E-ISSN 2527-9378

Jurnal Statistika Industri dan Komputasi
Volume 05, No. 2, Juli 2020, pp. 51-61

PERBANDINGAN HASIL KLASIFIKASI CURAH HUJAN MENGGUNAKAN *METODE SVM* DAN *NBC*

[1]Marthin Luter Laia, [2]Yudi Setyawan
Jurusan Statistika, Fakultas Sains Terapan, Institut Sains & Teknologi AKPRIND Yogyakarta
Jl. Kalisahak No. 28 Kompleks Balapan Tromol Pos 45 Yogyakarta 55222
marthinluterlaia403@gmail.com

Abstract

This study discusses rainfall classification and rainfall prediction using the Support Vector Machine and Naïve Bayes Classifier methods and looks at the accuracy of the two methods. Support Vector Machine is one of the machine learning methods that works based on the structural risk minimization principle which aims to find the best hyperplane that can separate classes. Whereas Naïve Bayes Classifier is a classification method used to determine the probability of class members. The variables used are average of temperature (X_1), average of humidity (X_2), average of solar radiation (X_3), and average of wind speed (X_4). While the dependent variable (Y) is the status of rainfall which is categorized into two namely rain and no rain. Data used from 2017 to 2018 were obtained from the BMKG Tanjung Priok Maritime Meteorological Station, North Jakarta.

Based on the results of the classification analysis, it was found that the best method, Support Vector Machine, was proven with an accuracy level of 79.45%, greater than the accuracy level of the Naïve Bayes Classifier method, which was 65.75%.

Keywords: *Rainfall, Support Vector Machine, Naïve Bayes Classifier, Accuracy.*

Abstrak

Penelitian ini membahas pengklasifikasi curah hujan serta memprediksi curah hujan dengan menggunakan metode *Support Vector Machine* dan *Naïve Bayes Classifier* serta melihat nilai akurasi kedua metode. *Support Vector Machine* adalah salah satu metode *machine learning* yang bekerja atas prinsip *structural risk minimization* yang bertujuan untuk menemukan *hyperplane* terbaik yang dapat memisahkan kelas. Sedangkan *Naïve Bayes Classifier* adalah metode klasifikasi yang digunakan untuk menentukan probabilitas suatu anggota dari suatu kelas. Variabel yang digunakan yaitu rata-rata temperatur (X_1), rata-rata kelembapan (X_2), rata-rata lama penyinaran matahari (X_3), dan rata-rata kecepatan angin (X_4). Sedangkan variabel dependen (Y) adalah status curah hujan dikategorikan menjadi dua yaitu hujan dan tidak hujan. Data yang digunakan periode tahun 2017 sampai tahun 2018 yang diperoleh dari BMKG Stasiun Meteorologi Maritim Tanjung Priok, Jakarta Utara.

Berdasarkan hasil analisis klasifikasi didapatkan bahwa metode terbaik yaitu *Support Vector Machine* hal ini dibuktikan dengan tingkat akurasi sebesar 79,45 % lebih besar dari tingkat akurasi metode *Naïve Bayes Classifier* yaitu 65,75%.

Kata Kunci : Curah Hujan, Support Vector Machine, Naïve Bayes Classifier, Akurasi.

1. Pendahuluan

Curah hujan adalah jumlah air yang jatuh di tanah datar selama periode tertentu yang diukur dengan satuan tinggi (mm) di atas permukaan horizontal bila tidak terjadi evaporasi, runoff dan infiltrasi. Jumlah curah hujan diukur sebagai volume air yang jatuh di atas permukaan bidang datar dalam periode waktu tertentu, yaitu harian, mingguan, bulanan, atau tahunan. Intensitas curah hujan yang tinggi yang sering disebut hujan ekstrem dapat mengakibatkan terjadinya banjir. Provinsi DKI Jakarta merupakan salah satu provinsi di Indonesia yang sering dilanda banjir akibat hujan, sehingga menimbulkan kerugian besar bagi warga setempat dan kota yang menjadi penelitian yaitu kota Jakarta Utara.

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan *Machine Learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terakit dari berbagai database besar (Turban, 2005). Klasifikasi adalah pemrosesan untuk menemukan sebuah model atau fungsi yang menjelaskan

dan mencirikan konsep atau kelas data, untuk kepentingan tertentu. Klasifikasi memiliki beberapa metode di antaranya adalah *Naïve Bayes Classifier, Support Vector Machine, Artificial Neural Network, Classification Tree, K-Nearest Neighbor,* Analisis Diskriminan, dan lain-lain. Pada penelitian ini metode klasifikasi yang digunakan adalah metode perbandingan antara *Support Vector Machine, dan Naïve Bayes Classifier.*

Prakiraan curah hujan menjadi salah satu masalah yang paling ilmiah yang menantang teknologi seluruh dunia pada abad terakhir. Dari masalah di atas maka metode *Support Machine Vector* dan *Naïve Bayes Classifier* cocok digunakan karena tidak membutuhkan pemenuhan asumsi seperti metode klasifikasi lainnya misalnya analisis diskiriminan.

