

KOMPARASI SUPPORT VECTOR MACHINE (SVM) DAN AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) PADA PERAMALAN HUJAN DI DAERAH ALBURY, AUSTRALIA

Nur Huda Riyantoni¹, Moh. Faqih Bahreisy², Irfan Hakim³, Dwi Rolliawati⁴

Program Studi Sistem Informasi,
Universitas Islam Negeri Sunan Ampel
Surabaya, Gunung Anyar, Surabaya

e-mail: ¹riyantoni2772@gmail.com, ²reikun1207@gmail.com,
³irfanhakim240702@gmail.com, ⁴dwi-roll@uinsby.ac.id

Abstract

Rain is one of the 4 weather conditions on earth. The rain itself is sometimes erratic. For some people, rain is sometimes seen as an obstacle in carrying out their daily activities. Therefore, predictions about rain are needed. Predictions about rain really help people in carrying out their daily activities. The research aims to compare the predictive performance of SVM and ARIMA in forecasting rain. The data used in this case study research is data regarding daily weather for 10 years in the Albury area, Australia from 2008-2017 with a lot of 3040 data. The results obtained from SVM for forecasting rain using daily weather data for 10 years in the Albury area, Australia with the best accuracy rate with the SVM model is 97.532% with an error rate of 2.468%, while in the ARIMA model, the results are MAE 0.181, RMSE 0.254 and MAPE 0.159. So it can be concluded that the ARIMA model has a better performance in predicting rain than the SVM method.

Keyword: Hujan, Komparasi, Peramalan, SVM, ARIMA

PENDAHULUAN

Hujan merupakan salah satu dari 4 cuaca yang ada di bumi. Hujan terjadi dikarenakan melewati proses penguapan air dari permukaan bumi yang naik ke atmosfer lalu mengembun dan menggumpal menjadi awan. Ketika embun awan tersebut mengalami perubahan suhu maka partikel-partikel es pada awan akan mencair menjadi tetesan air yang turun ke bumi sebagai hujan (Faradiba, 2021). Turunnya hujan sendiri terkadang tidak menentu waktunya.

Bagi sebagian masyarakat, hujan merupakan sesuatu yang dinanti-nanti. Namun, bagi masyarakat lainnya hujan terkadang mereka anggap sebagai halangan dalam menjalankan aktivitas sehari-hari terutama bagi wisatawan yang ingin liburan dengan mengunjungi berbagai objek wisata. Oleh karena itu, dibutuhkannya prediksi mengenai hujan. Adanya prediksi mengenai hujan dapat memudahkan masyarakat menjalankan aktivitas hariannya. Dengan adanya prakiraan hujan dapat juga membantu memandu dan mendorong wisatawan untuk mengunjungi dan menghindari daerah tertentu.

Machine Learning (ML) merupakan teknologi yang dapat digunakan untuk memprediksi hujan (Suma, 2020). Nantinya, ML akan menganalisis kondisi cuaca pada suatu daerah tertentu untuk menghasilkan prediksi cuaca. ML merupakan cabang dari *Artificial Intelligence* (AI) tentang algoritma komputer untuk meningkatkan kinerja program komputer secara otomatis berdasarkan data. (Supeno, 2019).

Support Vector Machine (SVM) adalah teknik berbasis ML yang bisa digunakan untuk prediksi. Selain itu, SVM dapat digunakan dalam peramalan *time series*. Hal ini dikarenakan SVM memiliki fungsi kernel sehingga dapat menyelesaikan permasalahan non-linier (Favorisen et al., 2019). Sedangkan *Autoregressive Integrated Moving Average* (ARIMA) adalah perpaduan model dari *Autoregressive* (AR) dan *Moving Average* (MA). AR adalah model yang menjelaskan pergerakan variabel berdasarkan pergerakan variabel tersebut di masa lalu. Sedangkan MA adalah sebuah model yang melihat pergerakan variabel berdasarkan residunya di masa lalu. Oleh karena

itu, ARIMA bagus digunakan untuk peramalan data jangka pendek (Hendrawan, 2012).

