadap data yang mengandung *noise*. Dalam hal ini data yang digunakan adalah data indeks saham IHSG dan JII.

Pada ICA, sinyal campuran X dapat diekspresikan sebagai $X = AS$. Di mana A adalah matriks pencampur dan S adalah sinyal sumber yang tidak dapat langsung diamati dari sinyal campuran X .

Sinyal sumber diasumsikan saling bebas satu sama lain secara statistik. Berdasarkan asumsi tersebut, solusi untuk ICA diperoleh melalui proses mendapatkan sebuah *demixing matrix* W . Algoritma FastICA dapat digunakan untuk menyelesaikan permasalahan *demixing matrix* W . Sebelumnya pada ICA terdapat proses *centering* dan *whitening* yang merupakan bentuk *preprocessing* dari ICA. *Demixing matrix* W ini kemudian digunakan untuk mentransformasikan sinyal campuran X yang telah diobservasi untuk menghasilkan sinyal bebas Y , $Y = WX$. Baris pada matriks Y ini selanjutnya disebut *independent components* (ICs) atau komponen bebas.

ICs tersebut merupakan komponen dari *forecasting variables*. Untuk mengurutkan ICs dapat menggunakan fungsi *Relative Hamming Distance* (RHD) pada proses *Test and Acceptance* (TnA). Pada proses TnA dilakukan penghapusan terhadap ICs yang merepresentasikan *noise*, berdasarkan nilai RHD yang didapat.

Berikut persamaan RHD menurut (Cheung and Xu) :

$$RHD = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i| \tag{2.1}$$

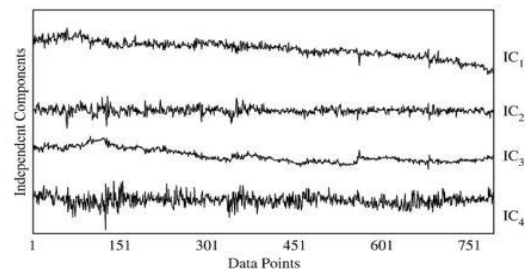
di mana,

x_i = nilai aktual; \hat{x}_i = nilai prediksi; n = total data.

Proses mengurutkan ICs ditujukan untuk mendapatkan ICs yang mengandung sedikit informasi, sebagai representasi *noise* dari sebuah data. Selanjutnya ICs lainnya yang *most information* akan digunakan sebagai variabel prediksi untuk membangun SVR.

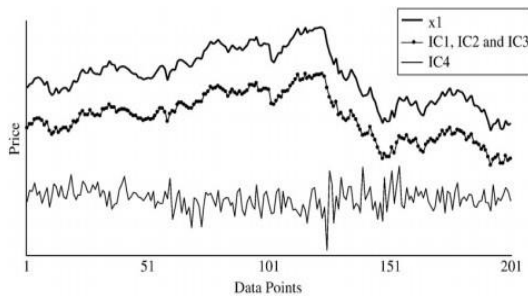
Berikut ilustrasi (Gambar 2-1 dan Gambar 2-2) mengenai konsep metode TnA dengan menggunakan RHD, berdasarkan Chi-Jie Lu [1], yang digunakan pada sistem ini. Pada Gambar (2-1), terlihat empat buah ICs dari sebuah data *time series*. Sedangkan pada Gambar (2-3), terlihat penggunaan RHD pada TnA yang digunakan untuk mengurutkan ICs. Pada Gambar (2-2) tersebut terlihat bahwa nilai RHD yang paling kecil adalah pada saat iterasi pertama, saat RHD bernilai 0.7662. Nilai RHD terkecil tersebut didapatkan dari kombinasi IC₁, IC₂, IC₃ tanpa mengikutsertakan IC₄.

Pada Gambar (2-3), dapat diketahui bahwa terdapat kemiripan yang tinggi antara IC₁, IC₂, IC₃ dengan data asli (x_1) *time series* yang digunakan. Hal ini mengindikasikan bahwa informasi yang terkandung pada ketiga IC mengenai *trend data time series* cukup banyak (*most information*). Sedangkan pada IC₄ mengindikasikan bahwa informasi yang terkandung mengenai *trend data time series* sangat sedikit, sehingga IC₄ dapat direpresentasikan sebagai *noise*. IC₄ yang direpresentasikan sebagai *noise* kemudian dihapus dari kombinasi *forecasting variable* yang digunakan.



Gambar 2-1. ICs dari data *time series*

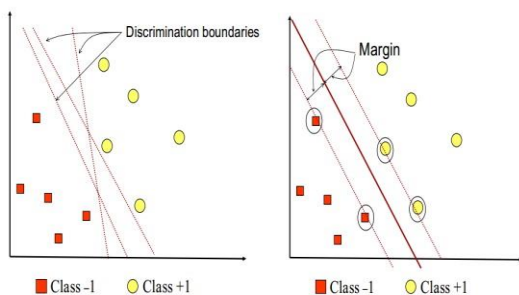
Iterations	The ICs included in the matrix (\hat{X})	RHD reconstruction error
1	IC ₁ , IC ₂ , IC ₃	0.7662
	IC ₁ , IC ₂ , IC ₄	1.7181
	IC ₁ , IC ₃ , IC ₄	1.4294
	IC ₂ , IC ₃ , IC ₄	1.8948
2	IC ₁ , IC ₂	1.7137
	IC ₁ , IC ₃	1.3346
	IC ₂ , IC ₃	1.9158
3	IC ₁	1.7531
	IC ₃	1.9779



Gambar 2-3. Perbedaan kombinasi IC₁, IC₂, dan IC₃, hanya IC₄, dan 201 data pertama (x1) dari *data time series* yang digunakan.