Penelitian sebelumya yang berkaitan adalah oleh (Fridayanthie, 2015) dengan judul "Analisa Data Mining Untuk Prediksi Penyakit Hepatitis Dengan Menggunakan Metode *Naive Bayes* dan *Support Vector Machine*". Penelitian yang dilakukan oleh (Vijayarani & Dhayanand, 2015) dengan judul "Prediksi Penyakit Hati Menggunakan Algoritma SVM dan Naïve Bayes". Penelitian yang dilakukan oleh (Riadi, Umar, & Aini, 2019) dengan judul "Analisis Perbandingan Detection Traffic Anomaly dengan Metode Naïve Bayes dan Support Vector Machine".

2. Metode Penelitian

a. Sumber Data

Data yang digunakan dalam penelitian adalah data sekunder yang diperoleh dari BMKG Stasiun Meteorologi Maritim Tanjung Priok, data yang diambil merupakan data iklim harian dan bisa di akses dengan website www.bmkg.go.id.

b. Variabel Penelitian

Adapun variabel yang digunakan dalam penelitian terdiri dari variabel dependen dan variabel independen.

1. Variabel dependen (*Y*) dalam penelitian ini yaitu status curah hujan yang dibagi dua kategori yaitu hujan dan tidak hujan.
2. Variabel independen (*X*) yang digunakan sebanyak empat variabel yaitu rata-rata temperatur, rata-rata kelembapan, rata-rata penyinaran matahari, dan rata-rata kecepatan angin.

c. Tahapan Analisis Data

Tahapan analisis data menggunakan perbandingan metode *SVM* dan *NBC* adalah sebagai berikut:

1. Mengumpulkan data, pengumpulkan data ini kemudian terbentuk *dataset* yang akan digunakan dalam penelitian.
2. Penanganan data missing menggunakan metode interpolasi linier.
3. Analisis Deskiriptif ini dilakukan pada masing-masing variabel.
4. Pada klasifikasi jenis data dibagi menjadi dua yaitu data *training* dan data *testing*. Pada penelitian ini data *training* yang digunakan adalah 90% data. Sedangkan data testing yang digunakan adalah 10% data. Setelah pembagian data *training* dan data *testing* dilakukan, data training di analisis pada proses SVM dan proses NBC.
5. Pada proses SVM, akan dicari fungsi *hyperplane* dengan menentukan fungsi kernel yang digunakan, setelah itu dihitung dengan matriks *mxm*. Selanjutmya ditentukan nilai parameter C untuk mendapatkan nilai alpha atau *support vector* pada data *training.* Nilai parameter C yang digunakan adalah 0.01, 0.1, 1, 10, dan 100. Setelah didapatkan nilai alpha, selanjutnya akan dicari nilai *bias* (*b*). Setelah nilai α, *b* ditemukan maka dilakukan prediksi kelas data *testing* berdasarkan persamaan *hyperplane* yang ada sehingga menghasilkan tabel *confusion matrix*. Tabel tersebut yang nantinya digunakan untuk menghitung akurasi.

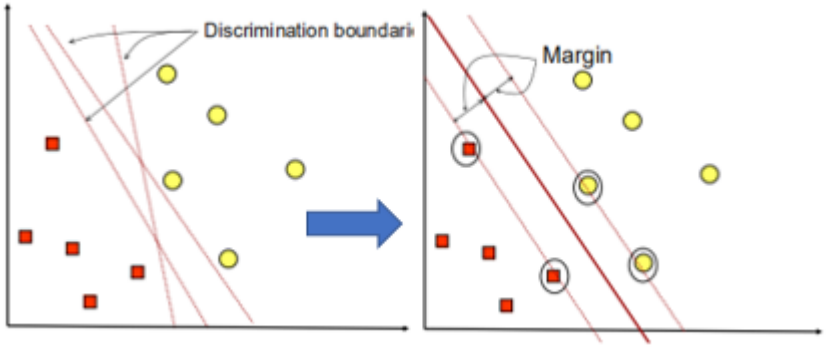
6. Pada proses metode NBC dilakukan perhitungan *probability* baik *prior* maupun *posterior*. Karena data yang digunakan data numerik maka penyelesaiannya menggunakan *densitas gauss*. Proses ini dilakukan pada masing-masing observasi dengan menghitung nilai rata-rata, dan standar deviasi. Setelah *mean* dan standar deviasi ditemukan maka dihitung nilai *likelihood*. Selanjutnya menghitung *posterior probability* dari setiap kategori (*class*) pada data *testing*. Maka tahap selanjutnya normalitas nilai *posterior probability* untuk penentuan kategori (*class*) yang memiliki nilai *posterior probability* yang paling tinggi akan menjadi kategori (*class*) dari data *testing*. Setelah proses NBC dilakukan, maka akan didapatkan hasil tabel *confusion matrix*. Tabel tersebut digunakan untuk mencari nilai akurasi pada metode NBC.
7. Setelah diperoleh nilai akurasi pada metode NBC dan SVM, maka dilakukan perbandingan nilai akurasi pada data *testing*. Nilai akurasi yang paling tinggi merupakan metode klasifikasi terbaik untuk data penelitian ini.
8. Kesimpulan