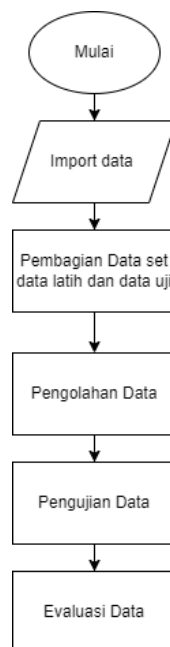
Telah terdapat beberapa penelitian terdahulu yang menggunakan SVM untuk memprediksi sesuatu. Seperti penelitian yang dilakukan oleh Imelda dkk (2021) dalam membandingkan SVM dan *Backpropagation* untuk meramalkan banyaknya wisatawan mancanegara yang datang ke Bali yang menunjukkan bahwa, tingkat akurasi terbaik diraih oleh SVM (Tarigan et al., 2021). Kemudian penelitian yang juga dilakukan oleh Prima dkk (2019) dalam membandingkan metode SVM, *Naive Bayes Classifier* (NBC) dan Diskriminan pada data kebakaran hutan yang juga menunjukkan bahwa, metode yang memiliki akurasi paling tinggi adalah metode SVM (Rahmadina, 2019).

Untuk ARIMA terdapat penelitian yang dilakukan oleh Sismi (2018) dalam perbandingan prediksi metode ARIMA dan *Moving average* pada prediksi harga saham PT. BRI, Tbk yang menghasilkan metode yang paling baik digunakan adalah ARIMA (Sismi & Darsyah, 2018). Penelitian mengenai ARIMA juga dilakukan oleh Safitri dkk (2021) dalam membandingkan metode ARIMA dan *Exponential Smoothing Holt-Winters* pada data kunjungan dengan hasil metode yang tepat untuk digunakan adalah ARIMA (Rumini, 2020).

Penelitian ini akan membandingkan kinerja antara SVM dan ARIMA dalam melakukan peramalan hujan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia dengan tujuan untuk memperoleh model dengan performa prediksi yang paling baik antara SVM dan ARIMA. Perolehan hasil perbandingan kinerja tersebut didapatkan melalui pengolahan data menggunakan KNIME dan Python.

METODE PENELITIAN

Gambar 1 menjelaskan alur penelitian yang digunakan untuk melakukan komparasi antara kinerja prediksi SVM dan ARIMA. Adapun data yang digunakan dalam penelitian studi kasus ini adalah data mengenai harian cuaca selama 10 tahun pada daerah Albury, Australia. dari tahun 2008-2017 dengan banyak data 3040 data.



Gambar 1. Alur Penelitian

Setelah data diperoleh, dilakukan tahapan import data pada Python dan KNIME. Adapun data yang diimport berjenis CSV (*Comma-separated values*). Setelah data diimport, dilakukan pembagian data menjadi dua jenis, yakni data latih dan data uji. Seperti namanya, data latih digunakan dalam melatih algoritma untuk mencari model yang sesuai, sedangkan data uji digunakan dalam hal pengujian kinerja model yang didapatkan pada tahap uji (Rezkia, 2021).

Fase pengolahan data dan pengujian data dilakukan menggunakan pemrograman Python

dan pada KNIME. Pada pemrograman Python digunakan beberapa library seperti numpy, pandas, seaborn, dll. Sedangkan pada KNIME akan dibentuk flow terlebih dahulu sebelum data diolah.

Setelah data diuji, akan didapatkan hasil akurasi yang didapatkan guna melakukan fase evaluasi data. Pada fase tersebut, kinerja SVM dan ARIMA akan dibandingkan untuk mencari model dengan performa prediksi paling baik.

Autoregressive Integrated Moving Average (ARIMA)

Autoregressive Integrated Moving Average (ARIMA) adalah model yang sepenuhnya mengabaikan variabel independen dalam membuat peramalan. ARIMA menggunakan nilai dependen dari masa lalu dan variabel masa yang sekarang untuk menghasilkan prakiraan jangka pendek yang akurat. ARIMA memiliki akurasi yang tinggi jika observasi dari deret waktu (*time series*) secara statistik berkorelasi satu sama lain (dependen) (Hendrawan, 2012).