2.2 Support Vector Regression (SVR)

Support Vector Regression (SVR) adalah sebuah adaptasi dari teori *machine learning* sebelumnya yang digunakan dalam permasalahan klasifikasi, yakni *Support Vector Machine (SVM)*. SVR merupakan penerapan metode SVM untuk kasus regresi. SVM diterapkan untuk kasus klasifikasi dan menghasilkan output berupa bilangan bulat atau diskrit, sedangkan SVR diterapkan pada kasus regresi yang menghasilkan output berupa bilangan riil atau kontinu.



Gambar 2-4. *Hyperplane*

Konsep dari SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah kelas pada ruang input. Pada Gambar (2-4) memperlihatkan beberapa data yang merupakan anggota dari dua buah kelas. Kelas -1 disimbolkan dengan warna merah sedangkan kelas +1 berwarna kuning. Pada Gambar (2-4) sebelah kiri, memperlihatkan beberapa alternative dari garis pemisah.

Hyperplane terbaik dapat ditemukan dengan mengukur *margin* dari *hyperplane* tersebut dan mencari titik maksimal dari *margin* tersebut. *Margin* merupakan jarak

antara *hyperplane* dengan data terdekat dari masing-masing kelas. Data yang terdekat dari *margin* disebut *support vector*. Pada gambar (2-2) sebelah kanan, terlihat garis solid yang menunjukkan *hyperplane* terbaik, yang terletak tepat ditengah kedua kelas. Titik-titik merah dan kuning yang berada dalam lingkaran hitam adalah *support vector*.

SVR yang merupakan adaptasi dari SVM memiliki kemampuan untuk mengatasi permasalahan *overfitting*, sehingga bisa mendapatkan suatu fungsi dengan tingkat kesalahan yang kecil dan menghasilkan prediksi yang bagus. Oleh karena itu dalam tugas akhir ini digunakan SVR untuk membangun model prediksi. Dalam studi ini, SVR digunakan untuk memprediksi nilai indeks harga saham IHSG dan JII. SVR menggunakan variable prediksi, yang telah diperoleh dengan menggunakan metode ICA, untuk membangun sistem prediksi *financial time series*, dalam hal ini indeks saham IHSG dan JII.

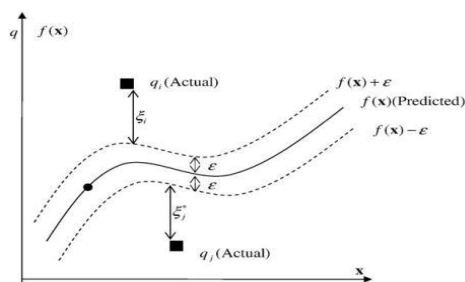
Pada SVR permasalahan regresi non-linear pada ruang input berdimensi rendah ditransformasikan ke dalam sebuah regresi linear pada ruang fitur berdimensi tinggi. Bentuk umum dari SVR secara linear dapat dituliskan sebagai berikut :

$$(2.2)$$

dimana $\phi(x)$ adalah fungsi pemetaan pada ruang fitur, v merupakan vector, dan b merupakan konstanta. Nilai dari koefisien v dan b dapat diestimasi dengan meminimalkan fungsi resiko :

$$(2.3)$$

Dimana ϵ merupakan ϵ -insensitive loss function, λ merupakan ketentuan regularisasi yang mengontrol kompleksitas fungsi, dan C merupakan konstanta yang mengatur regularisasi tersebut.



Gambar 2-5. ϵ -insensitive loss function pada SVR

ϵ -insensitive loss function merupakan parameter yang merepresentasikan radius dari *tube* yang terletak disekitar fungsi regresi ($f(x)$). Nilai ϵ mengatur batas *error* yang dari dataset pada kelasnya masing-masing. Nilai C kemudian menjadi nilai penalti dari dataset yang memiliki nilai *error* lebih dari ϵ . Setiap kesalahan yang nilainya lebih besar dari ϵ maka akan dikenakan penalti sebesar C . Jika nilai C terlalu kecil, maka sistem akan semakin toleran terhadap kesalahan tersebut. Sedangkan jika nilai C terlalu besar, maka sistem akan semakin tidak toleran terhadap kesalahan.

