BAB II

TINJAUAN PUSTAKA

Bagian ini berisi landasan teori terkait teori-teori yang digunakan dan pekerjaan yang sudah dilakukan oleh penelitian sebelumnya untuk mendukung penyelesaian penelitian yang akan dilakukan.

II.1 *Machine Learning*

II.1.1 Pengeritan *Machine Learning*

Machine Learning sering digambarkan sebagai pembelajaran dari pengalaman atau tanpa pengawasan dari manusia. Dalam masalah belajar yang diawasi, sebuah program memprediksi output untuk input dengan belajar dari pasangan input dan output berlabel; yaitu, program belajar dari contoh jawaban yang benar. Dalam tanpa pengawasan belajar, suatu program tidak belajar dari data berlabel *Machine Learning* mengeksplorasi studi dan konstruksi algoritma yang dapat belajar dari dan membuat prediksi pada data. Algoritma tersebut beroperasi dengan membangun model dari input .Contoh untuk membuat prediksi berbasis data atau keputusan, daripada mengikuti instruksi program yang benar-benar statis (Baharuddin, Azis, and Hasanuddin 2019).

Menurut *Expert System* pada tahun 2017, *machine learning* adalah aplikasi atau bagian dari kecerdasan buatan yang membuat sistem memiliki kemapuan belajar secara otomatis dan meningkatkan kemampuannya belajar secara otomatis dan meningkatkan kemampuan berdasarkan pengalaman tanpa program komputer dan ditulis secara statis. *Machine larning* juga didefenisikan sebagai algoritma yang bertujuan menemukan dan mengaplikasikan pola-pola di dalam

data(Hao,2018).

Machine Learning merupakan bagian dari kecerdasanbuatan dimana *machine learning* bertujuan untuk memahami atau mengenali struktur suatu data dan mengkonversi data tersebut ke dalam suatu model (Tagliafferi,2017).

Berdasarkan paparan diatas *machine learning* dapat didefinisikan sebagai suatu algoritma atau program komputer yang data membuat sistem menjadi cerdas dengan mempelajari data-data yang tersedia dimana algoritma atau program tersebut tidak didefenisikan eksplisit (Purba Daru Kusuma,2020).

II.1.2 Klasifikasi *Machine Learning*

Machine learning berdasarkam cara belajarnya dibagai menjadi tiga kelompok yaitu (Ibnu Daqiqil Id,2021) :

1. *Supervised Learning*

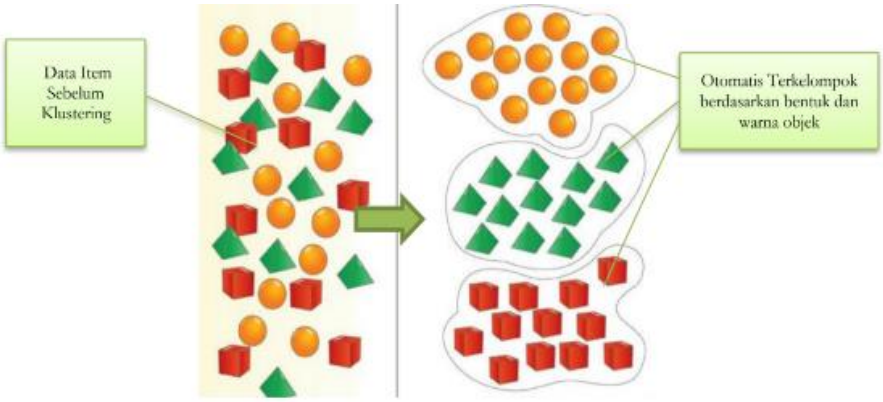
   Supervised leaning adalah pembelajaran terarah/terawasi, komputer atau mesin akan mempelajari data training yang berisi label. Selain klasifikasi , *regresi* juga dimasukkan sebagai pembelajaran terarah. Adapun contoh-contoh algoritma yang termasuk ke dalam kategori *supervised* adalah Linear *Regressuin (Regresi Linear), Logistic Regression (Regresi Logistik), Linear Discriminant Aalysis,k-Nearest Neighbors, Support Vector Machiner (SVMs), Radmom Forest, dan Neurol Network*

2. *Unsepervised Learning*

   Unsuvervised learning adalah kebalikan *supervised learning* dimana proses pembelajaran dilakukan untuk menemukan pola-pola didalam data.

Secara sistematis, *unsupervised learning* terjadi ketika kita memiliki sejumlah data masukan (X) dan tanpa variablel *output* yang berhubungan. Masalah *Unsupervise learning* dibagi menjadi dua jenis yaitu asosiasi dan *clustering*.

Sebagai contoh Gambar II.1 berisi dua jenis item yang berbeda. Kedua item tersebut akan dipisah menjadi beberapa kategori tergantung data.Komputer hanya mengetahui fitur-fitur yang akan digunakan untuk membedakan kedua item tersebut yaitu warna dan bentuk,Dengan menggunakan algoritma *clustering*, komputer akan membagai item-item memajdi dua kelompok tanpa harus diberi pengetahuan .Algoritma akan bekerja untk membagi menjadi beberapa kelompok dengan melihat isi data masing-masing item .Teknik-teknik mengelompokkan ini mengunakan algoritam k-means, *Self organing Maps* dan lain sebagainya.



Gambar II.1. Ilustrasi proses kluterisasi data

3. *Reunforcement Learning*

Machine learning berdasarkan cara kerjanya dapat dikelompokkan menjadi dua yaitu :

a. *Intance-based learning* (atau sering disebut memory-based laearning) adalah sebuah kelompok algoritma ML yang bekerja dengan membandingkan data testing dengan data yang telah dipelari pada proses traning. Algoritma tidak membuat sebuah generasi tetapi kea rah perbadingann dengan data yang disimpan dimemori.
b. *Model based learning* adalah kebalikan dari instance-based dimana menggunakan memori untuk melakukan pemecahan masalah, algoritma ini membuat sebuah model yang berisifat generik.

II.1.3 *Support Vector Machine* (SVM)

Support Vector Machine(SVM) adalah salah satu metode klasifikasi yang berdasarkan pada diskriminan linear margin maksimum. Adapun tujuan dari diskriminan linear margin maksimum adalah untuk mencari *hyperplane* dengan memaksimalkan jarak atau margin antar kelas. Cara terbaik untuk mencari *hyperplane* terbaik antara kedua kelas adalah dengan mengukur margin *hyperplane* dan kemudian mencari titik maksimalnya, yang diulistrasikan pada gambar II.4. Pada gambar (a) ada sejumlah pilihan *hyperlane* yang mungkin untuk set data. sedangkan gambar (b) merupakan *hyperlane* dengan margin yang paling maksimal. Meskipun sebenarnya pada gambar (a) bisa menggunakan *hyperlane* sembarang. tetapi *hyperlane* dengan margin yang maksimal akan memberikan generalisasi yang lebih baik pada metode klasifikasi. Margin merupakan jarak antara *hyperplane* dengan data terdekat dari masing- masing kelas. Data yang paling dekat ini disebut sebagai *support vector* (Chhajer, Shah, and Kshirsagar

2022).