1. *Autoregressive Model* (AR)

Umumnya Model *autoregressive* dengan ordo p (AR(p)) atau model ARIMA ($p,0,0$) dijelaskan sebagai berikut:

$$X_t = \mu' + \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \dots + \varphi_p X_{t-p} + e_t$$

dimana:

μ' = konstanta

φp = parameter autoregresif ke- p

e_t = nilai kesalahan pada saat t

2. *Moving Average Model* (MA)

Umumnya model moving average ordo q (MA(q)) atau ARIMA ($0,0,q$) dijelaskan sebagai berikut:

$$X_t = \mu' + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \dots - \theta_q e_{t-q}$$

dimana:

μ' = konstanta

θ_1 sampai θ_q adalah parameter-parameter moving average

e_{t-k} = nilai kesalahan pada saat $t - k$

Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah salah satu metode *supervised learning* yang sering digunakan untuk klasifikasi data (seperti *Support Vector Classification*) dan juga umum digunakan dalam regresi (*Support Vector Regression*) (Nugroho et al., 2003). Dalam membuat model klasifikasi, SVM memiliki konsep yang lebih sempurna dan mendetail secara matematis dibandingkan dengan teknik klasifikasi yang lainnya (Awad & Khanna, 2015). SVM juga digunakan untuk mengatasi masalah klasifikasi dan regresi secara linear maupun non-linear (Nanda et al., 2018).

Support Vector Machine (SVM) adalah salah satu metode machine learning yang bekerja berdasarkan prinsip *Structural Risk Minimization* (SRM) yang bertujuan untuk mencari hyperplane terbaik dengan cara memisah dua buah class pada input space. Tujuan penggunaan kernel yaitu untuk merubah data ke ruang yang berdimensi tinggi dengan cara menjadikan data non-linier terpisah secara linier. Terdapat beberapa jenis kernel berdasarkan fungsinya yang digunakan pada sebuah aplikasi untuk mengatasi masalah pada metode *Support Vector Machine* (SVM), antara lain (Zaiem Praghakusma & Charibaldi, 2021):

1. Kernel Linear

$$K(x_i, x) = x_i^T x$$

dimana:

x_i = data latih

x = data uji

2. Kernel Polynomial

$$K(x_i, x) = (\gamma(x_i^T x) + r)^p$$

dimana:

x_i = data latih

x = data uji

γ = *gamma*

r = koefisien

p = derajat polinomial

3. Kernel Radial Basis Function (RBF)

$$K(x_i, x) = \exp(-\gamma||x_i - x||^2 + C)$$

dimana:

x_i = data latih

x = data uji

c = cost

HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk menguji performa antara model SVM dan ARIMA dalam peramalan hujan dengan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia. Pembahasan dimulai dengan menentukan parameter terbaik pada setiap metode dan dilanjutkan dengan komparasi antara kedua metode.

Tabel 1. Parameter yang digunakan

Nama Parameter	Deskripsi	Tipe Data
Date	Waktu terjadinya cuaca	Date
Location	Tempat terjadinya cuaca	String
Min Temp	Tempertur minimum cuaca	Float
Max Temp	Tempertur maksimum cuaca	Float
Rainfall	Proses terjadinya curah hujan	Float
Wind Gust Speed	Kecepatan hembusan angin	Int
Wind Speed 9am	Kecepatan angin pada pukul 9 pagi	Int
Wind Speed 3pm	Kecepatan angin pada pukul 3 sore	Int
Humidity 9am	Kelembapan pada pukul 9 pagi	Int
Humidity 3pm	Kelembapan pada pukul 3 sore	Int
Pressure 9am	Tekanan pada pukul 9 pagi	Float
Pressure 3pm	Tekanan pada pukul 3 sore	Float
Temp 9am	Temperature pada pukul 9 pagi	Float
Temp 3pm	Temperature pada pukul 3 sore	Float
Rain Today	Terjadinya hujan	String

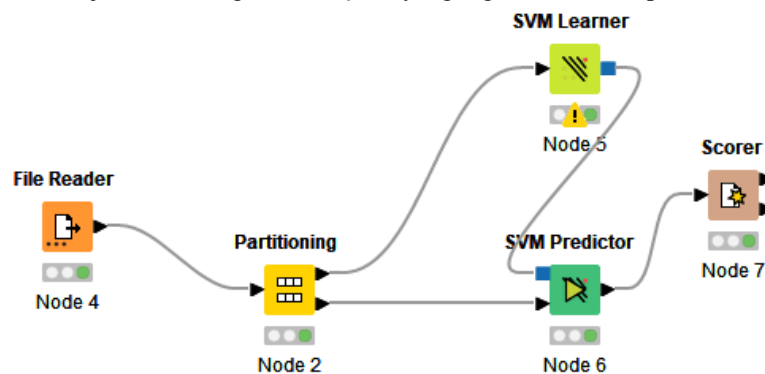
Sumber: Australia Weather Data | (Australia Weather Data / Kaggle, n.d.)