Berikut formula dari ϵ -insensitive loss function :

$$(2.4)$$

Pada gambar (2-5) terdapat *slack variable* (ξ),

yang digunakan untuk mengukur deviasi dari batas ϵ -insensitive, yang merepresentasikan jarak antara data dengan nilai batas zona ϵ -insensitive. Pada gambar (2-3), daerah terdekat dengan *tube* disebut sebagai ϵ -insensitive zone, dimana berdasarkan *loss function* nilai *error* pada zona ini bernilai nol, sehingga data yang terletak pada zona ini atau memiliki nilai *error* lebih kecil dari ϵ maka tidak dianggap sebagai *error*. Data yang tidak terletak di dalam *tube* merupakan *support vector* yang merupakan data penting untuk melakukan generalisasi keputusan.

Dengan menggunakan *slack variable* persamaan (2.3) dapat ditransformasikan dalam bentuk :
minimize :

$$(2.5)$$

subject to :

Dengan menggunakan *Lagrange multiplier* dan kondisi KKT pada persamaan (2.5), akan menghasilkan bentuk dual Lagrangian :

maximize :

$$(2.6)$$

dengan konstrain :

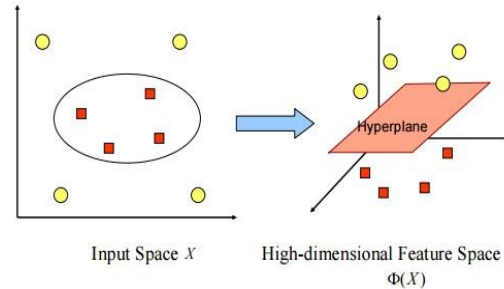
Multiplier Lagrange pada persamaan (2.6) memenuhi . Sehingga bentuk umum dari fungsi SVR-based regression dapat ditulis sebagai berikut :

$$(2.7)$$

dimana merupakan fungsi kernel.

Fungsi Kernel

Umumnya masalah dalam dunia nyata jarang yang bersifat *linear separable*, kebanyakan bersifat non linear. Untuk menyelesaikan permasalahan non linear ini digunakan fungsi kernel. Untuk memecahkan masalah linear dalam ruang dimensi tinggi, yang harus dilakukan adalah mengganti *inner product* (x_i dan x_j) dengan fungsi kernel yang sesuai.



Gambar 2-6. Penggunaan kernel

Ilustrasi konsep kernel ini dapat dilihat pada Gambar (2-6) di atas. Pada sisi kiri diperlihatkan data dari kedua kelas berada pada ruang input berdimensi dua yang tidak dapat dipisahkan secara linear. Sedangkan pada gambar sisi kanan terlihat fungsi ϕ memetakan setiap data pada ruang input ke ruang fitur baru yang berdimensi lebih tinggi (dimensi 3), dimana kedua kelas dapat dipisahkan secara linear oleh sebuah *hyperplane*.

didefinisikan sebagai fungsi kernel. Nilai dari fungsi kernel bernilai sama dengan *inner product* dari dua vektor (x_i dan x_j) pada ruang fitur dan , sehingga . Keunggulan dari penggunaan fungsi kernel ini adalah kemampuannya yang dapat berhubungan dengan ruang fitur berdimensi lebih tinggi tanpa perlu menghitung pemetaan dari secara eksplisit.

Fungsi kernel yang sering digunakan adalah *Radial Basis Function* (RBF), yang dapat didefinisikan sebagai berikut :

$$(2.8)$$

dimana σ^2 merupakan *bandwidth* dari RBF.

2.3 Mean Absolute Percentage Error (MAPE)

MAPE merupakan ukuran akurasi yang menunjukkan seberapa besar kesalahan prediksi dibandingkan dengan nilai sebenarnya. Akurasi dari MAPE diekspresikan dalam bentuk persentase. Semakin kecil nilai MAPE yang dihasilkan maka semakin baik akurasi performansi dari sistem prediksi yang dibangun. Berikut rumus perhitungan MAPE :

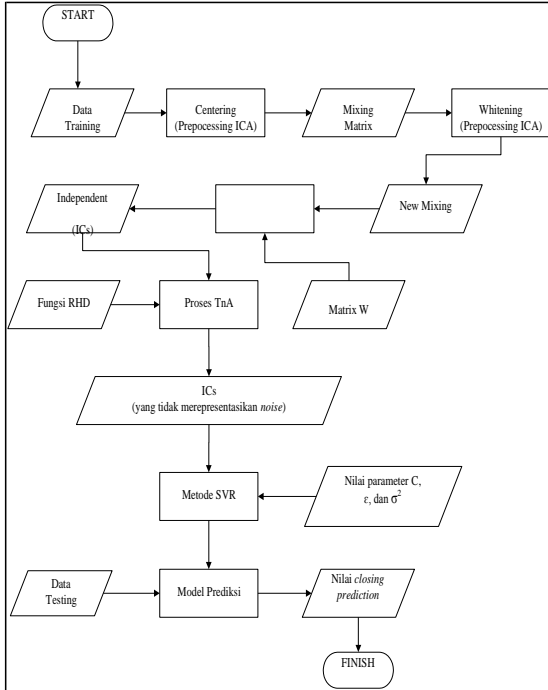
dimana merupakan nilai aktual dan merupakan nilai hasil prediksi.

Berdasarkan [12,13,14,15,16,17] acuan nilai MAPE untuk kasus prediksi indeks saham maupun prediksi saham adalah kurang dari 5% (nilai MAPE < 5%).