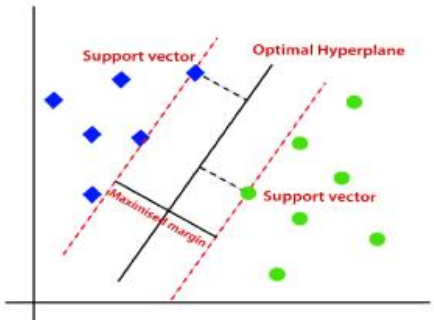
SVM merupakan *supervise learning*. Dengan demikian terdapat 2 jenis data yaitu data latih dan data uji.Data latih digunakan untuk menentukan pemisah antar kelompok data.Adapun data uji adalah data yang ditentukan klasifikasinya didalam data terdapat label yang merupakan identitas kelompok data.

1. *Hyperplane*

Hyperplane dari algoritma SVM didefinisikan sebagai batas keputusan terbaik dari berbagai kemungkinan batas keputusan yang secara akurat mengklasifikasikan kelas dalam ruang n-dimensi. Fitur dari kumpulan data menentukan dimensi dari hyperplane yaitu jika kumpulan data memiliki 2 fitur itu menyiratkan *hyperplane* satu dimensi sedangkan 3 fitur menyiratkan *hyperplane* dua dimensi. Bidang hiper yang memiliki margin maksimum, yang berarti jarak antara dua titik data maksimum, lebih disukai (Bansal, Goyal, and Choudhary 2022).

2. *Support Vector* (SV)

Support vector adalah jarak terdekat yang mempengaruhi posisi *hyperplane*. Karena sifatnya yang mendukung terhadap *hyperplane,* disebut sebagai vektor pendukung(Bansal, Goyal, and Choudhary 2022) .

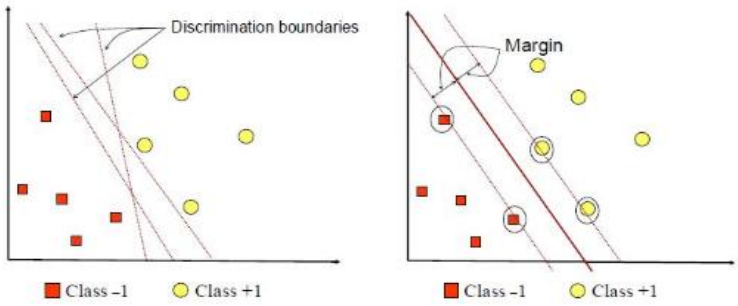


Gambar II.2 Implementasi SVM (Bansal, Goyal, and Choudhary 2022)

Menurut (Bansal, Goyal, and Choudhary 2022) ada 2 kasus dalam *Support Vector Machine*(SVM) yaitu:

1. Kasus data yang tepisah secara linear

   Kasus data yang terpisah secara linear adalah kasus yang mempermudah seperti yang terlihat pada gambar berikut :



(a) (b)

Gambar II.3 SVM Berusaha menemukan *Hyperlane* pemisah *(Bansal, Goyal, and Choudhary 2022)*

Gambar II.3 memperlihatkan beberapa pattern yang merupakan dari duah buah *class* +1 dan -1 yang mempunyai tupel pelatihan 2-D. *Pattern* yang bergabung pada *class* + 1 disimbolkan dengn lingkaran berwarna kuning. Masalah klasifikasi dapat diterjemahkan dengan usaha menemukan hyperline yang memisahkan

antara kedua kelompok tersebut. *Hyperlane* pemisah terbaik antara kedua class dapat menemukan dengan mengukur margin *hyperlane* dengan pattern terdekat masing-masing class. *Pattern* yang paling dekat ini disebut *support vector*. Garis solid pada gambar II.3 . Sebelah kanan menunjukkan *hyperlane* terbaik. yaitu yang terletak pada tengah-tengah kedua *class*. sedangkan titik merah dan kuning yang berada dalam lingkaran hitam adalah *support vector*. Data yang tersedia dinotasikan sebagai x€ R^d sedangkan label masing-masing dinotsikan $y_{i \in} \{-1, +1\}$ untuk i=1. 2. …. n yang mana n adalah banyaknya data. Diasumsikan kedua class dapat terpisah secara sempurna oleh *hyperplane* berdimensi d. yang didefenisikan pada persamaan 2.1.

$$W.x + b = 0 \quad (2.1)$$

Pattern x_1 yang terdapat pada class -1 dapat dirumuskan sebagai pattern yang memenuhi persamaan 2.2.

$$W.x + b \leq -1 \quad (2.2)$$

Pattern x_i yang terdapat pada class -1 dapat dirumuskan dengan persamaan 2.3.

$$W.x + b \geq 1 \quad (2.3)$$

Keterangan :

W = *Vector* bobot

x = nilai masukan atribut

b = bias

Margin terbesar dapat ditemukan dengan memaksimalkan nilai jarak antar jarak dan titik terdekatnya. yaitu 1 / || W ||. Hal ini dapat dirumuskan sebagai *quadratic programming* (QP) *probem* yaitu mencari titik minimal yang dinyatakan dalam persamaan dan memperhatikan kondisi yang harus dipenuhi pada persamaan 2.4 dan persamaan 2.5.

$$min_w = r(w) = \frac{1}{2} || W ||^2 \quad (2.4)$$

$$y_i (x_i . w + b) - 1 \geq 0 \quad (2.5)$$

Keterangan :

x_i = Data input

y_i = Keluaran dari data x_i. w n. b

w. b = Paramater yang dicari nilainya.

Problem ini dipecahkan dengan teknik komputasi. diantaranya *lagrange multiplier tang* dinyatakan pada persamaan 2.6.

$$\text{L (w. b. a)} = \frac{1}{2 || w ||^2} - \sum_{i=1}^{l} \propto_i \; y_i (x_i . w + b) - 1 \text{ dengan t} = 1,2 \ldots \text{l} \quad (2.6)$$

Dimana a_1 adalah *lagrange multiplier* yang bernilai 0 atau $a_1 \geq 0$. nilai optimal dari persamaan 2.6 dapat dihitung dengan meminimalkan L terhadap w dan b dan memaksimalkan L terhadap a1. Dengan memperhatikan sifat bahwa pada titik *gradient* L=0 persamaan 2.5 dapat dimodifikasi sebagai masksimasi *problem* yang hanya mengandung a1 sebagimana terlihat pada persamaan 2.7 dan 2.8.

$$\sum_{i=1}^{l} a1 - \frac{1}{2} \sum_{i,j=0}^{l} = 1 a_i \; a_j \; y_i \; y_j x_i \; x_j \quad (2.7)$$

$$\sum_{i}^{l} a_1 \; y_i = 0 \; untuk \; a_i \geq 0 (i = 1,2, \ldots l) \quad (2.8)$$

Dengan demikian. maka akan diperoleh a_1 yang kebanyakan bernilai positif yang

disebut sebagai *support Vector*.