A. Prediksi Menggunakan SVM

Prediksi untuk SVM dilakukan menggunakan KNIME dan pemrograman Python, berikut merupakan penjelasannya:

- Menggunakan KNIME

Pada Gambar 2 dijelaskan mengenai *workflow* yang digunakan untuk prediksi SVM







Gambar 2. Model SVM Menggunakan KNIME

Mula mula, data diimport menggunakan node file reader, data yang diimport berformat csv. Setelah data diimport, langkah selanjutnya adalah membagi data menjadi data latih dan data uji, hal ini dilakukan dengan bantuan node partitioning.

Data yang digunakan untuk data latih sebesar 80%, sedangkan data uji menggunakan 20% data sisanya. Pada node tersebut juga dilakukan pemilihan metode partisi. Langkah selanjutnya adalah memilih kernel yang akan digunakan pada node svm learner. Setelah itu, data akan diproses pada node svm predictor dan akan ditampilkan hasil Confusion Matrix dan tingkat akurasi prediksi pada node scorer.

Adapun penjelasan node-node yang digunakan pada *workflow* KNIME dijelaskan pada Tabel 2 berikut

Tabel 2. Penjelasan Node yang dipakai pada Flow SVM

Node	Penjelasan Node
File Reader 	Digunakan untuk mengimport file yang berisi data yang akan diprediksi
Partitioning 	Digunakan untuk membagi data menjadi data latih dan data uji. Node tersebut juga digunakan untuk menentukan jenis sampling apa yang dipakai dalam mengolah data
SVM Learner 	Digunakan untuk melatih mesin vektor dukungan pada data input. Node ini juga digunakan untuk memilih tipe kernel yaitu HyperTangent, Polynomial, dan RBF.
SVM Predictor 	Node ini menggunakan model SVM yang dihasilkan oleh node SVM learner untuk memprediksi output untuk nilai yang diberikan.



Untuk menampilkan Confusion Matrix data yang telah diproses serta menampilkan tingkat akurasi prediksi

Sumber: KNIME Community Hub | (Nodes – KNIME Community Hub, n.d.)

Agar mendapatkan tingkat akurasi yang paling tinggi, data diuji menggunakan tipe sampling yang berbeda serta kernel yang berbeda. Adapun hasil dari data yang sudah diproses dijelaskan pada tabel di bawah ini.

Tabel 3. Hasil pengolahan data SVM menggunakan KNIME

Kernel	Metode Sampling	Akurasi
Polynomial	Random Sampling	99,671%
	Stratified Sampling	99,507%
RBF	Random Sampling	78,783%
	Stratified Sampling	79,77%

Dari Tabel 3 didapatkan tingkat akurasi paling tinggi adalah ketika data diolah menggunakan kernel Polynomial dan menggunakan metode random sampling yakni sebesar 99,671%.

- Menggunakan Python

Setelah melalui tahapan pengkodean menggunakan Python, didapatkan hasil akurasi sebagai berikut.

Tabel 4. Hasil pengolahan data SVM menggunakan Python

Kernel	Metode Sampling	Akurasi
Polynomial	Random Sampling	93,914%
	Stratified Sampling	92,105%
RBF	Random Sampling	97,532%
	Stratified Sampling	96,710%

Dari hasil yang diperoleh, dapat disimpulkan bahwa, tingkat akurasi paling tinggi didapatkan ketika data diolah menggunakan kernel RBF dan menggunakan metode random sampling yakni sebesar 97,532%.

- Komparasi hasil SVM menggunakan KNIME & Python

Pengolahan data menggunakan KNIME memiliki tingkat akurasi paling tinggi sebesar 99,671% dengan menggunakan kernel Polynomial dan menggunakan metode *random sampling*.

Sedangkan hasil pengolahan data menggunakan Python memperoleh tingkat akurasi paling tinggi sebesar 97,532% dengan menggunakan kernel RBF dan menggunakan metode *random sampling*.

