**ANALISIS PREDIKSI HARGA SAHAM MANCHESTER UNITED MENGGUNAKAN METODE SVM (SUPPORT VECTOR MACHINE)**

**SKRIPSI**

Diajukan untuk memenuhi salah satu syarat

memperoleh gelar Sarjana Komputer

Program Studi Informatika

****

Diajukan Oleh:

**Fransiskus Jremiegi Saputra**

**205314062**

PROGRAM STUDI INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS SANATA DHARMA

YOGYAKARTA

2024

**MANCHESTER UNITED STOCK PRICE PREDICTION ANALYSIS USING SVM (SUPPORT VECTOR MACHINE) METHOD**

**A THESIS**

Submitted in partial fulfillment of the requirements

For the degree of Sarjana Komputer

In Informatics Study Program

****

Submitted By:

**Fransiskus Jremiegi Saputra**

**205314062**

INFORMATICS STUDY PROGRAM

FACULTY OF SCIENCE AND TEHCNOLOGY

SANATA DHARMA UNIVERSITY

YOGYAKARTA

2024

# **KATA PENGANTAR**

Berangkat dari rasa penasaran dan keingitahuan yang mendalam, penulis pada akhirnya memutuskan untuk mencoba meneliti permasalah ini sebagai topik tugas akhir. Namun, tak jarang pula, rasa keraguan mulai muncul sebagai penghambat seiring berjalanya waktu. Keraguan tersebut acap kali membuat penulis cukup resah dan ingin berujung dengan kata ‘meyerah’. Namun, berkat diskusi yang tak kenal lelah sekaligus dukungan dari berbagai pihak, penulis merasa dilahirkan kembali dengan rasa kepercayaan diri yang semakin kuat untuk menyelesaikan tugas akhir ini.

Melalui proses yang telah dilalui, penulis menyadari betul bahwa tantangan dan keraguan merupakan proses yang tak terpisahkan dalam perjalanan akademik, yang justru semakin memperkuat tekad dan semagat untuk terus maju dan berproses. Hal ini juga terkadang menjadi batu loncatan untuk semakin berani menerobos tantangan apapun serta menjadi penyemangat untuk terus berupaya meyelesaikan penelitian tugas akhir ini dengan penuh dedikasi dan komitmen yang tak tergoyahkan. Pada akhirnya, penulis menyadari bahwa setiap keresahan dan tantangan yang hadir merupakan suatu kesempatan untuk terus belajar dan bertumbuh jauh lebih baik lagi.

Penulis sungguh menyadari bahwa tulisan ini tidak terlepas dari dukungan dan kontribusi yang luar biasa dari berbagai pihak. Melalui dukungan serta kontribusi tersebut, pada akhirnya penulis dapat menyelesaikan tugas akhir ini dengan semaksimal mungkin. Oleh karena itu, penulis ingin mengucapkan terima kasih yang luar biasa kepada :

1. Yesus Kristus yang selalu menjadi penguat serta sumber kekuatan dalam setiap proses.
2. Ayah dan Ibu yang selalu menjadi pengingat melalui pertanyaan ‘kapan lulus ?’ serta dukungan dengan sepenuh hati dan cinta dalam menyelesaikan tugas akhir ini.
3. Adik terkasih, Maria Riska yang selalu memberikan semangat dalam menyelesaikan tugas akhir ini.
4. Bapak Ir. Kartono Pinaryanto S.T., M.Cs. selaku dosen pembimbing tugas akhir yang telah bersedia untuk memberikan bimbingan, arahan, diskusi, saran dan kritik terhadap penyelesaian tugas akhir ini hingga selesai.
5. Catrien Latumaerissa yang selalu bersedia untuk berdiskusi apapun dan kapanpun terkhususnya dalam perkuliahan maupun penelitian ini.
6. Teman-teman di komunitas Kebenaran yang selalu saling menguatkan satu sama lain serta tak pernah lelah dalam memberikan saran serta kritikan yang membangun.
7. Semua pihak yang tidak dapat saya sebutkan satu per satu dalam membantu penyelesaian tugas akhir ini.

Penulis menyadari bahwa tulisan ini masih banyak kekurangan dan jauh dari kata sempurna. Oleh karena itu, penulis dengan terbuka untuk menerima setiap diskusi, saran dan kritikan yang membangun demi kesempurnaan tugas akhir ini. Maka dari itu, semoga tulisan ini dapat berguna sebagai bentuk kontribusi yang bermanfaat untuk wawasan dan perkembangan pengetahuan pembaca.

# **ABSTRAK**

Penting untuk memahami bahwa investasi saham memerlukan pengetahuan yang cukup, riset yang matang, dan perencanaan yang hati-hati. Oleh karena itu, analisis prediksi saham yang akurat sangat penting untuk memberikan gambaran terkait prospek saham di masa mendatang. Dalam penelitian ini, akan dilakukan prediksi pergerakan harga saham Manchester United pada bursa efek NYSE (New York Stock Exchange) dalam rentang Waktu 5 tahun terakhir yakni 31 Oktober 2018 hingga 30 Oktober 2023. Model yang digunakan ialah SVM (Support Vector Machine) dengan kernel RBF (kernel Radial Basis Function) dan kernel Polynomial serta nilai C yakni 1, 10, 100 dan 1000 dan nilai gamma 1, 0.1, 0.01, 0.001, dan 0.0001 serta diuji menggunakan cross-validation dengan nilai k-fold yakni 3, 5, 7 dan 9. Hasil akurasi parameter terbaik diperoleh pada data yang dilakukan *balance* yakni parameter kernel RBF, nilai C sama dengan 1000 dan nilai gamma sama dengan 0.1 pada 7-fold yang menghasilkan akurasi sebesar **99.82%**. Parameter tersebut akan digunakan untuk melakukan prediksi model dan menghasilkan akurasi sebesar **99.70%**.