2. Kasus Data yang tidak terpisah secara linear

Kasus data yang tidak terpisah secara linear diasumsikan bahwa *class* pada input space tidak dapat terpisah secara sempurna. Hal ini menyebabkan constraint pada persamaan 2.3 tidak dapat terpenuhi. Sehingga Optimasi tidak dapat dilakukan. Untuk mengatasi masalah ini SVM dirumuskan ulang dengan memperkenalkan teknik softmargin. Dalam *softmargin* persamaan 2.5 dimodifikasi dengan menggunakan *slack* variabel sehingga pada persamaan 2.9.

$$y_i(x_i w + b) \geq 1 - \varepsilon_i \quad (2.9)$$

Dengan demikian persamaan 2.7 diubah menjadi persamaan 2.10

$$mtn_w = \tau(w) = \frac{1}{2||w||^2} + C \sum_{i=1}^{l} \varepsilon_i \quad (2.10)$$

Fitur C digunakan untuk mengontrol *tradeoff* antara margin dan kesalahan klasifikasi ε.

a. *Kernel Trick* dan *Non-Linear Classification* pada SVM

Pada umumnya masalah yang terjadi dalam dunia nyata jarang bersifat linear separable. Kebanyakan bersifat *non-linear*. SVM dimodifikasi dengan memasukkan fungsi kernel. Dalam *non linear* SVM. pertama-tama data x dipetakan oleh fungsi ϕ (x) keruang vektor yang berdimensi lebih tinggi. Pada ruang vektor yang baru ini. hyperlane yang memisahkan kedua class tersebut dapat dikonstruksikan. Hal ini sejalan dengan teori cover yang menyatakan “Jika suatu transformasi bersifat non linear dan dimensi dari *feature space* cukup tinggi. maka data pada input *space* dapat dipetakan ke *feature space* yang baru. dimana *pattern-pattern* tersebut pada probabilitas tinggi dapat dipisahka secara linear”.

Pemetaan ini dilakukan dengan menjaga topologi data. dalam artian dua data yang berjarak dekat pada input *space* akan berjarak dekat juga pada *feature* space. sebaliknya dua data yang berjarak jauh pada input *space* akan juga berjarak jauh pada *feature space*. Selanjutnya proses pembelajaran pada SVM dalam menemukan titik-titik *support vektor*. hanya bergantung pada dot *product* dari data yang sudah ditransformasikan pada ruang baru yang berdimensi lebih tinggi. yaitu $\phi\ (x_i).\ \phi\ (x_j)$. Karena umumnya transformasi ϕ ini tidak diketahui. dan sangat sulit untuk difahami secara mudah. maka perhitungan dot product tersebut sesuai teori Mercer dapat digantikan dengan *kernel* yang terlihat pada persamaan 2.11.

$$\mathrm{K}(x_i\ x_j\ ) = \phi\ (x_j\ ).\phi\ (x_j) \qquad (2.11)$$

Keterangan :

x_i = *Support Vector*

Beberapa kernel yang terdapat pada SVM meliputi: (Prasetyo. 2014).

1. Polinominal Derajat h

   Kernel trick polinominal cocok digunakan untuk menyelesaikan masalah klasifikasi. dataset pelatihan sudah normal. *Kernek trick* ini dinyatakan dalam persamaan 2.12.

$$K(x_i\ x_j) =\ (x_i\ x_j + 1)^h \qquad (2.12)$$

2. *Radial Basis Function*

   Kernel trick radial basis *function* merupakan kernel yang paling banyak digunakan untuk menyelesaikan masalah klasifikasi untuk dataset yang tidak terpisah secara linear. dikarenakan akurasi pelatihan dan akurasi prediksi yang

sangat baik pada kernel ini. dimana kernel radial basis *function* dinyatakan dalam persamaan 2.13.

$$K(x_i, x_j) = e^{-||x_i - x_i||^2 / 2a^2} \quad (2.13)$$

Dimana: X_i dan X_j = Pasangan dua data training.

3. *Sigmoid* (tangen hiperbolik)

Kernel sigmoid merupakan *kernel trick. Support Vektor Machine* yang merupakan pengembangan dari jaringan saraf tiruan. dimana *kernel* ini dinyatakan dengan persamaan 2.14.

$$K(x_i, x_j) = \tan(kx_i . x_j - \delta) \quad (2.14)$$

Kernel trick memberikan beberapa kemudahan. karena dalam proses pembelajaran SVM. untuk menentukan *support vecktor*. pengguna hanya cukup mengetahui fungsi kernek trick yang dipakai. tanpa perlu mengetahui wujud dari fungsi non-linier. Dari keseluruhan kernel trick tersebut. kernel trick radial basis function merupakan kernel trick yang memberikan hasil terbaik pada klasifikasi khususnya untuk data yang tidak bisa dipisahkan secara linear. Selain masalah data yang tidak dipisahkan secara linear ada masalah lain yang sering muncuk dalam penerapan metode klasifikasi *machine learning* seperti *support vector machine* adalah masalah dimensionalitas dataset atau sering disebut sebagi kutukan dimensionalitas (*curse of dimensionality*). jika dimensi meningkat. data kan meningkat secara halus dalam daerah yang ditempati. Untuk itu diperlukan pengurangn dimensi.

Manfaat dari pengurangan dimensi:

1. Mencegah terjadinya efek dari dimensionalitas.

2. Mengurangi jumlah waktu dan memori yang dibutuhkan oleh machine learning.
3. Membuat data lebih mudah divisualisasikan.
4. Membantu untuk mengurangi fitur-fitur yang tidak relevan

Teknik pengurangan dimensionalitas data diantaranya adalah principal component analysis (PCA). standar deviasi. *zero-mean. min-max normalization*. dan lain-lain.

a. Standar Deviasi

Standar deviasi disebut juga simpangan baku merupakan metode untuk mencari variasi suatu data dan merupakan metode yang digunakan untuk mengurangi dimensionalitas dari suatu dataset dan mempunyai satuan ukuran yang sama dengan data asal. Singkatnya. standar deviasi mengukur-bagaimana nilai-nilai data tersebar. bisa juga didefinisikan sebagai rata-rata jarak penyimpanan titik-titik. Standar deviasi merupakan hasil akar dari pengurangan dataset dengan nilai rata-rata dari dataset dibagi dengan jumlah dataset. Standar deviasi ini ditulis dengan persamaan 2.15.

$$S = \sum \sqrt{\frac{|x_i - x|^2}{n-1}} \qquad (2.15)$$

keterangan :

x = Rataan hitung

x_i = Input data

N = Jumlah data

s = Standar deviasi.

II.1.4 *Self Orginizing Map* (SOM)

SOM merupakan salah satu teknik dalam *Neural Network* yang bertujuan untuk melakukan visualisasi data dengan cara mengurangi dimensi data melalui penggunaan *self-organizing neural networks* sehingga manusia dapat mengerti high-dimensional data yang dipetakan dalam bentuk *low-dimensional* data.

Pada algoritma SOM. vektor bobot untuk setiap unit cluster berfungsi sebagai contoh dari input pola yang terkait dengan *cluster* itu. Selama proses *self-organizing*. *cluster* satuan yang bobotnya sesuai dengan pola vektor input yang paling dekat (biasanya. kuadrat dari jarak Euclidean minimum) dipilih sebagai pemenang. Unit pemenang dan unit tetangganya (dalam pengertian topologi dari unit cluster) terus memperbarui bobot merek (Sinaga *et. al*,. 2019). Setiap output akan bereaksi terhadap pola input tertentu sehingga hasil Kohonen SOM akan menunjukkan adanya kesamaan ciri antar anggota dalam cluster yang sama.

Ada 2 langkah dalam proses pemberian bobot w_i. (Srisawat *et al*. 2006). Pertama. algoritma *Self Organizing Map* digunakan untuk mempartisi semua SV menjadi *pre-defined* K *cluster*. Jika lebih dari satu label kelas muncul dalam sebuah cluster. intansi dengna kelas mayoritas akan lebih dapat dipercaya dari pada instansi dengan kelas minoritas. Setelah data dikelompokkan. label kelas dari semua SV dijadikan pertimbangan. Dalam langkah ini. setiap instansi dalam cluster diberikan bobot menurut proporsi dari label kelas instansi dalam cluster. Bobot sampel x_i yang dinyatakan dengan. diberikan dalam persamaan 2.16.

$$w_i = \frac{n(class(x_i))}{Total} \tag{2.16}$$

i= 1. 2. …. m (m adalah jumlah semua SV). sedangkan n(class(x_i)) adalah jumlah sampel dalam cluster yang mempunyai label kelas yang sama dengan sampel x_i. Total adalah jumlah total sampel dalam cluster.