d. Metode Analisis Data

1) *Support Vector Machine*

Support Vector Machine merupakan salah satu metode klasifikasi data *mining*. SVM adalah algoritma yang bekerja menggunakan pemetaan nonlinear untuk mengubah data pelatihan asli ke dimensi yang lebih tinggi. Dalam hal ini dimensi baru, akan mencari *hyperplane* untuk memisahkan secara linier dan dengan pemetaan *nonlinier* yang tepat ke dimensi lebih tinggi, data dari dua kelas selalu dapat dipisahkan dengan *hyperplane* tersebut (Ritonga & Purwaningsih, 2018).

Pada umumnya masalah dalam domain dunia nyata (*real world problem*) data yang diperoleh jarang yang bersifat linear separable. Kebanyakan bersifat *non*-linear. Dalam menyelesaikan problem non-linear, SVM dimodifikasi dengan memasukkan fungsi *kernel* (Nugroho, Satrio, & Witarto, 2003)



Gambar 1. Konsep *Hyperplane* pada SVM

Pada Gambar 1 menunjukan dua kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Persamaan *hyperplane* yang membatasi kedua kelas sebagai berikut:

$$H_1 : x_i w + b \geq 1 \text{ untuk } y_1 = +1 \quad (1)$$

$$H_2 : x_i w + b \leq -1 \text{ untuk } y_2 = -1 \quad (2)$$

Penggabungan dari persamaan (1) dan (2) menghasilkan pertidaksamaan:

$$y_i (x_i w + b) \geq 1, \text{untuk } \forall i = 1,2,\cdots,n \quad (3)$$

Margin antara dua kelas dapat dihitung dengan mencari jarak antara kedua *hyperplane* H_1 atau H_2. Setiap tupel pelatihan yang jatuh pada *hyperplane* H_1 atau H_2 yang memenuhi persamaan (1) disebut *support vector*. Jarak terdekat suatu titik di bidang H_1 terhadap pusat dapat dihitung dengan meminimalkan $x^T x$ dengan memperhatikan kendala $x_i w+b \geq 1$. Margin dapat dihitung dengan persamaan sebagai berikut:

$$\left|\frac{(1-b)}{\|w\|}-\frac{(-b-1)}{\|w\|}\right|=\frac{2}{\|w\|} \quad (4)$$

Oleh karena memaksimalkan $\frac{1}{\|w\|}$ sama dengan meminimumkan sama dengan meminimumkan $\|w\|^2$ dan untuk menyederhanakan penyelesaian ditambahkan faktor $\frac{1}{2}$ (Asiyah & Fithriasari, 2016). Dengan demikian, model persamaanya menjadi:

$$\min \frac{1}{2}\|w\|^2 \quad (5)$$

$y_i(x_i w+b)\geq 1$, untuk $\forall i=1,2,\cdots,n$ (*n* merupakan jumlah data training).