3. Perancangan Sistem

Dalam perancangan sistem prediksi dengan menggunakan metode *Independent Component Analysis* dan *Support Vector Regression* ini menggunakan diagram *flowchart* untuk menggambarkan pemodelan sistem yang dibangun.

Berikut gambaran perancangan sistem prediksi harga *closing* :



Gambar 3-1. *Flowchart* perancangan sistem prediksi harga *closing* indeks saham

Berikut penjelasan mengenai *flowchart* perancangan sistem prediksi yang dibangun :

1. Data *training* yang diinputkan berupa nilai harga *open*, *high*, *low*, *close* dari data IHSG dan JII.
2. Melakukan proses *centering* pada data *training*.
3. Melakukan proses *whitening* pada *mixing matrix* dari data *training* yang sebelumnya dihasilkan oleh proses *centering*.
4. *New mixing matrix* yang dihasilkan oleh proses

whitening kemudian diproses bersama *demixing matrix W* menggunakan metode *Independent Component Analysis*, untuk menghasilkan *Independent Components (ICs)*.

5. *Independent Components (ICs)* yang dihasilkan sebagai output dari metode *Independent Component Analysis* kemudian diurutkan

yang menggunakan nilai dari fungsi *Relative Hamming Distance* sebagai indeks pengurutannya.

6. ICs yang merepresentasikan *noise* kemudian

dihapus. Selanjutnya ICs yang tidak merepresentasikan *noise* kemudian digunakan untuk membangun model prediksi menggunakan metode *Support Vector Regression*. Pada proses ini

penentuan nilai parameter C , ϵ , dan σ^2 sangat penting, karena dapat mempengaruhi performansi sistem yang dibangun.

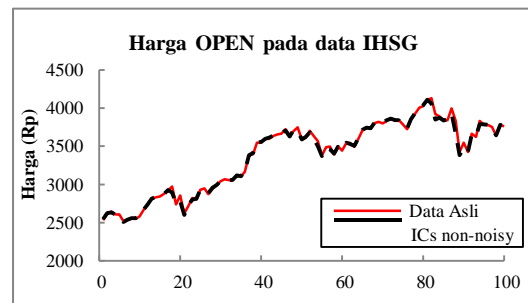
7. Model prediksi kemudian diuji untuk menentukan nilai *closing prediction* dari data *testing* yang digunakan. Data *testing* yang dimasukkan berupa data harga *open*, *high*, dan *low* dari data IHSG maupun JII.
8. Keluaran dari sistem yang berupa nilai *closing prediction* kemudian dibandingkan dengan nilai harga *closing* sebenarnya. *MAPE* digunakan untuk menghitung prosentase *error* yang terjadi antara nilai *closing prediction* dan nilai *closing* yang sebenarnya dari data *testing*.

4. Analisis Hasil Pengujian

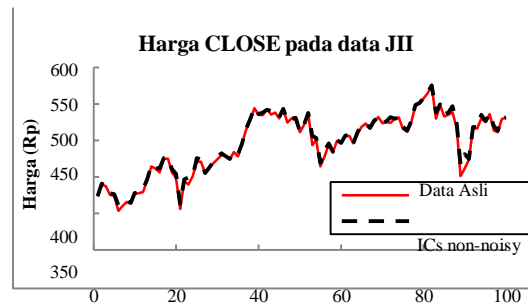
Strategi pengujian yang dilakukan pada sistem prediksi ini akan menggunakan beberapa skenario untuk menganalisis keterkaitan antara perubahan nilai parameter dengan performansi dari sistem prediksi, serta untuk mengetahui hasil identifikasi ICs yang merepresntasikan *noise*. Berikut beberapa skenario pengujian yang akan dilakukan :

1. Skenario untuk mengetahui perbandingan data asli dan ICs *non-noisy* dari data IHSG dan JII.
2. Skenario untuk mengetahui pengaruh perubahan nilai parameter *konstanta C* terhadap performansi sistem prediksi.
3. Skenario untuk mengetahui pengaruh perubahan nilai parameter *loss function* (ϵ) terhadap performansi sistem prediksi.
4. Skenario untuk mengetahui pengaruh perubahan parameter kernel RBF (σ^2) terhadap performansi sistem prediksi.
5. Skenario untuk mengetahui pengaruh penggunaan metode ICA terhadap performansi dari sistem prediksi.

4.1 Perbandingan data asli dan ICs non-noisy.



Gambar 4-1. Perbandingan data asli dan ICs non-noisy pada data IHSG



Gambar 4-2. Perbandingan data asli dan ICs non-noisy pada data JII

Dengan menggunakan metode ICA (*Independent Component Analysis*) dapat dilakukan identifikasi ICs (*independent components*) pada data asli IHSG maupun JII. ICs yang sudah teridentifikasi akan diproses dengan