Proses prediksi pada data uji dilakukan dengan algoritma K-NN klasik. Akan tetapi hanya SV terbobotlah yang dihitung dalam proses sebelumnya yang digunakan sebagai data latih K-NN. Ketika data uji dimasukkan. K-NN mencari sejumlah K instansi terdekat dari set SV. Bobot SV dengan kelas yang sama dijumlahkan dan labe dengan bobot maksimal dihasilkan sebagai label kelas untuk data uji. .(Srisawat *et al*. 2006).

Proses dalam Pengemlompokan SOM melibatkan dua proses utama yaitu inisialiasi dan pelatihan. Prosesnya dijelaskan dalam 5 langka sebagai berikut:

a. Inisialiasi

Bobot awal vektor masukkan SOM diinisialiasi dengan menggunakan metode berbasis acak. Selain itu. ukuran peta ditentukan dengan menggunakan lebar dan tinggi masukan parameter sesuai kisi-kisi dimensi.

b. Unit Pencocokan Terbaik (BMU)

BMU diidentifikasi melalui simpul terdekat jarak untuk lingkungan dengan jarak Euclidean. Persamaan untuk jarak Eucludean. pada persamaan 2.17

$$d_{ij}\sqrt{\sum_{k=1}^{n}(X_{ki} - X_{kj})^2} \qquad (2.17)$$

Dimana : d_{ij} = jarak antara instan i dan j

X_{ki}= nilai contoh j dan j

c. Berat diperbaharui (*Weight Updated*)

Bobot untuk node diperbaharui dalam BMU's *radius* lingkungan dengan demikian. ukuran linkungan sekitar menyusut untuk setiap itersi menggunakan Gaussian. didefnisikan pada persamaan 2.18.

$$\sigma\,(t) = \sigma_0 \ \exp(-\frac{t}{\pi}\,.t) = 1,2,3,\ldots. \qquad (2.18$$

Dimana $\sigma\,(t)$ = Jari-jari pada waktu t_0

t = Langkah waktu saat ini. t = 1. 2. 3

π = Waktu konstan.

Nilai Update neuron didefinisikan pada Persamaan 2.19.

$$Wi' = W2 + \sigma\,(t)\ Wi\ (Xi\text{-}Wi). \qquad (2.19)$$

Dimana : Wi adalah bobot. Xi = neuron

d. Pengulangan

Cluster yang bagus terbentuk dari jarak minimum kohesi dan maksimal jarak jarak pemisah percobaan dan hasil eksperimen untuk penelitain ini dibahas pada bagian selanjutnya.

II.1.5 *K-Nearest Neigbor* (K-NN)

K-Nearest Neighbor (k-NN atau KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran (neighbor) yang jaraknya paling dekat dengan objek tersebut. Dekat atau jauhnya neighbor biasanya dihitung berdasarkan jarak *Euclidean*. diperlukan suatu sistem klasifikasi sebagai sebuah sistem yang mampu mencari informasi . Metode KNN dibagi menjadi dua fase, yaitu pembelajaran (training) dan klasifikasi atau

pengujian (*testing*). Pada fase pembelajaran, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk data yang akan diuji coba (yang klasifikasinya tidak diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor data pembelajaran dihitung, dan sejumlah k buah neighbor yang paling dekat diambil (Liu,2022).

Sesuai dengan prinsip kerja *K-Nearest Neighbor* yaitu mencari jarak terdekat antara data yang akan dievaluasi dengan k tetangga (*neighbor*) terdekatnya dalam data pelatihan. Persamaan dibawah ini menunjukkan rumus perhitungan untuk mencari jarak terdekat dengan d adalah jarak dan p adalah dimensi data (Zhou, X., Wang, H., Xu, C. et al,2022), didefinikan pada persamaan 2.20.

$$d_i \sqrt{\sum_{i=1}^{p} (x_{2i-} \; x_{1i} \;)^2} \qquad (2.20)$$

Keterangan:

x_1 = Sampel data

x_2 = Data uji

I = Variabel data

D = Jarak

P = Dimensi data.

II.1.6 *Local Mean Based K-Nearest Neighbor* (LKMNN *Classifier*)

Klasifikasi *K- Nearest Neighbor* (KNN) adalah teknik non-parametrik yang sangat terkenal dan sederhana dalam klasifikasi pola. Tetapi kalsifikasinya

mudah dipengaruhi oleh outlier yang ada, khususnya dalam situasi ukuran sampel yang kecil. *Local Mean Based K- Nearest Neighbor* (LMKNN) adalah Sebagai perpanjangan dari KNN (Liu,2022).

LMKNN ini telah terbukti meningkatkan kinerja klasifikasi dan juga mengurangi efek outlier yang ada, terutama dalam ukuran data yang kecil, Proses LMKNN dapat digambarkan sebagai berikut (Liu,2022).

Langkah 1 : Penentuan Nilai K

Langkah 2 : Hitung jarak antara data uji dengan masing-masing semua data pelatihan menggunakan jarak *Euclidean* menggunakan persamaan 2.21.

$$D(x,y) = ||x-y||_2 = \sqrt{\sum_{j=1}^{N} |x-y|^2} \quad (2.21)$$

Langkah 3 : Urutkan jarak data dari yang terkecil ke yang terbesar sebanyak K untuk setiap kelas data

Langkah 4 : Hitung vektor rata-rata lokal dari setiap kelas dengan persamaan 2.22.

$$w_c = argmin_{wj}\left(x, m_{w_j}^k\right), j = 1,2,\dots,M \quad (2.22)$$

Langkah 5 : Tentukan kelas data uji dengan menghitung jarak terdekat ke vektor rata-rata lokal dari setiap kelas data menggunakan persamaan 2.23.

$$m_{w_j}^k = \frac{i}{k} \sum_{i=1}^{k} yi^N, j^N \quad (2.23)$$

Klasifikasi LMKNN sama dengan 1-NN jika nilai k = 1 . Nilai K dalam LMKNN berbeda dari kNN asli, dalam kNN asli nilai K adalah jumlah tetangga terdekat dari semua data pelatihan, sedangkan di LMKNN nilai K adalah jumlah tetangga

terdekat dari setiap kelas dalam data pelatihan. Dalam penentuan kelas dari data uji, LMKNN menggunakan pengukuran jarak terdekat dengan setiap vektor rata-rata lokal dari setiap kelas data, yang dianggap efektif dalam mengatasi efek negatif *outlier*.

II.2 Saham *(Stock)*

II.2.1 Teknik Prediksi Pasar Saham

Teknik prediksi pasar saham mempunyai dua pendekatan konvensional yang sering dipakai dan umumnya digunakan oleh para analisis dalam meramalkan harga saham kedepannya adalah memakai analisis teknikal dan fundamental.(Saini and Sharma 2022).