Dalam klasifikasi kadang-kadang dijumpai bidang pemisah yang tidak bisa diambil dengan linier sehingga diperlukan penyelesaian khusus untuk permasalahan ini. Untuk data-data yang tidak dapat dipisahkan secara linier tersebut ditambahkan variabel *slack* $\xi_i \geq 0$ ke pertidaksamaan (1) sehingga kendala dan fungsi tujuan menjadi:

$$y_i\left(x_i.w+b\right)-1+\xi_i \geq 0, \text{untuk } \forall i=1,2,\cdots,n \quad (6)$$

dengan *n* merupakan jumlah data *training*.

$$Min\frac{1}{2}\|w\|2+C\sum_{i=1}^{n}\xi_i \quad (7)$$

dengan $y_i\left(x_i.w+b\right)-1+\xi_i \geq 0,\ \xi_i \geq 0$, untuk $\forall i$

Hal ini dilakukan untuk mengurangi jumlah kesalahan klasifikasi (*misclassification error*) yang dinyatakan dengan adanya variabel *slack* ξ_i. . *C* adalah parameter yang menentukan besar penalti akibat kesalahan dalam klasifikasi data dan nilainya ditentukan oleh pengguna (Asiyah & Fithriasari, 2016). Untuk menyelesaikan persamaan tersebut, secara komputasi lebih sulit
dan perlu waktu lebih panjang. Untuk itu diperkenalkan pengali *Lagrange* α_i, dengan $i = 1,2,\ldots,n$. Sehingga diperoleh persamaan sebagai berikut:

$$L_D=\sum_{i=1}^{n}a_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}a_i a_j y_i y_j\left(x_i.x_j^T\right) \quad (8)$$

Untuk masalah pengklasifikasian kasus *non-linier* dapat menggunakan fungsi kernel atau disebut sebagai *kernel trick* dimana kita hanya cukup mengetahui fungsi kernel yang dipakai, dan tidak perlu mengetahui wujud dari fungsi non-linear φ (Nugroho, Satrio, & Witarto, 2003). Selanjutnya hasil klasifikasi dari data *x* diperoleh dari persamaan berikut:

$$f(x)=sign\left(\sum_{i=1}^{n}a_i y_i K(x_i^T.x_j)+b\right) \quad (9)$$

Pada penelitian ini menggunakan fungsi kernel linier. Berikut beberapa fungsi kernel sebagai berikut:

1. Kernel Linier
$$K\left(x_i,x_j\right)=x_i^T.x_j \quad (10)$$
2. Kernel Polynomial
$$K(\overrightarrow{x_i},\overrightarrow{x_j})=(\overrightarrow{x_i}.\overrightarrow{x_j}+1)^p \quad (11)$$

3. Kernel Gaussian atau RBF

$$K(\overrightarrow{x_i}, \overrightarrow{x_j}) = \exp\left(-\frac{\| \overrightarrow{x_i} - \overrightarrow{x_j} \|^2}{2\sigma^2} \right) \quad (12)$$

4. Kernel Sigmoid

$$K(\overrightarrow{x_i}, \overrightarrow{x_j}) = \tanh(\alpha \overrightarrow{x_i}.\overrightarrow{x_j} + \beta) \quad (13)$$

2) Naïve Bayes Classifier

Naïve Bayes Classifier merupakan salah satu algoritma dalam teknik data mining yang menerapkan teori bayes dalam klasifikasi (Ridwan, Suyono, & Sarosa, 2013). NBC juga sering disebut *Bayessian Classification* merupakan proses metode klasifikasi yang digunakan untuk menentukan probabilitas suatu anggota dari suatu kelas. NBC handal dalam menangani *dataset* yang berukuran besar serta dapat menangani data yang tidak relevan. Simbol untuk *X* adalah vektor masukan yang berisi data dan *Y* adalah label kelas. Persaman dari teorema bayes sebagai berikut:

$$P(Y \mid X) = \frac{P(X \mid Y)P(Y)}{P(X)} \quad (14)$$

Untuk menjelaskan metode *Naïve Bayes*, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut (Saleh, 2015). Karena itu, metode *Naïve Bayes* diatas disesuaikan sebagai berikut:

$$P(Y_j \mid X_1, \ldots, X_n) = \frac{P(Y)P(X_1, \ldots, X_n \mid Y_j)}{P(X_1, \ldots X_n)} \quad (15)$$

Dimana variabel Y_j mempresentasikan kelas, sementara varaibel $X_1 \ldots X_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas Y_j (*Posterior*) adalah peluang munculnya kelas Y_j (sebelum masuknya sampel tersebut, seringkali disebut *prior*), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas Y_j (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga *evidence*). Karena itu, rumus di atas dapat pula ditulis secara sederhana sebagai berikut:

$$Posterior = \frac{prior \times likelihood}{evidence} \quad (16)$$

Nilai *Evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai dari *posterior* tersebut nantinya akan dibandingkan dengan nilai-nilai *posterior* kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus *Bayes* tersebut dilakukan dengan menjabarkan ($Y_j|X_1,\ldots,X_n$) menggunakan aturan perkalian sebagai berikut:

$$P(Y_j \mid X_1, \ldots, X_n) = P(Y_j)P(X_1, \ldots, X_n \mid Y_j)$$
$$= P(Y_j)P(X_1 \mid Y_j)P(X_2, \ldots, X_n \mid Y_j, X_1)$$
$$= P(Y_j)P(X_1 \mid Y_j)P(X_2 \mid Y_j, X_1)P(X_3, \ldots, X_n \mid Y_j, X_1, X_2)$$
$$= P(Y_j)P(X_1 \mid Y_j)P(X_2 \mid Y_j, X1)P(X_3 \mid Y_j, X_1, X_2)P(X_4, \ldots, X_n \mid Y_j, X_1, X_2, X_3)$$
$$= P(Y_j)P(X_1 \mid Y_j)P(X_2 \mid Y_j, X1)P(X_3 \mid Y_j, X_1, X_2) \ldots P(X_n \mid Y_j, X_1, X_2, X_3, \ldots, X_{n-1})$$

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin kompleksnya faktor-faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk

dianalisa satu persatu. Akibatnya, perhitungan tersebut menjadi sulit untuk dilakukan. Disinilah digunakan asumsi independensi yang sangat tinggi (*naive*), bahwa masing-masing petunjuk (X_1, X_2, …, X_n) saling bebas (*independen*) satu sama lain. Dengan asumsi tersebut, maka berlaku satu kesamaan sebagai berikut:

$$P(X_i \mid X_j) = \frac{P(X_i \cap X_j)}{P(X_j)} = \frac{P(X_i)P(X_j)}{P(X_j)} = P(X_i) \quad (17)$$

Untuk i ≠ j, sehingga

$P(X_i \mid Y, X_j) = P(X_i \mid Y_j)$

Dari persamaan diatas dapat disimpulkan bahwa asumsi independensi *naïve* tersebut membuat syarat peluang menjadi sederhana, sehingga perhitungan menjadi mungkin untuk dilakukan. Selanjutnya, penjabaran $P(Y_j \mid X_1, \ldots, X_n)$ dapat disederhanakan menjadi:

$$P(Y_j \mid X_1, \ldots, X_n) = P(Y_j)P(X_1 \mid Y_j)P(X_2 \mid Y_j)P(X_3 \mid Y_j)\ldots P(X_n \mid Y_j)$$
$$= P(Y_j)\prod_{i=1}^{n} P(X_i \mid Y_j) \quad (18)$$

Sehingga hasil klasifikasi merupakan *class* yang menghasilkan nilai probabilitas maksimum atau dapat dinyatakan dalam persamaan sebagai berikut:

$$Y_{MAP} = \arg\max_{Y_j \in Y} (P(Y_j)\prod_{i=1}^{n} P(X_i \mid Y_j)) \quad (19)$$

Keterangan:

$P(Y_j \mid X_1, \ldots, X_n)$: *Posterior Probability*

$P(X_i \mid Y_j)$: *Likelihood*

$P(Y_j)$: *Prior Probability*

Y_{MAP} : *Class* dengan *Maximum A Posterior Probability*

Persamaan diatas merupakan model dari teorema *Naïve Bayes* yang selanjutnya akan digunakan proses klasifikasi. Untuk klasifikasi dengan data kuantitatif atau kontinyu digunakan rumus *Densitas Gauss*:

$$P(X_i = x_i \mid Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}} \quad (20)$$

dimana:

P = Peluang
X_i = Atribut ke *i*
x_i = Nilai atribut ke *i*
Y = Kelas yang dicari
y_j = Sub kelas *Y* yang dicari
μ_{ij} = *Mean* sampel dari data *training* yang menjadi milik y_j
σ^2_{ij} = Varian sampel data *training*

3) *Confusi Matrix*

Menurut (Prasetyo, 2012) matriks konfusi merupakan tabel pencatat hasil klasifikasi. Umumnya, pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi (*confusson matrix*). Matriks konfusi merupakan tabel pencatat hasil kerja klasifikasi. Berikut ini merupakan hasil dari *confusion matrix* (Novandya & Oktria, 2017).