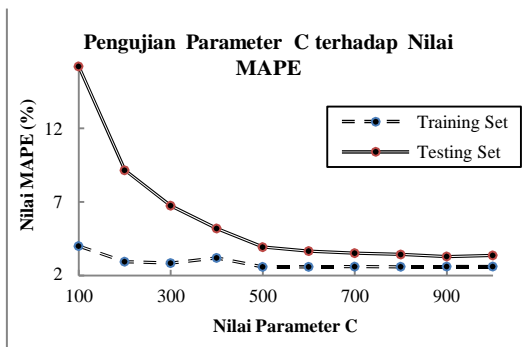
pendekatan *Test-and-Acceptance* yang menggunakan nilai RHD untuk mengurutkan ICs, hingga mendapatkan ICs yang merepresentasikan *noise*. Selanjutnya data ICs yang merepresentasikan *noise* akan dihapus. Pada beberapa gambar di atas, dapat dilihat bahwa terdapat kemiripan antara data asli dengan data ICs yang tidak merepresentasikan *noise* (ICs *non-noisy*). Data ICs yang tidak merepresentasikan *noise* tersebut kemudian akan dijadikan sebagai input dari metode SVR untuk

membangun sistem prediksi indeks saham.

4.2 Pengaruh konstanta C terhadap performansi sistem prediksi.

Pada skenario ini nilai parameter yang diubah hanya parameter C saja, sedangkan untuk parameter ϵ dan σ^2 nilainya tetap. Untuk data IHSG nilai parameter $\epsilon = 10$ dan $\sigma^2 = 100$, sedangkan untuk data JII nilai parameter $\epsilon = 10$ dan $\sigma^2 = 100$. Data IHSG.

- Data IHSG

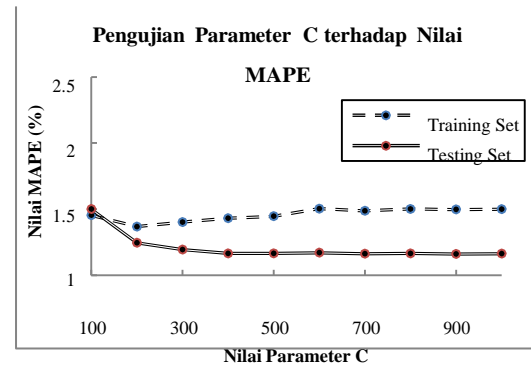


Gambar 4-3. Penguujian parameter C terhadap nilai MAPE pada data IHSG

Berdasarkan Gambar (4-3) untuk data IHSG ini terlihat bahwa, nilai MAPE untuk *training set* cenderung lebih bagus daripada nilai MAPE untuk *testing set*, baik pada saat parameter C bernilai kecil maupun besar. Hal ini berarti generalisasi keputusan yang dihasilkan pada proses *training* masih kurang baik karena hanya cenderung cocok untuk *training set*. Nilai parameter C menggambarkan besarnya penalti toleransi terhadap kesalahan klasifikasi data. Pada kasus ini, sistem yang telah dibangun, mampu dengan baik menoleransi kesalahan klasifikasi data pada *training set*, namun belum cukup baik menoleransi kesalahan klasifikasi data pada *testing set*, sehingga nilai MAPE pada *training set* cenderung lebih bagus daripada *testing set*.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai parameter $\epsilon = 10$, $\sigma^2 = 100$, dan C = 900, dimana untuk data *training set* didapat nilai MAPE = 2.60269% dan untuk data *testing set* didapat nilai MAPE = 3.2874%.

- Data JII



Gambar 4-4. Penguujian parameter C terhadap nilai MAPE pada data JII

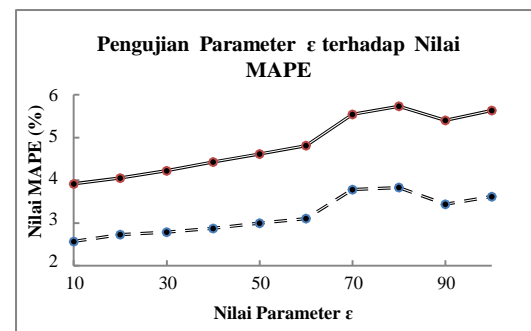
Berdasarkan Gambar (4-4), untuk data JII terlihat bahwa nilai MAPE *testing set* lebih bagus daripada *training set*, baik pada saat parameter C bernilai kecil maupun besar. Hanya saja pada saat nilai parameter C = 100, nilai MAPE *training set* sedikit lebih bagus daripada *testing set*. Hal ini berarti generalisasi keputusan yang dihasilkan pada saat proses *training* sudah cukup baik. Pada kasus ini, saat C bernilai kecil maupun besar, sistem yang telah dibangun mampu menoleransi kesalahan klasifikasi data *testing set* dengan cukup baik.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai parameter $\epsilon = 10$, $\sigma^2 = 100$, dan C = 200, dimana untuk data *training set* didapat nilai MAPE = 1.3679% dan untuk data *testing set* didapat nilai MAPE = 1.24556%.

4.3 Pengaruh loss function ϵ terhadap performansi sistem prediksi.

Pada pengujian ini, perubahan nilai hanya dilakukan pada parameter ϵ saja. Sedangkan untuk parameter C dan σ^2 nilainya tetap. Untuk data IHSG nilai parameter C = 500 dan $\sigma^2 = 100$, sedangkan untuk data JII nilai parameter C = 200 dan $\sigma^2 = 100$.