II.2.1.1 Analisis Teknikal

Analisis teknikal adalah teknik untuk memprediksi arah pergerakan harga saham dan indikator pasar saham lainnya berdasarkan pada data pasar historis seperti informasi harga dan volume. Penutup Analisis Teknikal kegiatan signifikan yang terkait dengan pergerakan harga saham seperti tren, pola, harga tertinggi, harga terendah, dan lain sebagainya. Kelebihan analisis teknikal adalah dapat dilakukan dengan cepat dan mudah. karena hanya perlu memberikan sinyal yang tepat untuk memulai atau berhenti dari proses trading saham (Saini and Sharma 2022) .

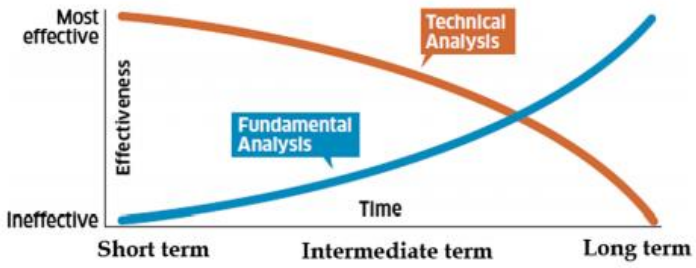
II.2.1.2 Analisis Fundamental

Analisis fundamental merupakan pendekatan analisis harga saham yang pada kinerja perusahaan yang mengeluarkan saham dan analisis ekonomi yang akan dipengaruhi masa depan perusahaan. Kinerja perusahaan dapat dilihat dari perkembangan perusahana. Kinerja perusahaan dapat dilihat dari perkembangan perusahaan. neraca perusahaan dan laporan laba ruginya. proyeksi usaha dan rencara perluasan dan kerjasama. Pada umumnya apabila kinerja perusahaan mangami perkembangan yang baik. maka harga saham akan meningkat (Saini and Sharma 2022).

Jika anda terjun dalam trading saham. juga akan mempelajari faktor fundamental suatu perusahaan seperti : pertumbuhan pendapatan. rasio laba terhadap saham. siklus bisnis. net profit margin. dan lain sejenisnya. Dengan memperhatikan data-data diatas. maka trader dapat memprediksikan apakah pergerakan harga akan naik atau turun. apakah saham layak untuk dibeli atau tidak. Analisa fundamental lebih cenderung *long term*/untuk jangka panjang. Analisis ini bekerja dengan melakukan penilaian terhadap kinerja dari perusahaan yang kita inginkan. Kinerja perusahaan tersebut akan dituliskan dalam bentuk laporan keuangan. (Saini and Sharma 2022).

Tabel II.1. Analisis antara pendekatan teknikal dan fundamental berdasarkan parameter yang berbeda (Saini and Sharma 2022)

Parameter	Analisis Teknikal	Analisis Fundamental
Deskripsi	Harga diprediksi dengan mengamati pada pergerakan harga sekuritas.	Harga diprediksi dengan menganalisis faktor-faktor ekonomi dan keuangan mempengaruhi bisnis.
Peralatan	Dalam analisis teknis, analisis dimulai dengan grafik saham, karena semuanya yang relevan keterangan sudah termasuk stok Harga.	Dalam proses analisis fundamental,Untuk analisis dimulai dari keuangan perusahaan laporan seperti laba rugi, saldo lembar, dan laporan arus kas.
Efektivitas waktu	Pendekatan jangka pendek	Pendekatan jangka Panjang
Perdagangan versus Investasi	Identifikasi jangka pendek hingga menengah pada perdagangan	Lakukan investasi jangka Panjang.



Gambar II.4. Prediksi pasar saham menggunakan algoritma pembelajaran mesin (Albahli et al. 2022)

II.2.2 Indikator Saham

Indikator saham merupakan indikator yang digunakan untuk melihat pergerakan harga saham. (Utami & Gusnarsih, 2019). Ada beberapa jenis indikator Saham antara lain sebagai berikut :

II.2.2.1 *Moving Average*

Moving Average adalah sebuah peramalan yang menggunakan rata-rata dan data aktual periode sebelumnya untuk meramalkan periode selanjutnya. Metode ini disebut rata-rata bergerak karena setiap kali data actual baru bersedia. maka data paling awal atau terdahulu diganti dengan data baru. lalu dihitung dan hasilnua digunakan sebagai ramalan permintaan untuk periode yang akan datang. Tujuan utama dari penggunaan rata-rata bergerak adalah untuk menghilangkan atau mengurangi variasi acak permintaan dalam hubungannya dengan waktu. (Habis, 2008).

Apabila data aktual yang diperoleh memperlihatkan permintaan produk tersebut tidak mengalami peningkatan maupun penurunan serta tidak mempunyai

karakteristik musiman. maka *moving average* bergunak untuk menghilangkan fluktuasi acak pada peramalan tersebut. Rumus matematis *moving average* dinyatakan dalam persamaan sebagai berikut. (Habib, 2008). Pada persamaan 2.24 dan persamaan 2.25.

$$F_t = \frac{\sum Permintaan\ dalam\ periode\ n\ sebelumnya}{n} \quad (2.24)$$

$$F_t = \frac{(A_{t-1}+A_{t-2}+\cdots+A_{t-n}}{n} \quad (2.25)$$

Keterangan :

Ft = Nilai peramalan untuk periode berikutnya

A_{t-1} = Nilai permintaan actual periode sebelumnya

n = Jumlah periode yang digunakan.

II.2.2.2 *Exponential Moving Averange* (EMA/XMA)

Indikator *exponential moving average* (EMA/XMA) sebagai salah satu varian dari *moving average* (MA). menggunakan formulasi perhitungan yang memberikan bobot pada harga sekarang secara *relative* terhadap harga awal dari perhitungan *exponential moving average* pada rentang waktu tertentu. Semakin pendek rentang waktu yang digunakan. semakin berbobot penerapan penggunaan *exponential moving average* (EMA/XMA) ini untuk memberikan nilai rata-rata terkini dari sekuritas (Widodo & Hansun. 2015).

XMA/EMA pada persamaaan 2.26 sampai persamaan 2.27.

$$XMA = K * (C-P) + P \quad (2.26)$$

$$K = \frac{2}{n+1} \quad (2.27)$$

Keterangan :

K = *Smoothing constant*

C = *current close price* (Harga penutupan saat ini)

P = *previous* XMA (XMA sebelumya)

N = jumlah periode XMA

Smoothing constant adalah variabel yang digunakan dalam analisis deret waktu berdasarkan pemulusan eksponensial. Konstanta ini menentukan bagaimana nilai-nilai seri waktu historis tertimbang.

$$SMA = \frac{\sum_{t=1}^{t=n} Close_n}{n} \quad (2.28)$$

Keterangan :

$\sum_{t=1}^{t=n} Close_n$ = Total harga *close* (tutup)

n = Jumlah periode pengamatan.

Khusus pada indikator ini nilai *previous* XMA setiap periode tidak memiliki nilai. Sehingga nilai SMA yang mana nilai dari SMA adalah nilai rata-rata dari periode hari yang digunakan. Sedangkan pada proses selanjutnya *provious* XMA menggunakan hasil XMA sebelumnya. Berikut ini adalah formula untuk SMA.