**Kata kunci:** *saham, prediksi, manchester united, support vector machine*

# **ABSTRACK**

It is important to understand that investing in stocks requires considerable knowledge, thorough research and careful planning. Therefore, accurate stock prediction analysis is very important to provide an overview of the future prospects of the stock. In this research, a prediction of Manchester United's stock price movement on the NYSE (New York Stock Exchange) will be carried out in the last 5 years, namely 31 October 2018 to 30 October 2023. The model used is SVM (Support Vector Machine) with RBF kernel (Radial Basis Function kernel) and Polynomial kernel and C values of 1, 10, 100 and 1000 and gamma values of 1, 0.1, 0.01, 0.001, and 0.0001 and tested using cross-validation with k-fold values of 3, 5, 7 and 9. The best parameter accuracy results are obtained on data that is balanced, namely RBF kernel parameters, C value equal to 1000 and gamma value equal to 0.1 at 7-fold which results in an accuracy of **99.82%**. These parameters will be used to predict the model and produce an accuracy of **99.70%**.

**Keywords**: stock, prediction, manchester united, support vector machine

# **DAFTAR ISI**

[KATA PENGANTAR iii](#_Toc171967160)

[ABSTRAK v](#_Toc171967161)

[ABSTRACK vi](#_Toc171967162)

[DAFTAR ISI vii](#_Toc171967163)

[DAFTAR GAMBAR x](#_Toc171967164)

[DAFTAR TABEL xii](#_Toc171967165)

[BAB I PENDAHULUAN 1](#_Toc171967166)

[1.1 Latar Belakang 1](#_Toc171967167)

[1.2 Rumusan Masalah 2](#_Toc171967168)

[1.3 Batasan Masalah 3](#_Toc171967169)

[1.4 Tujuan Penelitian 3](#_Toc171967170)

[1.5 Manfaat Penelitian 3](#_Toc171967171)

[1.6 Sistematika Penulisan 4](#_Toc171967172)

[BAB II TINJAUAN PUSTAKAN 6](#_Toc171967173)

[2.1 Kajian Teori 6](#_Toc171967174)

[2.1.1 Saham 6](#_Toc171967175)

[2.1.2 Analisis Teknikal 6](#_Toc171967176)

[2.1.3 RSI (Relative Strength Index) 7](#_Toc171967177)

[2.1.4 Manchester United 9](#_Toc171967178)

[2.1.5 SVM (Support Vector Machine) 10](#_Toc171967179)

[2.1.6 SMOTE (Synthetic Minority Over-sampling Technique) 12](#_Toc171967180)

[2.1.7 Performance Metrics 13](#_Toc171967181)

[2.2 Kajian Hasil Riset 14](#_Toc171967182)

[BAB III METODE PENELITIAN 20](#_Toc171967183)

[3.1 Pengumpulan Data 21](#_Toc171967184)

[3.2 Preprocessing 22](#_Toc171967185)

[3.3 Ekstraksi Fitur 23](#_Toc171967186)

[3.4 Eksplorasi Data: Balance vs Imbalance 24](#_Toc171967187)

[3.5 Pembagian Data 24](#_Toc171967188)

[3.6 Modeling 25](#_Toc171967189)

[3.7 Evaluasi 26](#_Toc171967190)

[3.8 Skenario Pengujian 26](#_Toc171967191)

[BAB IV PEMBAHASAN 30](#_Toc171967192)

[4.1 Data 30](#_Toc171967193)

[4.2 Preprocessing Data 30](#_Toc171967194)

[4.2.1 Menghapus Duplikasi 30](#_Toc171967195)

[4.2.2 Menghapus Atribut Tidak Lengkap 31](#_Toc171967196)

[4.3 Ekstraksi Fitur 32](#_Toc171967197)

[4.3.1 Menemukan Nilai RSI (Relative Strength Index) 32](#_Toc171967198)

[4.3.2 Menentukan Target/Label 35](#_Toc171967199)

[4.3.3 Normalisasi Data 36](#_Toc171967200)

[4.4 Balence Data 38](#_Toc171967201)

[4.5 Pembagian Data 40](#_Toc171967202)

[4.6 Modeling 41](#_Toc171967203)

[4.6.1 Pemodelan SVM (Support Vector Machine) 41](#_Toc171967204)

[4.6.2 Mengatur Parameter Nilai Pada GridSearchCV 42](#_Toc171967205)

[4.6.3 Pencarian Parameter Terbaik 44](#_Toc171967206)

[4.6.4 Pengujian Menggunakan Paramter Terbaik 45](#_Toc171967207)

[4.7 Hasil Dan Evaluasi 46](#_Toc171967208)

[4.7.1 Hasil Pengujian Parameter Pada 3-Fold 46](#_Toc171967209)

[4.7.2 Hasil Pengujian Parameter Pada 5-Fold 50](#_Toc171967210)

[4.7.3 Hasil Pengujian Parameter Pada 7-Fold 53](#_Toc171967211)

[4.7.4 Hasil Pengujian Parameter Pada 9-Fold 57](#_Toc171967212)

[4.7.5 Akurasi Parameter Terbaik Dari K-Fold Yang Diuji 60](#_Toc171967213)

[4.7.6 Hasil Pengujian Model Menggunakan Parameter Terbaik 61](#_Toc171967214)

[BAB V KESIMPULAN DAN SARAN 63](#_Toc171967215)

[5.1 Kesimpulan 63](#_Toc171967216)

[5.2 Saran 64](#_Toc171967217)

[DAFTAR PUSTAKA 66](#_Toc171967218)

# **DAFTAR GAMBAR**

[Gambar 3. 1 Gambaran umum penelitian 20](#_Toc168643199)

[Gambar 3. 2 Contoh data harga saham Manchester United 22](#_Toc168643200)

[Gambar 3. 3 Alur proses preprocessing 23](#_Toc168643201)

[Gambar 3. 4 Alur proses ekstraksi fitur 23](#_Toc168643202)

[Gambar 3. 5 Alur proses balance data dan imbalance data 24](#_Toc168643203)