- Data IHSG



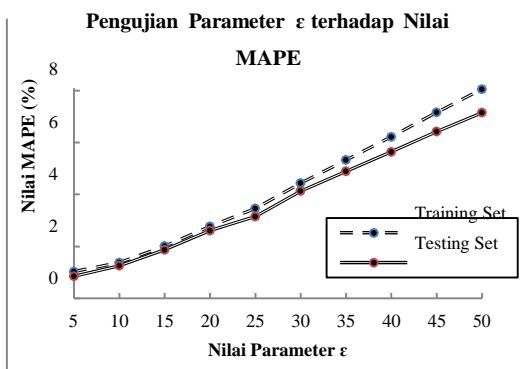
Gambar 4-5. Penguujian parameter ϵ terhadap nilai MAPE pada data IHSG

Berdasarkan Gambar (4-5) untuk data IHSG, terlihat bahwa nilai MAPE untuk *training set* dan *testing set* cenderung membesar baik saat parameter ϵ bernilai kecil maupun besar. Dari grafik tersebut dapat dilihat pula, nilai MAPE *training set* cenderung lebih bagus daripada

testing set baik saat parameter ϵ bernilai kecil maupun besar. Hal ini berarti generalisasi keputusan pada proses training belum cukup baik, karena hanya cenderung cocok untuk training set. Pada kasus ini, sistem yang telah dibangun mampu dengan baik membatasi kesalahan klasifikasi data pada training set daripada testing set, sehingga nilai MAPE untuk training set juga cenderung lebih bagus daripada testing set.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai MAPE = 2.56841% untuk training set dan saat nilai MAPE = 3.91631% untuk testing set, yakni saat nilai parameter $C = 500$, $\sigma^2 = 100$, dan $\epsilon = 10$.

- Data JII



Gambar 4-6. Penguujian parameter ϵ terhadap nilai MAPE pada data JII

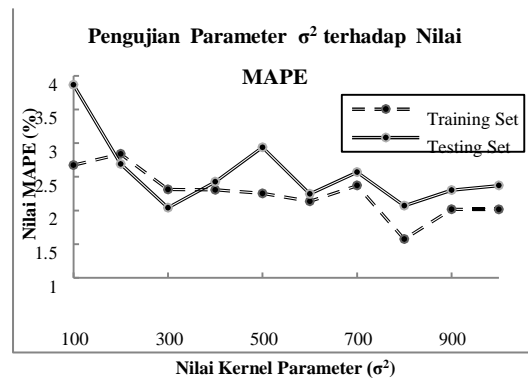
Berdasarkan Gambar (4-6) untuk data JII, terlihat bahwa nilai MAPE untuk testing set maupun training set cenderung membesar seiring semakin besarnya nilai parameter ϵ . Pada kasus ini, generalisasi keputusan yang dibangun pada proses training sudah cukup baik, karena sistem yang telah dibangun mampu dengan baik membatasi kesalahan klasifikasi data pada testing set. Hal ini terlihat dari nilai MAPE untuk testing set yang cenderung lebih bagus dibandingkan nilai MAPE untuk training set, baik saat parameter ϵ bernilai kecil maupun besar.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai parameter $C = 200$, $\sigma^2 = 100$, dan $\epsilon = 10$, dimana untuk data training set didapat nilai MAPE = 1.0325% dan untuk data testing set didapat nilai MAPE = 0.851463%.

4.4 Pengaruh parameter kernel Radial Basis Function (σ^2) terhadap performansi sistem prediksi.

Pada pengujian ini, perubahan nilai hanya akan dilakukan pada parameter σ^2 saja, sedangkan untuk parameter C dan ϵ nilainya tetap. Untuk data IHSG nilai parameter $C = 500$ dan $\epsilon = 10$, sedangkan untuk data JII nilai parameter $C = 200$ dan $\epsilon = 5$.

- Data IHSG



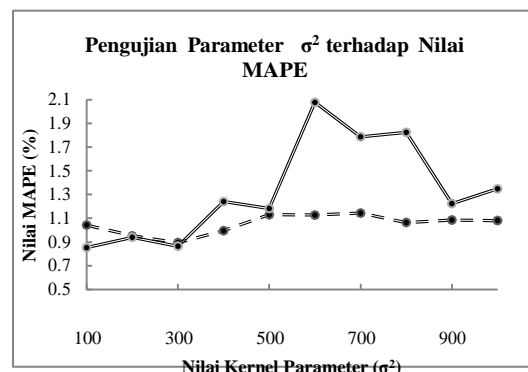
Gambar 4-7. Penguujian parameter σ^2 terhadap nilai MAPE pada data IHSG

Berdasarkan Gambar (4-7) untuk data IHSG, terlihat bahwa pada saat parameter σ^2 bernilai 100 dan 400-1000, MAPE dari training set cenderung lebih bagus daripada testing set. Hal ini berarti persebaran data pada sistem yang dibangun saat proses training cenderung lebih cocok untuk training set dibandingkan untuk testing set. Sedangkan saat σ^2 bernilai 200-300, MAPE dari testing set cenderung lebih bagus dibandingkan training set. Hal ini berarti saat parameter σ^2 bernilai