Tabel 1. Tabel Confusion Matrix

		Prediksi

Aktual		C1	C2
	C1	TP	FN
	C2	FP	TN

Keterangan :
TN = Jumlah prediksi yang tepat bersifat negatif (*True Negative*)
FN = Jumlah prediksi yang salah bersifat positif (*False Negative*)
FP = Jumlah prediksi yang salah bersifat negatif (*False Positive*)
TP = Jumlah prediksi yang tepat bersifat positif (*True Positive*)

Aktual merupakan klasifikasi status hujan yang sebelumnya telah diklasifikasikan terlebih dahulu. Prediksi merupakan hasil dari klasifikasi variabel status yang dihasilkan oleh program/software. Dari pembentukan konfusi matriks maka dapat dihitung beberapa nilai lain yang dapat dijadikan nilai kinerja klasifikasi (Faisal & Nugrahadi, 2019). Nilai-nilai tersebut sebagai berikut:

a. *Accuracy* merupakan proporsi klasifikasi benar melakukan prediksi. Rumus akurasi adalah:

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (21)$$

b. *Error Rate* merupakan proporsi klasifikasi melakukan kesalahan prediksi, dengan perhitungan persamaan sebagai berikut:

$$ErrorRate = \frac{FP + FN}{TP + TN + FP + FN} \quad (22)$$

3. Hasil dan Pembahasan

1) Analisis Deskriptif

Pada analisis deskriptif dapat diketahui bahwa curah hujan, temperatur, kelembapan udara, lama penyinaran matahari, dan kecepatan angin setiap harinya berbeda-beda atau terjadi fluktuasi yang artinya tidak menetap ataupun tidak secara signifikan naik dan turunnya semua variabel.

2) *Support Vector Machine*

Analisis SVM menggunakan model kernel *Linier*, dengan nilai *C* (Cost) yang digunakan dalam analisis ini yaitu 0.01, 0.1, 1, 10, dan 100. Nilai-nilai tersebut akan diterapkan pada data *training* untuk mencari nilai alpha atau *support vector*. Hal ini dapat disajikan pada Tabel 2.

Tabel 2. Nilai *C* Model Kernel Linier

Nilai Cost	Jumlah Support Vector
0,01	487
0,1	488
1	492
10	492
100	496

Berdasarkan Tabel 2. Dapat dilihat bahwa semua nilai cost 0.01, 0.1, 1, 10, dan 100 menghasilkan jumlah *support vector* yang berbeda yaitu 487, 488, 492, 492, dan 496. Sehingga model yang terbaik akan dilihat dari nilai cost yang paling rendah yaitu nilai cost 0.01, maka nilai alpha model kernel dengan parameter c = 0.01 dapat dilihat pada Tabel 3 sebagai berikut:

Tabel 3. Nilai Alpha Linier C =0,01

No	Alpha	Data Ke
1	0,01	1
2	0,01	2
3	0,01	3
…	…	…
…	…	…
…	…	…
479	0,01	648

Dari nilai alpha di atas dapat membantu untuk menemukan nilai bias, sehingga diperoleh nilai bias yaitu 0,9999137. Setelah diperoleh nilai alpha (α) dan bias, maka dapat dilakukan prediksi terhadap data testing sebagai berikut:

Tabel 4. Prediksi Metode SVM

No	$\sum_{i=1}^{n} a_i y_i K(x_i, x_j^T) + b$	$f(x) = sign\left(\sum_{i=1}^{n} a_i y_i K(x_i, x_j^T) + b\right)$	Status
1	-12329422037518	-1	Hujan
2	-12054959987476	-1	Hujan
3	-12661963414482	-1	Hujan
…	…	…	
…	…	…	
…	…	…	
73	-12148035831902	-1	Hujan

Selanjutnya akan dilakukan perbandingan hasil prediksi dengan data aktual yang disajikan pada Tabel 5 sebagai berikut:

Tabel 5. Perbandingan hasil prediksi dengan data aktual SVM

No	**Aktual**	**Prediksi**
1	Hujan	Hujan
2	Hujan	Hujan
3	Hujan	Hujan
…	…	…
…	…	…
…	…	…
73	Tidak Hujan	Hujan

Setelah diperoleh perbandingan hasil prediksi dengan aktual maka dapat dibentuk tabel *confusion matrix* sebagai berikut:

Tabel 6. *Confusion Matrix*

		Prediksi	
		Hujan	Tidak Hujan
Aktual	Hujan	58	0
	Tidak Hujan	15	0

Dari tabel matriks konfusi diatas, maka dapat dihitung beberapa nilai lain yang dijadikan nilai kinerja klasifikasi yaitu nilai akurasi dan nilai *error* (kesalahan klasifikasi). Perhitungan nilai kinerja klasifikasi sebagai berikut:

1. $Akurasi = \frac{TP + TN}{TP + TN + FP + FN} = \frac{58 + 0}{58 + 0 + 15 + 0} = \frac{58}{73} = 0{,}7945$
2. $ErrorRate = \frac{FP + FN}{TP + TN + FP + FN} = \frac{15 + 0}{58 + 0 + 15 + 0} = \frac{15}{73} = 0{,}2055$

Berdasarkan hasil di atas diketahui bahwa tingkat akurasi dari kinerja klasifikasi metode SVM sebesar 0,7945 atau 79,45% dengan nilai kesalahan klasifikasi sebesar 0,2055 atau 20,55%.