II.2.2.3 *Stochastic Oscillator* (STI)

Stochatic Oscilator adalah sebuah alat analisis yang pertama kali kembangkan oleh George C Lane pada akhir 1950an. Indikator ini bekerja berdasarkan observasi bahwa pada saat naik. harga saham cenderung ditutupi mendekati ujung atas *price range*. sementara pada saat pasar turun. cenderung

mendekati ujung bawah (Mutmainah & Sulasmiyati, 2017). pada persamaan 2.29.

$$\%K = \frac{(\text{Today Close} - \text{Lowest low in K period})}{(highest\ High \text{ in K period} - \text{Lowest Lowin K period})} \text{ x } 100 \qquad (2.29)$$

Keterangan:

%K	=	Posisi relatif
Today close	=	Harga penutupan pada hari pengamatan
Lowest low	=	Harga *low* terendah pada hari pengamatan
Highest high	=	Harga *high* tertinggi pada periode hari pengamatan.
(n)	=	Periode hari pengamatan.

Gambar II.5 *Stochastic Oscillator.*

II.2.2.4 *Accumulation/Distribution Oscilator* (ADO/CLV)

Indikator ini merupakan indikator volume yang memiliki gagasan dasar bahwa pergerakan volume akan selalu mendahului pergerakan harga sehingga

pengguna dapat memprediksi pergerakan volume. Berikut ini adalah *Formula Accumulation Oscikator* (ADO) (Akbar *et. al.*, 2020). pada persamaan 2.30.

$$CLV = ((Close - Low) - (High - Close)) / (High - Low) \quad (2.30)$$

Keterangan :

Close = Harga Penutupan

Low = Harga Terendah

High = Harga Tertinggi

Gambar II.6 *Accumulation/Distribution Oscilator*.

II.2.2.5 *Price Rate of Change (PROC)*

Price Rate of Change (PROC) adalah indikator yang manampilkan perbedaan antara harga saat ini dan harga pada beberapa periode yang lalu dengan dinyatakan kedalam persentase. Perbedaan indikator ini dengan indikator momentum yang juga membandingkan harga saat ini dengan harga saat lalu adalah keluaran. Informasi hasil dari PROC ditampilkan dalam bentuk persentase sedangkan indikator momentum berupa rasio (Akbar *et. al.*, 2020). Berikut ini adalah *formula PROC*). Didefinisikan pada persamaan 2.31.

$$\text{PROC} = \left(\frac{((Today's\ Close - Close\ x\ Periods\ ago}{Close\ x\ Periods\ ago}\right) \quad (2.31)$$

Keterangan:

Today's Close = Harga penutupan sekarang

Close x Periods ago = Harga penetupan waktu yang lalu sesuai periode yang ditentukan.

Gambar II.7 *Rate of Change*

II.2.2.6 *Relative Strength Index (RSI)*

RSI digunakan untuk menghitung perbandingan antara daya tarik kenaikan dan penurunan harga. nilainya berkisar 0-100. Dengan RSI Anda dapat mengetahui apakah suatu harga sudah overbought atau oversold. Pada prinsipnya. penggunaan RSI sangat mudah. Jika RSI bernilai sangat tinggi (di atas 70) artinya pasar sudah overbought (jenuh beli) sehingga ada potensi turun. saatnya untuk jual. Sebaliknya jika RSI bernilai sangat rendah (di bawah 30) artinya pasar sudah oversold (jenuh jual) sehingga ada potensi naik. saatnya untuk beli (Shynkevich *et. al,*, 2017). Pada persamaan 2.32

$$\text{ROC} = \left(\frac{X}{Y}\right) * 100 \qquad (2.32)$$

Keterangan :

X = *Closing price* sekarang

Y = *Closing price* waktu yang lalu sesuai periode yang ditentukan.

II.3 Normalisasi data

Normalisasi data dilakukan dengan menggunakan *formula adaptive combiner linear* merupakan metode normalisasi dengan melakukan transformasi linier terhadap data asli sehingga menghasilkan keseimbangan nilai perbandingan antar data saat sebelum dan sesudah proses. Proses ini berfungsi memperkecil nilai dengan rentang -1 sampai +1 yang selanjutnya akan digunakan kedalam pengolahan melalui indikator saham yang dipilih. (Akbar *et. al.*, 2020). pada persamaan 2.33

$$y = \frac{2 * x - (\max + \min)}{(\max + \min)} \qquad (2.33)$$

keterangan :

y = Nilai normalisasi.

x = Harga saham sebenarnya.

Max = Harga saham terbesar dalam periode pengamatan

Min = Harga saham terkecil dalam periode pengamatan.

II.4 *Confusion Matrix*

Perhitungan akurasi di perlukan untuk mengetahui baik atau tidak nya performa suatu model kalsifikasi dan keakuratan prediksi. Nilai *precision* atau dikenal juga dengan nama *confidence* merupakan proporsi jumlah kasus yang

diprediksi positif yang juga positif benar pada data yang sebenarnya. Sedangkan nilai dari *recall* atau *sensitivity* merupakan proporsi jumlah kasus positif yang sebenarnya yang diprediksi positif secara benar (Visa *et. al*., 2011). pada persamaan 2.34

Accuracy (ACC) $\frac{TP}{P} - \frac{TN}{N} = \frac{TP+TN}{TP+TN+FP+FN}$ (2.34)

Keterangan :

P = Kondisi positif

N = Kondisi negatif

TP = *True positive*

TN = *True negatif* (TN)

FP = *False positive* (FP)

FN = *False negative* (FN)

II.5 Penelitian Terdahulu

Pada Penelitaian sebelumnya telah melakukan prediksi harga saham oleh (Sinaga, *et. al.,* 2019), menggunakan teknik SV-kNNC *(Support Vector - k Nearest Neighbor Clustering)* dan SOM (*Self Organizing Map*) untuk prediksi harga saham naik dan turun . Salah satu kelemahan dasar dari KNN adalah penempatan data baru ke dalam kelas dengan jumlah tetangga terbanyak (*Vote Majority System*) yang mana hal ini dapat menjadi masalah ketika jarak dari tiap *nearest neighbor* memiliki perbedaan yang besar terhadap *distance* dari data *testing.*Pada penelitaian ini mengkombinasikan SOM SVR dan LMKNN untuk prediksi harga saham , yang dimana *Local Mean-Based K-Nearest Neighbor*

(LMKNN) menjadi solusi untuk metode KNN . LMKNN dapat menempatkan suatu *instance* atau pola baru ke *label class* berdasarkan pada jarak terdekat dengan menggunakan nilai rata-rata dari *local centroid*. LMKNN telah memberikan hasil yang baik di dalam melakukan prediksi dan tingkat akurasi yang sangat baik (Fadilah et al., 2020).

Tabel II.2 Penelitian Terdahulu

No	Penelitian/ Tahun Terbit	Judul Penelitian	Hasil Penelitian
1.	Syaliman et al.,2018	Peningkatan Akurasi Pada Metode Klasifikasi *K-Nearest Neighbor* Menggunakan *Local Mean Based K-NearesrNeighbor* dan *DistanceNearesr Neighbor*.	• Metode yang diusulkan mampu meningkatkan nilai akurasi, di mana pada peningkatan nilai rata-rata akurasi dari seluruh data uji adalah sebesar 2,492%. Nilai rata-rata tertinggi akurasi diperoleh pada data Gejala Nyeri Punggung Bawah sebesar 3,91%, dan nilai rata-rata terendah ditemukan pada data iris sebesar 1,66%. • Dalam kasus memprediksi