[Gambar 3. 6 Alur proses metode k-flod cross validation 25](#_Toc168643204)

[Gambar 4. 1 Code untuk load data 30](#_Toc171967071)

[Gambar 4. 2 Code untuk cek duplikasi nilai 31](#_Toc171967072)

[Gambar 4. 3 Code untuk mengetahui nilai yang kosong dari tiap-tiap atribut 31](#_Toc171967073)

[Gambar 4. 4 Code untuk cek nilai atribut yang tidak lengkap 32](#_Toc171967074)

[Gambar 4. 5 Code untuk jumlah periode yang digunakan 32](#_Toc171967075)

[Gambar 4. 6 code untuk menghitung nilai selisih antara atribut Close hari ini dengan nilai atribut Close hari sebelumnya 33](#_Toc171967076)

[Gambar 4. 7 Code untuk menentukan atau mengelompokkan nilai penutupan naik/up dan nilai penutupan turun/down 33](#_Toc171967077)

[Gambar 4. 8 Code untuk menghitung nilai rata-rata penutupan naik/up dan rata-rata penutupan turun/down 34](#_Toc171967078)

[Gambar 4. 9 Code untuk menghitung nilai RSI 34](#_Toc171967079)

[Gambar 4. 10 Output dari hasil perhitungan RSI 35](#_Toc171967080)

[Gambar 4. 11 Code untuk menentukan target/label 36](#_Toc171967081)

[Gambar 4. 12 Output setelah menentukan nilai pada label/target 36](#_Toc171967082)

[Gambar 4. 13 Code untuk normalisasi nilai pada data 37](#_Toc171967083)

[Gambar 4. 14 Output sebelum nilai pada data di normalisasi 37](#_Toc171967084)

[Gambar 4. 15 Output setelah nilai pada data di normalisasi 37](#_Toc171967085)

[Gambar 4. 16 Code untuk belance data 38](#_Toc171967086)

[Gambar 4. 17 code serta output pada program sebelum dilakukan belence data 39](#_Toc171967087)

[Gambar 4. 18 code serta output pada program setelah dilakukan belence data 40](#_Toc171967088)

[Gambar 4. 19 Pembagian data traning dan testing menggunakan train\_test\_split pada dataset imbalance dan balance 41](#_Toc171967089)

[Gambar 4. 20 Data traning imbalance maupun balance akan dilakukan pembagian data secara silang berdasarkan nilai k-fold menggunakan GridSearchCV 41](#_Toc171967090)

[Gambar 4. 21 Code untuk inisialisasi model SVM (Support Vector Machine) 42](#_Toc171967091)

[Gambar 4. 22 Code untuk parameter model SVM (Support Vector Machine) 42](#_Toc171967092)

[Gambar 4. 23 Code yang mendefinisikan nilai dari variabel scorcing 43](#_Toc171967093)

[Gambar 4. 24 Code yang mendefinisikan nilai dari function custom\_refit 44](#_Toc171967094)

[Gambar 4. 25 Code yang menerapkan GridSearchCV 44](#_Toc171967095)

[Gambar 4. 26 Code yang digunakan untuk melakukan pelatihan model 44](#_Toc171967096)

[Gambar 4. 27 Code yang digunakan untuk menemukan hasil parameter terbaik serta akurasinya 45](#_Toc171967097)

[Gambar 4. 28 Code untuk melakukan prediksi atau pengujian model 46](#_Toc171967098)

[Gambar 4. 29 Perbandingan hasil parameter terbaik untuk tiap-tiap fold yang diuji pada data imbalance dan data balance 61](#_Toc171967099)

[Gambar 4. 30 Performance metrics dari prediksi model 62](#_Toc171967100)

# **DAFTAR TABEL**

[Tabel 2. 1 Rumus fungsi karnel 11](#_Toc168996893)

[Tabel 2. 2 Hasil kajian teori 14](#_Toc168996894)

[Tabel 3. 1 Skenario pengujian 26](#_Toc168996895)

[Tabel 3. 1 Skenario pengujian 26](#_Toc171964889)

[Table 4. 1 Hasil pengujian kernel, nilai C dan nilai menggunakan 3-fold pada data imbalance 46](#_Toc168996914)

[Table 4. 2 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 3-fold pada data balance 48](#_Toc168996915)

[Table 4. 3 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 5-fold pada data imbalance 50](#_Toc168996916)

[Table 4. 4 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 5-fold pada data balance 51](#_Toc168996917)

[Table 4. 5 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 7-fold pada data imbalance 54](#_Toc168996918)

[Table 4. 6 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 7-fold pada data balance 55](#_Toc168996919)

[Table 4. 7 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 9-fold pada data imbalance 57](#_Toc168996920)

[Table 4. 8 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 3-fold pada data balance 59](#_Toc168996921)

# **PENDAHULUAN**

## **Latar Belakang**

Dewasa ini, trend terhadap saham kian populer dikalangan masyarkat mulai dari iming-iming manis yang menjanjikan untuk cuan sampai pada penipuan yang merugikan bahkan tak jarang adu gengsian agar tidak jadi bahan gosipan. Adanya jaminan kemudahan dalam mengakses pasar saham juga menjadi pemikat agar masyarkat semakin tertarik untuk terjun ke dunia investasi. Lantas, fenomena ini menujukan bahwa pasar saham menjadi daya tarik utama dikalangan masyarakat umum sehingga mencerminkan minat yang meningkat terhadap investasi dan keuangan.