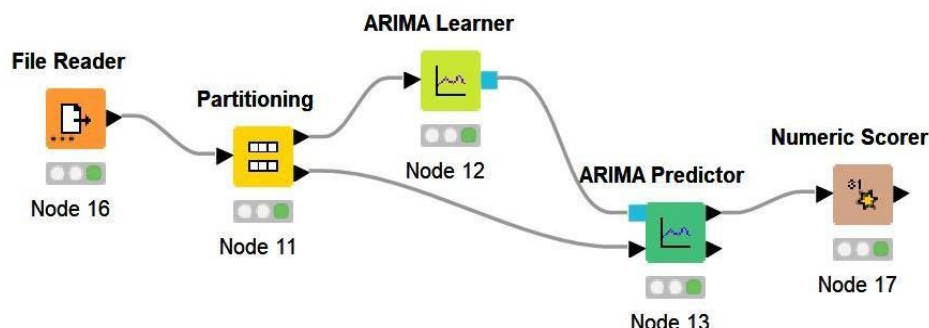
Dari hasil yang sudah diperoleh tersebut, dapat disimpulkan bahwa pengolahan data untuk peramalan hujan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia paling baik adalah menggunakan pemrograman Python dengan memakai kernel RBF dan menggunakan metode *random sampling*.

B. Prediksi Menggunakan ARIMA

Prediksi untuk ARIMA dilakukan menggunakan KNIME dan pemrograman Python, berikut merupakan penjelasannya.

- Menggunakan KNIME

Pada Gambar 3 dijelaskan mengenai *workflow* yang digunakan untuk prediksi ARIMA





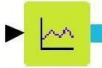


Gambar 3. Model ARIMA Menggunakan KNIME

Hal yang pertama dilakukan adalah mengimport data menggunakan node file reader. Setelah data diimport, langkah selanjutnya adalah membagi data menjadi data latih dan data uji, hal ini dilakukan dengan bantuan node partitioning.

Data yang digunakan untuk data latih sebesar 80%, sedangkan data uji menggunakan 20% data sisanya. Pada node tersebut juga dilakukan pemilihan metode partisi. Langkah selanjutnya adalah mengisi parameter ARIMA pada node ARIMA learner, parameter yang digunakan adalah (1,1,1) dan mengisi periode yang ingin diramal pada node ARIMA predictor.

Setelah itu, data akan diproses dan akan ditampilkan tingkat error dari data yang sudah diproses pada node Numeric Scorer. Adapun penjelasan dari node-node yang digunakan pada workflow KNIME dapat dilihat pada Tabel 5 berikut.

Tabel 5. Penjelasan Node yang dipakai pada Flow ARIMA

Node	Penjelasan Node
File Reader 	Digunakan untuk mengimport file yang berisi data yang akan diprediksi
Partitioning 	Digunakan untuk membagi data menjadi data latih dan data uji. Node tersebut juga digunakan untuk menentukan jenis sampling apa yang dipakai dalam mengolah data
ARIMA Learner 	Digunakan untuk memperkirakan parameter time series pada model ARIMA
ARIMA Predictor 	Untuk Menghitung prediksi dari estimasi model ARIMA
Numeric Scorer 	Untuk menampilkan dan menghitung statistik antara nilai kolom numerik (r i) dan nilai prediksi (pi i).

Sumber: KNIME Community Hub | (Nodes – KNIME Community Hub, n.d.)

Supaya mendapatkan tingkat akurasi yang paling tinggi, data diuji menggunakan tipe sampling yang berbeda. Adapun hasil dari data yang sudah diproses dijelaskan pada tabel di bawah ini.

Tabel 6. Hasil pengolahan data ARIMA menggunakan KNIME

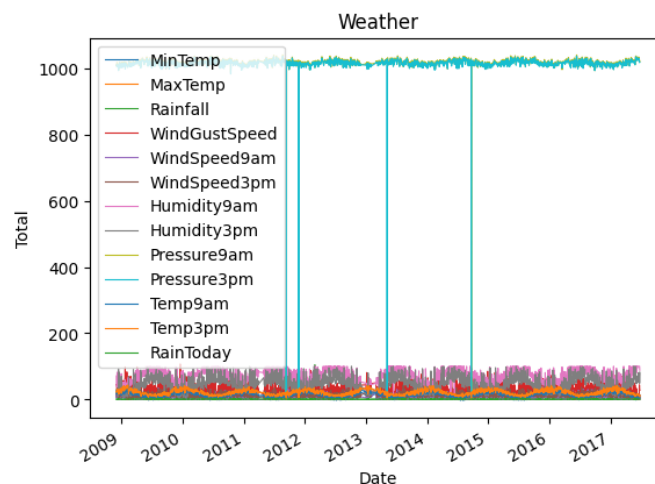
Metode Sampling	MAE	RMSE	MAPE
Random Sampling	1,188	1,255	0,989
Stratified Sampling	1,188	1,254	0,989