			jurusan siswa di sekolah menengah negeri di kota Tualang, Indonesia, peneliti mengusulkan penggunaan metode yang diusulkan, karena metode ini terbukti lebih baik daripada kNN asli dengan rata-rata.
2.	Sinaga, *et all.*, 2019	Pengembangan Aplikasi Prediksi Trend Harga Saham Menggunakan Metode SV-KNNC dan SOM	Setelah melakukan pengujian pada aplikasi prediksi harga saham dengan menggunakan metode SV-KNNC dan SOM, maka dapat ditarik kesimpulan sebagai berikut: • Dengan mengubah tahapan Clustering pada algoritama SV-KNNC. Algoritma Self Organizing Map (SOM) lebih baik dibandingkan K-Means. • Pada algoritma *K-Fold Cross Validation* ketika user menggunakan data

			pada perusahaan TLKM akurasi yang diperoleh adalah = 71.55%. PGAS = 94.46%. ASII = 67.36%. BBCA = 80.33% dan BBRI = 97.21%. kNN asli dengan rata-rata.
3.	Fadla, et all, 2023	Analisis Prediksi HargaSaham PT.Telekomunikasi Indonesia Menggunakan Metode *Support Vector Machine* (SVM)	Setelah dilakukan prediksi harga PT.Indonesia dengan menggunkan metode *Support Vektor Machine* (SVM) dengan data harian harga saham dari *finance.yahoo.com* tanggal 29 April 2015 hingga 28 April 2020 sebanyak 1265 dataset. Diproses untuk persiapan data awal 20% testing dari dataset keseluruhan dan 80% training, kemudian dilanjutkan hasil uji pada testing menggunakan metode SVM akurasi yang diperoleh nilai sebesar 0.9641 dengan nilai RMSE 0.0932

			sedangkan pada metode KNN akurasi yang diperoleh sebesar 0.945.Dan Setelah dilakukan perbandingan dengan menguji masing-masing metode maka didapatkan disimpulkan : Bahwa metode *Suport Vecktor Machiine* (SVM) memiliki tingkat akurasi lebih tinggi dibandingkan metode KNN. Dari hasil pengujian akurasi algoritma SVM lebih tinggi daripada KNN, tetapi tidak memungkinkan jika terdapat metode lain yang akurasi dan keakuratan dalam hal prediksi/ peramalan lebih tinggi dengan dibandingkan Metode SVM.
4.	Koukaras,et all., 2023	*Stock Market Prediction Using Microblogging Sentiment Analysis and*	Dalam penelitian ini, kami mengkaji subjek SMP dengan menggunakan pendekatan ML dengan SA. Kami melakukan eksperimen dengan data-data

		Machine Learning	*Twitter* dan *StockTwits*, serta data keuangan. Dua alat SA, yaitu *TextBlob* dan VADER, digunakan untuk menganalisis sentimen data *microblogging*. Kebaruan dari pendekatan ini dapat diatribusikan pada integrasi berbagai metode SA dan ML. Kami juga mencoba menekankan pentingnya data *microblogging* media sosial dan sentimen publik sebagai fitur tambahan untuk SMP. Setelah melakukan pemrosesan data, kami mengevaluasi berbagai kinerja model kami dalam empat situasi *(TextBlob* dengan data *Twitter* atau *StockTwits*, dan VADER dengan data *Twitter* atau StockTwits) menggunakan tujuh algoritma ML, termasuk KNN, SVM, LR, NB, DT, RF,

			dan MLP. Model dievaluasi dengan dua metrik: *F-score* dan AUC. Kemampuan dalam prediksi aktual dievaluasi dari 10 Oktober 2020 hingga 31 Oktober 2020, menunjukkan kemungkinan di konsekuensi dunia nyata dalam meramalkan pergerakan nilai harga saham Microsoft. Nilai dataset yang digunakan yaitu : *'StockTwits TextBlob'*, SVM tampil terbaik dengan *F-score* sebesar 68,7% dan AUC sebesar 53,3%. Prediksi algoritma ini benar selama sepuluh hari, di mana nilai data saham mengalami kenaikan.Sama-sama dengan Metode SVM.
5.	Zhou, et all.,2022	*Application of kNN and SVM to predict the prognosis of advanced*	Model prediksi untuk prognosis pasien yang schistosomiasis memiliki tingkat lanjut yang sampelnya kecil belum banyak

		schistosomiasis	diteliti. Kami bertujuan untuk membuat model yang prediksi prognosis pasien yang memiliki schistosomiasis tingkat lanjut dengan nilai sampel kecil menggunakan dua algoritma pembelajaran mesin, yaitu k-*nearest neighbor* (kNN) dan *support vector machine* (SVM), dengan menggunakan data yang biasanya yang tersedia dalam program bantuan medis dari pemerintah. Mode prediksi ini dibuat berdasarkan data dari 229 pasien di Xiantao dan divalidasi eksternal dengan menggunakan data dari 77 pasien di Jiayu, dua kota tingkat kabupaten di Provinsi Hubei, Tiongkok. Prediktor-prediktor kandidat dipilih berdasarkan pandangan para ahli dan laporan literatur, termasuk fitur klinis,

			karakteristik sosiodemografi, dan hasil pemeriksaan medis. Area diantara bawah kurva karakteristik operasi penerima (AUC), sensitivitas, dan untuk spesifisitas digunakan untuk mengevaluasi kinerja model prediksi. Nilai AUC adalah 0,879 untuk model kNN dan 0,890 untuk model SVM pada data pelatihan, serta 0,852 untuk model kNN dan 0,785 untuk model SVM pada data validasi eksternal. Model kNN dan SVM dapat digunakan untuk meningkatkan dengan layanan kesehatan yang disediakan oleh perencana kesehatan, dokter, dan pembuat kebijakan.
8	Chang ,et all., 2022	*Long-Term Flooding Maps Forecasting System Using Series Machine Learning*	Dalam penelitian ini, kami mengusulkan QPF-RIF, yang yang dapat meramalkan peta kedalaman banjir di masa depan

		and Numerical WeatherPrediction System	(dengan lead time 72 jam) tanpa harus mendirikan stasiun pemantauan banjir perkotaan dan dibangun berdasarkan basis data banjir yang disimulasikan. Hasil penelitian ini dapat dirangkum menjadi poin-poin berikut. Pertama, dan hasil peramalan SVM-MSF 1 jam ke depan hampir sempurna sesuai dengan volume banjir total yang disimulasikan oleh SOBEK, dan kesalahan rata-rata kedalaman banjir dapat dikendalikan dengan kurang dari 1 cm. Ini dapat memberikan peramalan jauh ke depan, lebih dari 46 jam ke depan, di bawah kondisi bahwa data hujan ensemble peramalan dapat dipercaya. Kedua, kami dapat dengan jelas dapat melihat bahwa dengan algoritma pengelompokan /