4) Naïve Bayes Classifier

Pada analisis *naïve bayes classifier* akan disajikan nilai probabilitas kelas dan hasil prediksi kelas pada Tabel 7 sebagai berikut:

Tabel 7. Nilai Probabilitas

No	Hujan	Tidak Hujan	Status
1	0,9364	0,0636	Hujan
2	0,2267	0,7733	Tidak Hujan
3	0,4945	0,5055	Tidak Hujan
…	…	…	…
…	…	…	…
…	…	…	…
73	0,4376	0,5624	Tidak Hujan

Berdasarkan Tabel 7. Dapat dilihat pada data pertama terdapat nilai probabilitas status hujan sebesar 0,9364 sedangkan nilai probabilitas status tidak hujan sebesar 0,0636, karena 0,9364 (hujan) > 0,0636(tidak hujan) sehingga diperoleh keputusan bahwa tidak hujan. Pada data kedua diperoleh nilai probabilitas status hujan sebesar 0,2267 sedangkan nilai probabilitas status tidak hujan sebesar 0,7733, karena 0,2267 (hujan) < 0,7733 (tidak hujan) sehingga diperoleh keputusan bahwa tidak hujan. Lalu data ketiga diperoleh nilai probabilitas status hujan sebesar 0,4945 sedangkan nilai probabilitas status tidak hujan sebesar 0,5055, karena 0,4945 (hujan) < 0,5055 (tidak hujan) sehingga diperoleh keputusan bahwa tidak hujan. Demikian interpretasi selanjutnya.

Pada tahap selanjunya akan dilakukan perbandingan hasil prediksi dengan data aktual yang disajikan pada Tabel 4.9 sebagai berikut:

Tabel 9. Perbandingan hasil prediksi dengan data aktual NBC

No	Aktual	Prediksi
1	Hujan	Hujan
2	Tidak Hujan	Tidak Hujan
3	Tidak Hujan	Tidak Hujan
…	…	…
…	…	…
…	…	…
73	Hujan	Tidak Hujan

Setelah diperoleh perbandingan hasil prediksi dengan aktual maka dapat dibentuk tabel *confusion matrix* sebagai berikut:

Tabel 10. *Confusion Matrix*

		Prediksi	
		Hujan	Tidak Hujan
Aktual	Hujan	27	16

	Tidak Hujan	9	21

Dari tabel matriks konfusi diatas, maka dapat dihitung beberapa nilai lain yang dijadikan nilai kinerja klasifikasi yaitu nilai akurasi dan nilai *error* (kesalahan klasifikasi). Perhitungan nilai kinerja klasifikasi sebagai berikut:

1. $Akurasi = \frac{TP + TN}{TP + TN + FP + FN} = \frac{27 + 21}{27 + 21 + 9 + 16} = \frac{48}{73} = 0{,}6575$
2. $ErrorRate = \frac{FP + FN}{TP + TN + FP + FN} = \frac{9 + 16}{27 + 21 + 9 + 16} = \frac{25}{73} = 0{,}3425$

Berdasarkan hasil diatas diketahui bahwa tingkat akurasi dari kinerja klasifikasi metode naïve bayes sebesar 0,6575 atau 65,75% dengan nilai kesalahan klasifikasi sebesar 0,3425 atau 34,25%.

5) Perbandingan Nilai Akurasi

Berdasarkan hasil analisis nilai kinerja klasifikasi kedua metode, maka nilai akurasi akan dibandingkan, metode klasifikasi yang terbaik akan digunakan untuk memprediksi data testing. Berikut dapat dilihat hasil akurasi kedua metode pada Tabel 11.

Tabel 11. Perbandingan Nilai Akurasi

Metode	Nilai Akurasi
Support Vector Machine	79,45%
Naïve Bayes Classifier	65,75%

Berdasarkan Tabel 4.22. Dapat dilihat bahwa metode yang memiliki nilai akurasi terbesar yaitu metode *Support Vector Machine* dengan nilai akurasi 0,7945 atau 79,45%. Maka metode SVM layak digunakan untuk memprediksi data testing.