Dari Tabel 6 didapatkan bahwa, tingkat error yang paling kecil adalah ketika data diolah menggunakan metode stratified sampling yakni memiliki MAE 1,188, RMSE 1,254 dan MAPE 0,989.

- Menggunakan Python

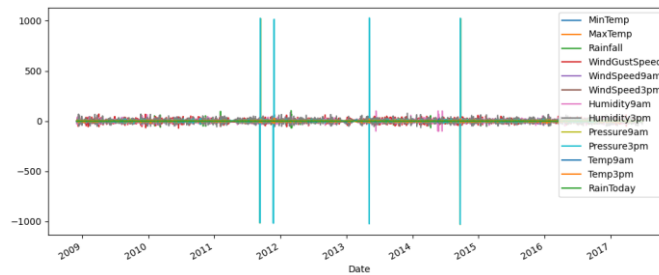
Dalam memprediksi data hujan menggunakan ARIMA dengan python diawali dengan mengimport data dari csv kedalam kode dan juga dilakukan parsing kolom disesuaikan dengan tipe data kolom. Setelah itu dilakukan penghapusan kolom yang tidak digunakan, seperti kolom lokasi. Lalu dilakukang mapping pada data yang akan diprediksi yaitu RainToday dikarenakan Pada kolom tersebut terdapat data berjenis string dengan dilakukan perubahan data dengan nilai 'NO' menjadi 1, dan juga mengubah data dengan nilai 'YES' menjadi 2.

Sebelum melakukan mengolah data hujan menggunakan ARIMA, diperlukan data yang stationer, maka setelah dilakukan beberapa langkah diatas data diolah terlebih dahulu menjadi data stationer dengan dilakukan diferensiasi pada hujan data yang masih belum stasioner. Perbandingan Dataset harian cuaca sebelum dan sesudah dilakukan diferensiasi ditunjukkan pada Gambar 4 dan 5.



Gambar 4. Dataset Harian Cuaca Selama 10 Tahun pada Daerah Albury Australia sebelum dilakukan diferensiasi (non-stationer)

Gambar 5. Dataset Harian Cuaca Selama 10 Tahun pada Daerah Albury Australia



setelah dilakukan diferensiasi (stationer)

Setelah data dilakukan diferensiasi dan telah menjadi data stationer, langkah berikutnya yaitu dengan membagi data menjadi data training dan data testing, dengan 80% data digunakan untuk data training, dan 20% data sisanya digunakan untuk data testing.

Setelah mendapatkan data stationer dan dilakukan partisi menjadi data testing dan training maka dilakukan permodelan ARIMA dengan parameter (1,1,1). Setelah melalui tahapan pengkodean menggunakan Python, didapatkan hasil akurasi sebagai berikut.

Tabel 7. Hasil pengolahan data ARIMA menggunakan Python

Metode Sampling	MAE	RMSE	MAPE
Random Sampling	0,181	0,244	0,159
Stratified Sampling	0,181	0,244	0,159

Dari Tabel VII diketahui bahwa, masing-masing metode sampling memiliki tingkat error yang sama, yakni sama-sama memiliki MAE 0,181, RMSE 0,254 dan MAPE 0,159

- Komparasi hasil ARIMA menggunakan KNIME & Python

Pengolahan data menggunakan KNIME memiliki tingkat error yang paling kecil dengan MAE 1,188, RMSE 1,254 dan MAPE 0,989 dengan menggunakan metode stratified sampling. Sedangkan hasil pengolahan data menggunakan Python memperoleh tingkat error dengan MAE 0,181, RMSE 0,254 dan MAPE 0,159.

Dari hasil yang sudah diperoleh tersebut, dapat disimpulkan bahwa pengolahan data untuk peramalan hujan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia paling baik adalah menggunakan pemrograman Python.