Penting untuk memahami bahwa investasi saham memerlukan pengetahuan yang cukup, riset yang matang, dan perencanaan yang hati-hati. Hal ini dikarenakan saham bergerak secara non linier dan perubahan harga bisa terjadi begitu cepat yang dipengaruhi oleh banyak faktor seperti kondisi keuangan perusahaan, laju inflasi, tingkat suku bunga, permintaan dan penawaran, dan masih banyak lagi (Fadilah, Agfiannisa, & Azhar, 2020) (Wulandari & Anubhakti, 2022) (Syahputra, Ramadhan, & Burha, 2022) (Maulana & Kumalasari, 2019) (Lestari & Pangaribuan, 2020). Oleh karena itu, penting untuk mengetahui dan mempelajari perilaku pergerakan harga saham dengan melakukan analisis prediksi harga saham sehingga berguna untuk mengetahui prospek dan alur investasi di masa mendatang serta membantu dalam mempertimbangkan pengambilan keputusan transaksi saham (Fadilah, Agfiannisa, & Azhar, 2020) (Wulandari & Anubhakti, 2022) (Syahputra, Ramadhan, & Burha, 2022) (Maulana & Kumalasari, 2019) (Lestari & Jasuni, 2023) . Dengan demikian, Adanya model prediksi pergerakan harga saham yang akurat dapat menjadi tools yang bisa membantu para investor dalam melakukan pengambilan keputusan terkait dengan saham dikarenakan pergerakan harga saham yang cenderung non linier ini akan menyulitkan investor dalam melakukan prediksi. Terdapat beberapa penelitian yang membahas terkait dengan prediksi saham dengan menggunakan beberapa model algoritma.

Penelitian yang dilakukan Wulandari & Anubhakti (2022) dalam memprediksi harga saham PT. Garuda Indonesia, Tbk menggunakan pengujian model SVM yakni menghasilkan performance prediction trend accuracy sebesar 0.545 dari 520 dataset yang ada (Wulandari & Anubhakti, 2022). Masih menggunakan metode yang sama yakni SVM namun dikombinasikan dengan melakukan perbandingan menggunakan KNN untuk memprediksi harga saham PT. Telekomunikasi Indonesia menghasilkan nilai akurasi tertinggi yakni 0.9653 dan nilai RMSE sebesar 0.0091 untuk algoritma SVM sedangkan algoritma KNN nilai akurasi sebesar 0.9456 dan RMSE sebesar 0.1162 (Fadilah, Agfiannisa, & Azhar, 2020). Tidak hanya itu, terdapat juga penelitian terhadap prediksi harga saham Bank BRI menggunakan algoritma Linier Regression dan memperoleh akurasi R-2 sebesar 0.9125 dengan nilai error pada prediksi yang diukur menggunakan MAPE dan menghasilkan persentase sebesar 13,751% untuk data pelatihan, 13,773% untuk data pengujian, dan 13,755% untuk data keseluruhan atau persentase error berada pada rentang 10% - 20% (Syahputra, Ramadhan, & Burha, 2022).

Berdasarkan latar belakang yang telah diuraikan tersebut maka dapat dilakukan penelitian dalam memprediksi harga saham dengan pendekatan yang serupa sehingga dapat membantu para investor dalam melihat prospek suatu saham tertentu. Tidak hanya itu, metode algoritma yang digunakan juga diharapkan mampu menjawab permasalahan yang telah diuraikan serta dapat mengetahui tingkat keakuratan dari suatu model algoritma dalam memprediksi harga saham.

## **Rumusan Masalah**

1. Bagaimanakah metode Support Vector Machine (SVM) dapat digunakan untuk memprediksi harga saham secara akurat ?
2. Bagaimana hasil dari tingkat keakuratan dari suatu model algoritma yang digunakan dalam memprediksi harga saham dapat menjawab permasalah seperti yang telah diuraikan pada bagian latar belakang ?

## **Batasan Masalah**

1. Data saham yang diambil ialah saham Manchester United dengan kode MANU pada bursa efek New York Stock Exchange (NYSC).
2. Data yang diambil yakni data pergerakan harian harga saham selama 5 tahun terakhir dengan rentang waktu yakni 31 Oktober 2018 sampai 30 Oktober 2023.
3. Algoritma yang digunakan hanya terbatas pada Support Vector Machine (SVM).

## **Tujuan Penelitian**

Pentingnya pengetahuan yang cukup, riset yang matang dan perencanaan yang hati-hati dalam investasi membutuhkan analisis prediksi saham yang akurat sehingga memberikan gambaran terkait dengan prospek saham di masa yang mendatang. Dengan demikian, hadirnya model dalam memprediksi pergerakan harga saham dapat menjadi tools untuk membantu dalam menganalisis serta mengambil keputusan sebagai bahan pertimbangan. Di sisi lain, metode yang digunakan juga menjadi tolak ukur untuk melihat sejauh mana keakuratan yang dihasilkan serta seberapa efektif model tersebut melahirkan prediksi yang akurat ditengah kondisi pasar saham yang bergerak secara non-linier.

## **Manfaat Penelitian**

1. Membantu para investor dalam memprediksi harga saham dan mengetahui prospek saham di masa yang akan datang sehingga dapat meminimalkan resiko dan memaksimalkan keuntungan.
2. Membantu investor dalam mempertimbangkan pengambilan keputusan sehingga menjadi panduan dalam menyusun starategi investasi baik jangka pendek maupun jangka Panjang.
3. Metode yang digunakan untuk memprediksi menjadi tolak ukur dalam mengetahui tingkat keakuratan dan efektifitas sehingga membantu sejauh mana model tersebut dapat diandalkan dan menghasilkan prediksi yang sesuai ditengah pasar saham yang bergerak secara non-linier.

## **Sistematika Penulisan**

Pada penelitian ini, sistematika penulisan tugas akhir dibagi dalam berberap bab yakni:

**BAB I PENDAHULUAN**

Bab ini membahasa terkait dengan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Bab ini membahasa terkait dengan tinjauan pustaka serta teori-teori yang mendukung dan relevan dengan penelitian tugas akhir ini seperti teori terkait dengan Support Vector Machine (SVM), Saham, Analisis Teknikal dan Manchester United. Tidak hanya itu, pada bab ini juga dilengkapi dengan hasil kajian riset dari penelitian sebelumnya.