200-300, persebaran data pada sistem yang telah dibangun sudah baik, karena cenderung cocok saat diuji dengan testing set.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai MAPE = 1.57612% untuk training set dan saat nilai MAPE = 2.07139% untuk testing set, yakni saat nilai parameter $C = 500$, $\epsilon = 10$, dan $\sigma^2 = 800$

- Data JII



Gambar 4-8. Penguujian parameter σ^2 terhadap nilai MAPE pada data JII

Berdasarkan Gambar (4-8) untuk data JII, terlihat bahwa saat σ^2 bernilai 100-300, MAPE dari testing set cenderung lebih bagus dibandingkan training set. Hal ini berarti saat parameter σ^2 bernilai 100-300, persebaran data pada sistem yang telah dibangun sudah baik, karena cenderung cocok saat diuji dengan testing set. Dari grafik dapat dilihat pula, saat σ^2 bernilai 400-1000 nilai

MAPE saat *training set* cenderung lebih bagus dibandingkan nilai MAPE dari *testing set*. Dalam kasus

ini berarti pada saat σ^2 bernilai 400-1000, persebaran data pada sistem yang telah dibangun cenderung lebih cocok untuk *training set* dibandingkan untuk *testing set*, sehingga MAPE dari *training set* cenderung lebih bagus.

Pada kasus ini, sistem prediksi memiliki performansi paling baik pada saat nilai parameter $C = 200$, $\epsilon = 5$, dan $\sigma^2 = 300$, dimana untuk *training set* didapat nilai MAPE = 0.894803% dan untuk *testing set* didapat nilai MAPE = 0.864913%.

4.5 Pengaruh penggunaan metode ICA terhadap performansi sistem prediksi.

- Data IHSG

Tabel 4-1. Perbandingan nilai MAPE dari ICA-SVR dan SVR pada IHSG

Data IHSG	MAPE (%)	
	ICA – SVR	SVR
Training Set	1.57612	1.5655
Testing Set	2.07139	2.9137

- Data JII

Tabel 4-2. Perbandingan nilai MAPE dari ICA-SVR dan SVR pada JII

Data JII	MAPE (%)	
	ICA – SVR	SVR
Training Set	0.895803	1.5023
Testing Set	0.864913	1.3834

Salah satu permasalahan dalam prediksi *time series* adalah adanya *noise* pada data *time series* yang digunakan. *Noise* ini direpresentasikan sebagai suatu anomali pada data yang tidak dapat menangkap *trend* dari data *time series* yang digunakan. *Noise* akan sangat mempengaruhi akurasi dari sistem prediksi. Dalam hal ini, ICA digunakan untuk melakukan identifikasi ICs. ICs yang merepresentasikan *noise* kemudian akan dihapus. Selanjutnya ICs yang tidak merepresentasikan *noise* akan digunakan sebagai input untuk membangun sistem prediksi. Berdasarkan Tabel (4-1) dan Tabel (4-2), terlihat bahwa nilai MAPE pada metode ICA-SVR lebih bagus daripada pada metode SVR saja. Hal ini berarti, dengan menggunakan metode ICA, akurasi dari sistem prediksi bisa lebih bagus, karena data yang digunakan pada saat proses membangun sistem prediksi sudah bersih dari *noise*.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan pengujian yang telah dilakukan, dapat ditarik beberapa kesimpulan, antara lain :

1. Dengan menggunakan metode ICA dapat dilakukan identifikasi ICs dari data asli IHSG dan JII. Dari hasil identifikasi tersebut terlihat bahwa terdapat kemiripan antara data asli dengan data ICs yang tidak merepresentasikan *noise* (ICs *non-noisy*). Selanjutnya ICs *non-noisy* tersebut digunakan untuk membangun sistem prediksi.

2. Semakin besar nilai parameter C maka sistem akan semakin tidak menoleransi adanya kesalahan klasifikasi dari data, akibatnya performansi sistem semakin bagus, yang ditandai dengan nilai MAPE yang cenderung kecil. Sedangkan semakin kecil nilai C , maka sistem semakin toleran terhadap kesalahan klasifikasi data, akibatnya performansi sistem semakin tidak bagus, yang ditandai dengan nilai MAPE yang semakin besar.
3. Semakin besar nilai *loss function* ϵ maka batas *error* akan semakin besar, yang artinya *e-insensitive zone* akan semakin besar rentangnya. Karena pada zona ini kesalahan bernilai nol (data yang terletak pada zona ini tidak dianggap salah), maka memungkinkan data yang mengalami salah klasifikasi akan ditoleransi, sehingga menurunkan akurasi performansi sistem yang dibangun, yang ditandai dengan nilai MAPE yang semakin besar. Sedangkan semakin kecil nilai *loss function* ϵ maka semakin bagus performansi sistem yang dibangun.
4. Dalam kasus ini, untuk data IHSG, baik untuk *training set* maupun *testing set*, semakin kecil nilai σ^2 maka penyebaran datanya cenderung kurang baik, terlihat dari nilai MAPE yang cenderung besar. Sedangkan untuk data JII, baik untuk *training set* maupun *testing set*, semakin kecil nilai σ^2 maka penyebaran datanya cenderung baik, terlihat dari nilai MAPE yang cenderung kecil.
5. Dalam kasus ini, penggunaan metode *Independent Component Analysis* (ICA) dan *Support Vector Regression* (SVR) pada data IHSG dan JII menghasilkan sistem prediksi dengan akurasi yang cukup bagus, dengan MAPE rata-rata kurang dari 5%.
6. Pada data IHSG, sistem akan menghasilkan performansi yang paling baik pada saat $C = 500$, $\epsilon = 10$ dan $\sigma^2 = 800$. Sedangkan pada data JII, sistem akan memberikan nilai akurasi yang paling bagus pada saat $C = 200$, $\epsilon = 5$ dan $\sigma^2 = 300$.
7. Penggunaan metode ICA pada studi ini dapat meningkatkan akurasi hasil prediksi. Hal ini terlihat dari nilai MAPE pada metode ICA-SVR lebih bagus daripada metode SVR saja.