4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka kesimpulan yang didapatkan adalah sebagai berikut:

1) Kondisi curah hujan di Jakarta Utara pada tahun 2017-2018 bahwa sering terjadi hujan hal ini dibuktikan dengan angka kejadian hujan pada tahun 2017 yaitu 235 dari 365 hari. Sedangkan pada tahun 2018, angka kejadian hujan yaitu 239 dari 365 hari.
2) Prakiraan curah hujan kedua metode menghasilkan tingkat akurasi yang berbeda, metode *Naïve Bayes Classifier* menghasilkan tingkat akurasi 65,75% dengan prakiraan angka kejadian hujan sebanyak 27 dari 73 hari pada data *testing*. Sedangkan metode *Support Vector Machine* menghasilkan prakiraan hujan sebanyak 58 dari 73 hari pada data *testing* dengan tingkat akurasi 79,45%.
3) Berdasarkan tingkat akurasi yang dihasilkan kedua metode bahwa metode terbaik yaitu *Support Vector Machine* hal ini dibuktikan dengan tingkat akurasi sebesar 79,45 % lebih besar dari tingkat akurasi metode *Naïve Bayes Classifier* yaitu 65,75%.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada dosen pembimbing Bapak Drs. Yudi Setyawan, M.S.,M.Sc dan dosen pembimbing II Ibu Noviana Pratiwi, S.Si., M.Sc yang telah memberikan banyak waktu, tenaga, dan pikiran mulai dari awal sampai akhir laporan penelitian ini. Penulis juga mengucapkan terima kasih kepada kedua orang tua, saudara, serta teman-teman penulis yang selalu memberi dukungan, dorongan, serta doa dari awal sampai akhir penyusunan penelitian ini.

Daftar Pustaka

[1] Asiyah, S. N., & Fithriasari, K. (2016). Klasifikasi Berita Online Menggunakan Metode Support Vector Machine dan K-Nearest Neighbor. *Jurnal Sains & Seni*, 317-322.

[2] Faisal, M., & Nugrahadi, D. (2019). *Belajar Data Science Klasifikasi dengan Bahasa Pemrograman R.* Banjarbaru, Kalimanta Selatan: Scripta Cendekia.

[3] Fridayanthie, E. W. (2015). Analisa Data Mining Untuk Prediksi Penyakit Hepatitis Dengan Menggunakan Metode Naive Bayes dan Support Vector Machine. *Jurnal Khatulistiwa Informatika*, 24-36.

[4] Mujiasih, S. (2011). Pemanfaatan Data Mining Untuk Prakiraan Cuaca. *BMKG*, 189-195.

[5] Nello, C., & Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.* New York: Cambridge University Press.

[6] Novandya, A., & Oktria, I. (2017). Penerapan Algoritma Klasifikasi Data Mining C4.5 Pada Dataset Cuaca Wilayah Bekasi. *Jurnal Format*, 98-106.

[7] Novianti, F., & Purnami, S. (2012). Analisis Diagnosis Pasien Kanker Payudara Menggunakan Regresi Logistik dan Support Vector Machine (SVM) Berdasarkan Hasil Mamografi. *Jurnal Sains & Seni*, 147-152.

[8] Nugroho, Satrio, A., & Witarto. (2003). Support Vector Machine Teori dan Aplikasinya dalam Bionformatika. *Jurnal Ilmu Komputer*, 1-11.

[9] Octaviani, P., Wilandari, Y., & Ispriyanti, D. (2014). Penerapan Model Klasifikasi Support Vector Machine (SVM) Pada Data Akreditasi Sekolah Dasar di Kabupaten Magelang . *Jurnal Gaussian*, 811-820.

[10] Prasetyo. (2012). *Data Mining Konsep dan Aplikasi Menggunakan Matlab,*. Yogyakarta: Andi Offset.

[11] Prawaka, F., Zakaria, A., & Tugiono, S. (2016). Analisis Data Curah Hujan Yang Hilang Dengan Menggunakan Metode Normal Ratio, Inversed Square Distance, dan Rata-rata Aljabar . *JRSDD*, 397-406.

[12] Riadi, I., Umar, R., & Aini, F. (2019). Analisis Perbandingan Detection Traffic Anomaly dengan Metode Naive Bayes dan Support Vector Machine (SVM). *Jurnal Ilmiah*, 17-24.

[13] Ridwan, M., Suyono, H., & Sarosa, M. (2013). Penerapan Data Mining Untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier. *Jurnal EECCIS*, 59-64.

[14] Saleh, A. (2015). Implementasi Metode Klasifikasi Naive Bayes Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga. *Journal Citec*, 207-217.

[15] Scholkopf, B., & Smola, A. (2002). *Learning With Kernels.* Cambridge, Massachusetts, London, England: The MIT Press.

[16] Vijayarani, S., & Dhayanand, S. (2015). Prediksi penyakit hati menggunakan algoritma SVM dan Naive Bayes. *International Journal of Science, Engineering and Technology Research (IJSETR)*, 816-820.