**BAB III METODE PENELITIAN**

Bab ini membahasa terkait dengan prosedur yang akan dilakukan oleh penulis dalam menyelesaikan tugas akhir ini. Prosedur tersebut berupa suatu gambaran atau konsep alur kerja secara sistematis berupa *inputan* dan *ouput* dari tiap-tiap proses yang dilakukan. Andapun proses yang dilalui penulis seperti pengambilan data, preprocessing, ekstraksi fitur, pembagaian data, modeling dan evaluasi serta scenario pengujian.

**BAB IV PEMBAHASAN**

Bab ini membahasa terkait dengan hasil dari penelitian yang telah dilakukan berupa analisis dan evalusi terhadap kinerja model yang telah dibangun.

**BAB V KESIMPULAN**

Bab ini membahasa terkait dengan kesimpulan akhir serta menjawab pertanyaan pada rumusan masalah yang telah dijabarkan pada bab pertama. Tidak hanya itu, penulis juga memberikan saran atau kritik yang membangun untuk pengembangan penelitian yang dilakukan pada tugas akhir ini.

# **TINJAUAN PUSTAKAN**

## **Kajian Teori**

### **Saham**

Saham merupakan bukti tanda kepemilikan dan penyertaan modal perseroan terbatas yang memberikan hak atas dividen dan lain-lain berdasarkan besar kecilnya modal (Rusmalawati, Furqon, & Indriati, 2018). Dengan membeli saham perusahaan menandakan bahwa adanya penanaman modal yang nantinya akan digunakan oleh pihak manejemen perusahaan untuk membiayai operasional perusahaan. Lantas, keuntungan yang diperoleh dari penanaman modal tersebut yakni *dividen* yang merujuk pada laba yang akan dibagian kepada pemegang saham sesuai dengan porsi kepemilikannya dan *capital gain* yang merujuk pada keuntungan yang didapatkan dengan memperdagangkan saham sehingga keuntungan yang diperoleh sangat bergantung pada harga saham di pasar modal sendiri (Rusmalawati, Furqon, & Indriati, 2018). Namun perlu digarisbawahi bahwa *dividen* tidak wajib dibayarakan/dibagiakan oleh perusahaan kepada para pemegang saham melainkan bisa saja perusahaan menggunakan laba yang dimiliki tersebut untuk keperluan ekspansi bisnis perusahaan maupun reinvestasi sehingga menghasilkan keuntungan lebih lanjut dengan meningkatnya nilai harga saham (Dr. Mamduh M. Hanafi, 2016).

### **Analisis Teknikal**

Analisis teknikal merupakan metode analisis yang digunakan untuk memprediksi harga saham di masa yang akan datang berdasarkan data historikal harga dan jumlah/volume transaksi yang direpresentasikan dalam bentuk grafik, garis gambar, chart atau candle (Hidayahtullah, 2022). Analisis teknikal juga bisa digunakan untuk melihat perilaku pasar atau arah pergerakan harga berdasarkan silkus sebelumnya sehingga bisa membantu dalam melihat trend di masa yang akan datang (Hidayahtullah, 2022). Hadirnya analisis teknikal ini bisa dijadikan sebagai parameter acuan dalam menentukan sebuah keputusan atau kelayakan ditengah perilaku pergerakan pasar saham yang cenderung non-linier.

Analisis teknikal diperkirakan sudah ada sejak 1000 tahun yang lalu, yang ditandai dengan kehadiran para ahli-ahli dalam melakukan analisis pasar keuangan. Hal ini dapat dilihat dengan adanya bukti tertua yang ditemukan berupa catatan-catatan yang digunakan dalam menganalisis pasar keuangan Belkamu oleh Joseph Penso de la Vegas pada abab ke-17. Sedangkan di Asia, analisis teknikal dikembangkan oleh Homma Munehisa dari Jepang pada awal abad ke-18 berupa analisis/teknik candlestick. Tidak hanya itu, ia juga mengenalkan konsep *doji* seperti *bearish* dan *bullish* atau yang disebut sebagai *yin* dan *yang*. Untuk zaman modern, Charles Dow atau yang dikenal dengan bapak analisis teknikal menjadi pelopor pertama dengan menciptakan sebuah teori analisis teknikal atau yang dikenal dengan teori dow (*dow theory*) yang sangat bepengaruh sampai sekarang. Kehadiran teori yang ia buat memicu pada perkembangan yang pesat dari analisis teknikal di akhir abad ke-19 bahakan menjadi dasar dari berbagai macam indicator analisis teknikal di zaman sekarang seperti *moving average*, *MACD*, *alligator*, dll (Hidayahtullah, 2022).

Dengan berbagai perkembangan teori dan teknik dalam analisis teknikal yang ada hingga sampai saat ini, tentunya menjadi alat yang penting bagi para investor dalam memahami perilaku pasar serta menjadi pendukung dalam menentukan setiap keputusan yang tepat. Oleh karena itu, investor dapat dengan mudah memahami setiap momentum yang tepat dalam setiap fase.

### **RSI (Relative Strength Index)**

Relative Strength Index (RSI) merupakan salah satu jenis indikator teknikal yang cukup popular dalam dunia analisis saham. Indikator ini diperkenalkan pertama kali oleh J. Welles Wilder dalam bukunya yang berjudul New Concept In Technical Analysis pada tahun 1978 (Hidayahtullah, 2022). Indikator ini digunakan untuk mengidentifikasi kondisi ekstrem momentum jangka pendek atau dengan kata lain untuk mengukur kecepatan perubahan dalam pergerakan harga saham. RSI juga mampu memberikan gambaran terkait dengan kondisi pasar apakah sedang mengalamai kondisi *overbought* atau *oversold*.RSI bergerak dalam rentang 1 sampai dengan 100 yang dimana ketika berada diatas rentang 70 maka mengalami kondisi *overbought* yang mengindikasikan bahwa waktunya untuk jual sedangkan apabila berada dibawah rentang 30 maka mengalami kondisi *oversold* yang mengindikasikan bahwa waktunya untuk beli(Hidayahtullah, 2022)(Monika & Yusniar, 2020). Pada umumnya, periode yang sering digunakan dalam RSI yakni 9, 14 dan 25. Akan tetapi tidak ada periode yang paling baik sebab semua bergantung pada kondisi pasar. Semakin kecil periode yang digunakan maka semakin sensitif terhadap fluktuasi harga jangka pendek (Hidayahtullah, 2022).