			Clsutering yang digunakan dalam penelitian ini dapat secara efektif membedakan banjir parah di berbagai jenis area banjir atau area dangkal berukuran besar. Selain itu, itu dapat memetakan volume banjir total ke setiap grid, dan kesalahan rata-rata dari semua grid berada dalam batas 7 cm. Ketiga, peta kedalaman banjir yang diperkirakan oleh QPF-RIF, yang disatukan oleh SVM-MSF dan SOM, dapat sangat mirip dengan peta yang disimulasikan oleh SOBEK yang dilihat sebagai nilai target dalam penelitian ini, terutama untuk area dengan banjir parah dan memiliki kinerja yang lebih akurat. Mungkin ada sedikit overestimasi untuk grid akumulasi air yang sedikit, dan

			ini tidak memengaruhi akurasi ketersediaan dengan cara keseluruhan. Akhirnya, hasil verifikasi untuk tiga badai taifun dan satu kejadian hujan lebat tunggal menunjukkan bahwa QPF-RIF memiliki kemampuan dalam peramalan kedalaman dan luas banjir yang tinggi (83% dari area banjir). Kecuali beberapa area di mana banjir tidak disimulasikan dalam basis data banjir dan tidak dapat diramalkan secara akurat, QPF-RIF dapat secara efektif dan akurat meramalkan area rawan banjir dan area banjir yang lebih dalam beberapa jam ke depan. kesimpulan, QPF-RIF yang diusulkan dalam penelitian ini dapat dengan nilai akurat meramalkan distribusi dan

			kedalaman banjir pada jangka panjang dan yang memberikan informasi real-time dan masa depan yang lebih dapat diandalkan. Di masa depan, mengurangi ketidakpastian yang disebabkan oleh data hujan ensemble peramalan dan menggabungkan dengan data pemantauan real-time, seperti menerapkan berbagai metode pembelajaran mesin, metode statistik, dan informasi stasiun pemompaan bergerak untuk mencapai tujuan koreksi real-time peta genangan, mungkin menjadi tantangan yang layak.
9.	*K U Syaliman et all., 2018*	*Improving the accuracy of k-nearest neighbor using local mean based and distance weight*	Berdasarkan temuan dan diskusi dalam pada bagian sebelumnya, dapat disimpulkan bahwa: • Metode yang diusulkan mampu meningkatkan nilai

			akurasi, dimana dalam peningkatan nilai rata-rata akurasi pada semua data uji adalah 2.492%. Nilai rata-rata tertinggi dari akurasi diperoleh pada data Gejala Nyeri Punggung Bawah sebesar 3.91%, dan nilai rata-rata terendah (*low*) ditemukan pada data iris sebesar 1.66%. • Dalam kasus memprediksi jurusan siswa di sekolah menengah negeri di Kota Tualang, Indonesia,dalam peneliti ini mengusulkan penggunaan metode yang diusulkan, karena metode yang diusulkan terbukti lebih baik daripada kNN asli dengan perbedaan akurasi rata-rata sebesar 5.16%, di mana nilai akurasi

			tertinggi adalah 90.91% ketika k = 10. Sama Menggunakan Metode LMKNN.
10.	*Gosh et all. ,2019*	*A Study on Support Vector Machine based Linear and Non-Linear Pattern Classification*	Dalam makalah ini, konsep SVM beserta dalam berbagai modelnya telah disajikan. Fenomena yang mendorong munculnya konsep SVM pada tahun 90-an juga disebutkan. SVM terutama digunakan untuk analisis klasifikasi dan regresi, lebih baik untuk batas keputusan linear maupun non-linear. Dengan Metode SVM mengklasifikasikan batas-batas keputusan non-linear dengan bantuan kernel seperti RBF, sigmoid, polinomial, dan sebagainya. Terkecuali dalam beberapa keterbatasan, SVM adalah alat pembelajaran mesin yang sangat berguna yang diterapkan dalam kehidupan

			sehari-hari kita, nilai seperti penyaringan spam, deteksi wajah, dan sebagainya. SVM memberikan hasil yang paling dioptimalkan dan akurat di dalamSebuah klasifikasi dinilai berdasarkan waktu pelatihan, waktu pengujian, dan akurasi klasifikasi. Untuk menentukan hubungan antara kernel dan distribusi data yang akan memilih fungsi kernel yang tepat untuk dataset tertentu digunakan memaksimalkan pemisahan kelas antara poin dari data kelas merupakan tantangan yang sangat besar. Juga dapat dilihat bahwa saat ini dalam aplikasi seluler mengumpulkan sejumlah besar data dengan pengguna untuk meningkatkan pengalaman pengguna dalam aplikasi. Hal

			ini dapat dilakukan dengan menggunakan kernel yang dapat diubah ukurannya dengan baik, sehingga dalam pengumpulan data data dalam jumlah besar yang tidak akan diperlukan pada domainnya dengan dibandingkan.SVM (Support *Vector Machine*) memiliki nilai aplikasi dalam memprediksi bencana alam seperti gempa bumi dan tsunami. Sinyal yang telah diproses dari gelombang seismik dimasukkan ke dalam pengklasifikasi SVM sebagai input nilai yang kemudian memprediksi lokasi gempa bumi yang akan datang. Diketahui bahwa dalam hal lokasi gempa bumi di masa depan, Dengan SVM dapat memberikan akurasi sebesar

			77%.
11.	*Abuzneid, et all., 2019*	*Enhanced Human Face Recognition Using LBPH Descriptor, Multi-KNN, and BackPropagation Neural Network*	Kami mengusulkan kerangka kerja yang ditingkatkan untuk pengenalan wajah manusia menggunakan yaitu deskriptor LBPH, multi-KNN, dan juga jaringan saraf pada BPNN. Deskriptor LBPH yang baru dan juga multi-KNN telah membantu memberikan dataset pelatihan dengan pengualangan pola perbedaan berdasarkan korelasi antara gambar dan pelatihan asli. Dataset T yang baru diperoleh membantu BPNN untuk konvergensi lebih cepat dan dengan akurasi yang lebih tinggi. Ini dicapai dengan menggabungkan metode jarak, karena setiap metode jarak memiliki berbagai kelebihan dibandingkan dengan metode lainnya, yang memperkuat

			seluruh sistem. Kami mencapai akurasi yang lebih tinggi dan mengurangi waktu komputasi. Selain itu, kami melebihi kerangka kerja terkini yang ada. Kami belum dapat menerapkan jaringan saraf modern untuk membuktikan bahwa kami dapat mencapai akurasi yang lebih tinggi bahkan dengan ekstraksi fitur tradisional dan pada metode pengurangan nilai domain menggunakan dataset pelatihan yang berkorelasi di antara gambar. Namun, dalam sebuah pekerjaan masa depan, kami berencana menggunakan nilai metode ekstraksi fitur seperti jaringan saraf tiruan dan konvolusional dan juga membandingkannya dengan hasil saat ini.Hasil dari metode

			yang kami usulkan dan perbandingan dengan metode terbaik yang ada. Kami mencapai hasil akurasi sebesar 95.71%, yang dimana dapat dibandingkan dengan metode terbaik yang ada, dan kami mencapai akurasi kedua tertinggi setelah metode. Namun, kesalahan standar kami lebih rendah daripada yang yang menghasilkan nilai akurasi keseluruhan yang lebih tinggi

II.6 Kontribusi Penelitian

Adapun kontribusi penelitian ini adalah menggabungkan Penerapan SVR,SOM untuk meningkatkan kinerja LMKNN dimana peran dari SVR dan SOM sebagai berikut :

1. Penerapan SVR (*Support Vector Regression*) untuk memperoleh *Vector* terbaik dari Data Saham (Dataset), kemudian dilakukan pelabelan pada data yaitu (Harga Naik, Harga Tetap dan Harga Turun)

2. *Clustering* Data menggunakan *Self-Organizing Maps* (SOM): Pada tahapan ini dilakukan pengelompokan dan pembobotan data. Proses dalam Pengemlompokan SOM melibatkan dua proses utama yaitu inisialiasi dan pelatihan . Dimana pada inissialiasi untuk memperoleh bobot awal vektor menggunakan metode berbasis acak, melakukan pencocokan unit terbaik dengan jarak yang terdekat, melakukan pembaharuan bobot dari setiap iterasi menggunakan Gausin,kemudian dari jarak yang paling kecil akan dimasukkan kedalam cluster yang terpilih.