5.2 Saran

Adapun saran untuk pengembangan sistem prediksi indeks saham dengan menggunakan metode *Independent Component Analysis* (ICA) dan *Support Vector Regression* (SVR) ini antara lain :

1. Memperbanyak data *training set* dengan nilai yang lebih bervariasi, dengan harapan agar dapat menghasilkan fungsi prediksi yang lebih bagus.
2. Mengkombinasikan metode ICA dengan metode prediksi lainnya untuk mengevaluasi kemampuan ICA dalam melakukan *denoising* data indeks saham untuk menghasilkan suatu prediksi.
3. Mengkombinasikan metode SVR dengan metode pemrosesan sinyal lainnya untuk meningkatkan performansi dari prediksi yang dihasilkan.

4. Menambahkan berbagai teknikal indikator untuk mengevaluasi pengaruhnya terhadap prediksi nilai *closing* indeks saham.

Daftar Pustaka

- [1] C.J. Lu, T.S. Lee, C.C. 2009. Chiu. Financial Time Series Forecasting using Independent Component Analysis and Support Vector Regression. Taiwan.
- [2] T.S. Lee, N.J. Chen. Investigating the Information Content of Non-Cash Trading Index Futures Using Neural Networks. *Expert Systems Applications* 22 (2002) 225-234.
- [3] F.E.H. Tay, L. Cao. 2001. Application of Support Vector Machine in Financial Time Series. Singapore.
- [4] L.J. Cao. Support Vector Machines Experts for Time Series Forecasting. *Neurocomputing* 51 (2003) 321-339.
- [5] N. Cristianini, J. Shawe-Taylor. 2000. An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press. United Kingdom.
- [6] V. Cherkassky, Y. Ma. Practical Selection of SVM Parameters and Noise Estimation for SVM Regression. *Neural Networks* 17 (2004) 113-126.
- [7] Y. Cheung, L. Xu. Independent Component Ordering in ICA Time Series Analysis. *Neurocomputing* 41 (2001) 145-152.
- [8] A. Hyvärinen, J. Karhunen, E. Oja. 2001. Independent Component Analysis. John Wiley & Sons. New York.
- [9] A. Hyvärinen, E. Oja. Independent Component Analysis : Algorithms and Applications. *Neural Networks* 13 (2000) 411-430.
- [10] D. Basak, S. Pal, D.C. Patranabis. 2007. Support Vector Regression. India.
- [11] Nugroho, A.S., Witarto, A. B., Handoko, D. 2003. Support Vector Machine - Teori dan Aplikasinya dalam Bioinformatika.
- [12] Setiawan, Wahyudi. 2008. Prediksi Harga Saham Menggunakan Jaringan Syaraf Tiruan Multilayer Feedforward Network Dengan Algoritma Backpropagation. Bali.
- [13] Anshari, M.F.Lutfi., Ispriyanti, Dwi. 2013. Prediksi Interval Harga Penutupan Saham Dengan Fuzzy Autoregressive (FAR) (Studi Kasus pada Saham PT. Bank Mandiri (Persero) Tbk).
- [14] Amalia, Nur., Kristanti, Farida Titik. Model GARCH (Generalized Autoregressive Conditional Heteroscedasticity) Untuk Prediksi dan akurasi harga saham masa depan.
- [15] Hartanti, Ana. 2014. Peramalan Harga Saham Pada Lima Emiten Terbaik Versi Forbes Tahun 2012 Menggunakan Fuzzy Model.
- [16] Zulkarnain, Iskandar. Akurasi Peramalan Harga Saham Dengan Model ARIMA dan Kombinasi MAIN CHART + ICHIMOKU CHART.
- [17] Hidayatullah, Taufik. 2014. Kajian Komparasi Penerapan Algoritma Support Vector Machine (SVM) dan Multilayer Perceptron (MLP) Dalam Prediksi Indeks Saham Sektor Perbankan : Studi Kasus Saham LQ45 IDX Bank BCA. Jakarta.
- [18] Zenda Dalam Time Series. 2011. "Baca-Baca Time Series Yukk". <http://zendadelia.blogspot.com/2011/12/baca-baca-time-series-yukk.html>