Untuk melakukan perhitungan pada Relative Strength Index (RSI) langkah pertama yang perlu dilakukan yakni menghitung nilai rata-rata penutupan naik/*up* (*up close* yakni harga penutupan yang lebih tinggi dibandingkan dengan harga penutupan sebelumnya) dan nilai rata-rata penutupan turun/*down* (*down close* yakni harga penutupan yang lebih rendah dibandingkan dengan harga penutupan sebelumnya) pada hari sebelumnya dalam periode waktu tertentu. Misalnya sebagai contoh menggunakan 14 hari harga penutupan sebelumnya. Setelah itu, lakukan pembagian dari hasil nilai rata-rata penutupan naik/*up* dengan nilai rata-rata penutupan turun/*down* sehingga diperoleh nilai RS. Berikut merupakan formula untuk mendapatkan nilai RS (Wilder, 1978) :

|  |  |
| --- | --- |
|  | (2.1) |

Keterangan:

*Average 14 day’s closes Up* : Harga rata-rata tertinggi selama 14 hari terakhir.

*Average 14 day’s closes Down* : Harga rata-rata terendah selama 14 hari terakhir.

Namun perlu diperhatikan bahwa untuk mendapatkan nilai rata-rata penutupan naik/*up* pada proses berikutnya yakni dengan melakukan perkalian pada nilai rata-rata penutupan naik/*up* sebelumnya dengan 13, lalu ditambahkan dengan jumlah penutupan naik/*up* hari ini (jika ada) dan bagi totalnya dengan 14. Begitu pula untuk mendapatkan nilai rata-rata penutupan turun/*down* pada prosesberikutnya yakni dengan melakukan perkalian pada nilai rata-rata penutupan turun/*down* sebelumnya dengan 13, lalu ditambahkan dengan jumlah penutupan turun/*down* hari ini (jika ada) dan bagi totalnya dengan 14. Kemudian setelah itu, lakukan pembagian dari hasil nilai rata-rata penutupan naik/*up* dengan nilai rata-rata penutupan turun/*down* sehingga diperoleh nilai RS. Setelah mendapatkan nilai RS maka kemudian dapat melakukan perhitungan nilai RSI dengan formula sebagai berikut (Wilder, 1978) :

|  |  |
| --- | --- |
|  | (2.2) |

Relative Strength Index (RSI) menjadi indikator teknikal yang cukup populer dalam analisis saham sebab kemampuannya mampu membantu dalam mengidentifikasi kondisi ekstrem momentum jangka pendek atau dengan kata lain bisa mengukur kecepatan perubahan dalam pergerakan harga saham overbought atau oversold. Relative Strength Index (RSI) juga mampu untuk melihat divergence yang terjadi diantara harga dan indikator. Oleh sebab itu, tidak heran jika Relative Strength Index (RSI) masih cukup popular dan relevan digunakan oleh para investor dalam melakukan analisis saham.

### **Manchester United**

Mendengar kata Manchester United tidak terlepas dari bayang-bayang klub sepak bola liga Inggris dengan segudang prestasi. Lantas, Manchester United bukan hanya sekedar klub sepak bola professional di Inggris melainkan juga perusahaan yang bergerak di bidang olahraga serta berinvestasi di bidang properti. Manchester United Ltd atau yang sekarang dikenal dengan Manchester United plc ini mengoperasikan Manchester United Football Club serta membangun hubungan pemasaran dan sponsorship dengan perusahaan-perusahaan internasional dan regional untuk meningkatkan merekanya (Manchester United, n.d.). Tidak hanya itu, Perusahaan ini juga menjual pakaian olahraga, pakaian rekreasi, pakaian latihan dan pakaian lainnya serta menjual produk berlisensi lainya dengan menampilkan merek dagang Manchester United. Produk-produk tersebut di distribusikan melalui pusat ritel bermerek Manchester United, platform *e-commerce* dan saluran distribusi grosir mitra perusahaan. Selain itu, perusahaan juga mendistribusikan konten sepak bola secara langsung dan bersama dengan mitra komersialnya serta menyiarkan hak siar televisi yang berkaitan dengan kompetisi klub bahkan menayangkan program-program Manchester United melalui saluran televisi MUTV ke berbagai belahan wilayah dunia (Manchester United, n.d.). Kini, Manchester United plc berkantor pusat di Manchester, Inggris.

### **SVM (Support Vector Machine)**

Dalam konteks analisis prediksi saham, sangat penting untuk memahami secara cermat setiap trend dan pergerakan yang terjadi. Hal ini memungkinkan untuk mengidentifikasi potensi cuan secara optimal. Namun hal tersebut tidak terlepas dari adanya analisis fundamental dan analisis teknikal yang telah cukup populer dikalangan para investor. Akan tetapi, pendekatan lain juga bisa dilakukan dalam melakukan analisis saham yakni dengan menggunakan metode algoritma support vector machine (SVM).

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vapnik pada tahun 1992 sebagai konsep unggulan dalam bidang pattern recognition dan memiliki kemampuan dalam memilih model otomatis serta tidak memiliki masalah overfitting. Sejauh ini, SVM tidak hanya mampu menyelesaikan kasus secara linier melainkan juga bisa berkerja pada kasus non-linier dengan menggunakan konsep karnel pada ruang kerja berdimensi tinggi, dengan mencari *hyperplane* yang dapat memaksimalkan jarak antar kelas data (Fadilah, Agfiannisa, & Azhar, 2020).