KESIMPULAN

Penelitian ini, membandingkan antara SVM dan ARIMA dalam melakukan peramalan hujan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia. Hasil yang diperoleh dari SVM untuk peramalan hujan menggunakan data harian cuaca selama 10 tahun pada daerah Albury, Australia dengan tingkat akurasi paling bagus dengan model SVM adalah sebesar 97,532% dengan tingkat error sebesar 2,468%, sedangkan pada model ARIMA diperoleh hasil MAE 0,181, RMSE 0,254 dan MAPE 0,159. Sehingga dapat disimpulkan bahwa, model ARIMA memiliki kinerja yang lebih baik dalam memprediksi hujan dibandingkan dengan metode SVM.

SARAN

Saran untuk penelitian selanjutnya diharapkan dapat menggunakan metode peramalan lainnya untuk dapat menghasilkan tingkat akurasi yang beragam. Adanya hal tersebut, guna mendapatkan metode peramalan yang terbaik untuk meramalkan hujan.

DAFTAR PUSTAKA

- Australia Weather Data / Kaggle. (n.d.). Retrieved January 4, 2023, from <https://www.kaggle.com/datasets/arunavakrchakraborty/australia-weather-data>
- Awad, M., & Khanna, R. (2015). *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*. Apress Berkeley.
- Faradiba, N. (2021). *Proses Terjadinya Hujan dan Penjelasannya Menurut Sains Halaman all - Kompas.com*. KOMPAS.COM.
<https://www.kompas.com/sains/read/2021/05/20/100400923/proses-terjadinya-hujan-dan-penjelasannya-menurut-sains?page=all>
- Favorisen, R. L., Sani, R. S., Kurniawan, D., & Irawati, A. R. (2019). IMPLEMENTASI METODE SUPPORT VECTOR MACHINE DALAM PREDIKSI PERSEBARAN DEMAM BERDARAH DI KOTA BANDAR LAMPUNG. *Jurnal Komputasi*, 7(2), 63–73.
- Hendrawan, B. (2012). Penerapan Model Arima dalam Memprediksi IHSG. *Jurnal Integrasi*, 4(2).
- Nanda, M. A., Seminar, K. B., Nandika, D., & Maddu, A. (2018). A comparison study of kernel functions in the support vector machine and its application for termite detection. *Information*, 9(5), 1–14.
- Nodes – KNIME Community Hub. (n.d.). Retrieved January 4, 2023, from <https://hub.knime.com/search?type=Node&sort=maxKudos>
- Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). Support Vector Machine –Teori dan Aplikasinya dalam Bioinformatika. *Proceeding of Indonesian Scientific Meeting in Central Japan*.
- Rahmadina, R. (2019). *Perbandingan Metode Klasifikasi Support Vector Machine (SVM), Naïve Bayes Classifier (NBC), dan Diskriminan pada Data Forest Fires*. .
- Rezkie, S. M. (2021, February 19). *Algoritma Supervised Vs Unsupervised Learning, Cari Tahu Bed...* <https://dqlab.id/algoritma-supervised-vs-unsupervised-learning>
- Rumini, N. (2020). PERBANDINGAN METODE ARIMA DAN EXPONENTIAL SMOOTHING HOLT-WINTERS UNTUK PERAMALAN DATA KUNJUNGAN. *Jurnal Sistem Informasi*, 9(3), 622–632.
- Sismi, & Darsyah, M. Y. (2018). *Perbandingan Prediksi Harga Saham PT.BRI, Tbk dengan METODE ARIMA dan MOVING AVERAGE*.
- Suma, B. (2020). *PENERAPAN MACHINE LEARNING DI DALAM PREDIKSI CUACA*. Universitas Pasundan.
- Supeno, H. (2019). *Pertemuan 14 Machine Learning*. Universitas Pasundan.
- Tarigan, I. A., Bayupati, I. P. A., & Putri, G. A. A. (2021). Komparasi model support vector machine dan backpropagation dalam peramalan jumlah wisatawan mancanegara di provinsi Bali. *Jurnal Teknologi Dan Sistem Komputer*, 9(2), 90–95.
- Zaiem Praghakusma, A., & Charibaldi, N. (2021). *Komparasi Fungsi Kernel Metode Support Vector Machine untuk Analisis Sentimen Instagram dan Twitter (Studi Kasus : Komisi Pemberantasan Korupsi)*. 9(2), 33–42. <https://doi.org/10.12928/jstie.v8i3.xxx>