Tingkat keakurasian motode SVM sangat dipengaruhi oleh fungsi karnel dan parameter yang digunakan. Fungsi karnel ini digunakan untuk memetakan dimensi awal (dimensi yang lebih rendah) himpunan data ke dimensi baru (dimensi yang realtif tinggi). Lantas, fungsi karnel kelak akan menentukan fitur baru dimana *hyperplane* akan dicari serta menjadi garis pemisah beberapa kelompok data (Fadilah, Agfiannisa, & Azhar, 2020).

Dengan demikian, secara matematis *hyperplane* klasifikasi linier pada support vector machine (SVM) dapat dilihat melalui persamaan yakni:

|  |  |
| --- | --- |
|  | (2.3) |

atau dengan kata lain dapat juga menggunakan persamaan:

|  |  |
| --- | --- |
|  | (2.4) |
|  | (2.5) |

dimana untuk nilai merupakan himpunan bilangan traning, dan merupakan label dari kelas . Sedangkan untuk fungsi karnel sendiri digambarkan melalui persamaan sebagai berikut:

Tabel 2. 1 Rumus fungsi karnel

|  |  |  |
| --- | --- | --- |
| Karnel | Persamaan |  |
| Polynomial |  | (2.6) |
| RBF |  | (2.7) |

dimana untuk nilai dan merupakan pasang dari dua data traning dan parameter merupakan konstanta. Pemilihan fungsi karnel ini bergantung pada sifat data sehingga memainkan peran penting dalam mencari dan menentukan *hyperplane* yang terbaik. Oleh karena itu, tingkat akurasi dari metode support vector machine (SVM) sangat dipengaruhi sekali oleh faktor dari fungsi karnel dan parameter yang digunakan.

Keterangan:

: Bobot.

: Vektor fitur input.

: Bias.

: Vecktor fitur input himpunan bilangan traning dari bilangkan ke-i.

: Label dari kelas .

: Nilai kernel antar dua vector dan vector .

: Vektor fitur dari dua pasang data traning dan data traning .

: Jarak euclidien antara vector dan vector .

: Sigma.

: Konstanta.

: Derajat.

### **SMOTE (Synthetic Minority Over-sampling Technique)**

SMOTE (Synthetic Minority Over-sampling Technique) merupakan salah satu teknik *oversampling* yang digunakan untuk melakukan peyeimbangan data dengan menciptakan data berupa sampel sintetik buatan dengan nilai yang sama untuk kelas minoritas sehingga meningkatkan distribusi kelas minoritas (Shen, et al., 2021). Proses perhitungan sampel buatan ini dilakukan dengan melakukan indetifikasi terhadap vector fitur yang dilambangkan dengan dan K tetangga terdekat yang dilambangkan dengan . Kemudian, menghitung nilai selisih antara vector fitur dan tetangganya. Nilai selisih ini akan dilakukan perkalian dengan nilai acak antara 0 atau 1, kemudian hasilnya ditambahkan ke vector fitur untuk mengidentifikasi titik baru pada segmen baris. Berikut merupakan persamaan untuk menghitung sampel sintetik buatan (Zhao & Bai, 2022) :

|  |  |
| --- | --- |
|  | (2.8) |

Keterangan:

: Sampel buatan.

: Vector fitur.

: K tetangga terdekat.

: Nilai acak antara 0 atau 1.

### **Performance Metrics**

*Performance metrics* merupakan alat ukur yang digunakan untuk mengevaluasi kemampuan model dalam melakukan prediksi. Adanya *performance metrics* ini membantu dalam menghitung jumlah data yang diprediksi secara benar maupun salah oleh model. Oleh karena itu, untuk membangun *performance metrics* yang mencerminkan kinerja model secara keseluruhan maka dibutuhkan beberapa metrix kinerja seperti akurasi, presisi, recall dan F1-score. Berikut merupakan definisi serta formula dari masing-masing metrix (Anggrawan, Hairani, & Satria, 2023) (Shen, et al., 2021) :

1. Akurasi merujuk pada hasil pengukuran dengan nilai yang aktual atau nilai sebenarnya.

|  |  |
| --- | --- |
|  | (2.9) |

1. Presisi merujuk pada seberapa dekat perbedaan hasil pengukuran pada pengukuran yang berulang.

|  |  |
| --- | --- |
|  | (2.10) |

1. Recall merujuk pada tingkat keberhasilan dalam mengambil informasi.

|  |  |
| --- | --- |
|  | (2.11) |

1. F1-Score merujuk pada nilai rata-rata antara nilai presisi dan nilai recall.

|  |  |
| --- | --- |
|  | (2.12) |

Keterangan:

TP (*True Positif*) : Kelas yang diprediksi benar.

TN (*True Negatif*) : Kelas yang sebenarnya salah namun diprediksi benar.

FP (*False Positif*) : Kelas yang diprediksi salah.

FN (*False Negatif*) : Kelas yang sebenarnya benar namun diprediksi salah.

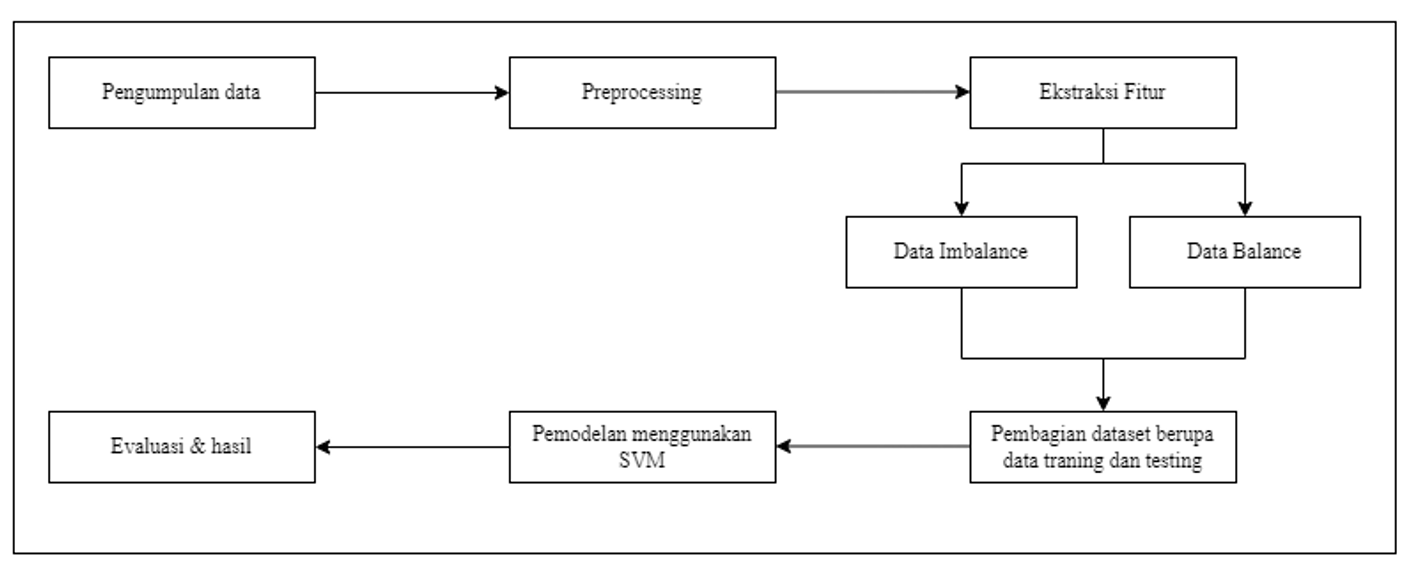
## **Kajian Hasil Riset**

Tabel 2. 2 Hasil kajian teori

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Peneliti, Penerbit, dan Tahun Terbit** | **Judul** | **Algoritma** | **Hasil Penelitian** |
| 1 | Peneliti (Wulandari & Anubhakti, 2022)  Ratih Febrilia Tri Wulandari & Dian Anubhakti  Penerbit dan Tahun Terbit  Indonesia Journal Information System (IDEALIS), vol 4 no 2 tahun 2021 | Implementasi Algoritma Support Vector Machine (SVM) Dalam Memprediksi Harga Saham PT. Garuda Indonesia Tbk | Support Vector Machine (SVM) | Hasil dari pengujian model SVM yakni menghasilkan performance prediction trend accuracy sebesar 0.545 dari 520 dataset yang ada dan dengan hasil nilai trend akurasi prediksi tersebut bahwa SVM dapat digunakan sebagai model untuk memprediksi harga saham penutupan pada PT. Garuda Indonesia, Tbk. |
| 2 | Peneliti (Syahputra, Ramadhan, & Burha, 2022)  Janur Syahputra, Rima Dias Ramadhan & Auliya Burhanudin  Penerbit dan Tahun Terbit  Journal of Dinda, vol 2 no 1 tahun 2022 | Prediksi Harga Saham Bank BRI Menggunakan Algoritma Linier Regression Sebagai Strategi Jual Beli Saham | Linier Regression | Penelitian ini melakukan pembagian data yakni 60:40, 65:35, 70:30, 75:25, dan 80:20 untuk mengetahui akurasi latih dan uji tertinggi. Rasio terbaik ada pada 80:20 yang menghasilkan akurasi train dan test sebesar 0,89 dan 0,91. Kemudian masing-masing data training dan data testing dimasukan kedalam model linear regression untuk kemudian dilakukan prediksi. Adapun hasil akurasi yang dihasilkan dari model yang sudah diuji dengan data prediksi dalam bentuk R¬2 yaitu sebesar 0.9127. Hasil error dari prediksi dihitung menggunakan MAPE dan menghasilkan persentase sebesar 13,751% untuk data pelatihan, 13,773% untuk data pengujian, dan 13,755% untuk data keseluruhan. Ketiga hasil tersebut mengidikasikan bahwa persentase error berada pada rentang 10% - 20%. |
| 3 | Peneliti (Maulana & Kumalasari, 2019)  Reza Maulana & Devy Kumalasari  Penerbit dan Tahun Terbit  Jurnal Informatika Kaputama (JIK), vol 3 no 1 tahun 2019 | Analisis Dan Perbandingan Algoritma Data Mining Dalam Prediksi Harga Saham GGRM | Algoritma Data Mining seperti Neural Network, Linear Regression, Support Vector Machine, Gaussian Process, dan Polynomial Regression. | Dalam penelitian ini akan dilakukan prediksi harga saham GGRM dengan membandingkan beberapa model algoritma yaitu Neural Network, Linear Regression, Support Vector Machine, Gaussian Process, dan Polynomial Regression. Untuk mengukur tingkat akurasi dari tiap model algoritma digunakan model validasi 10 Fold Cros Validation dan evaluasi menggunakan Root Mean Square Error (RMSE). Hasil penelitian diperoleh sebagai berikut:   * Algoritma NN memperoleh RMSE 612.472 +/- 89.402 (mikro 618.916 +/- 0.000) * Algoritma LR memperoleh RMSE 659.806 +/- 76.480 (mikro 664.360 +/- 0.000) * Algoritma SVM memperoleh RMSE 7668.182 +/- 627.179 (mikro 7694.784 +/- 0.000) * Algoritma GP memperoleh RMSE 52295.739 +/-448.393 (mikro 52297.714 +/- 0.000) * Algoritma PR memperoleh RMSE 13049082159319.207 +/-19076530613123.832 (mikro 23133462625506.945 +/- 0.000)   Dapat dilihat bahwa algoritma Neural Network memiliki hasil akurasi prediksi RMSE paling kecil yakni 612.474 +/- 89.402 (mikro: 618.916 +/- 0.000) dibandingkan dengan model algoritma lainnya. |
| 4 | Peneliti (Lestari & Pangaribuan, 2020)  Jefri Junifer Pangaribuan & Megawaty Lestari  Penerbit dan Tahun Terbit  Information System Development, vol 5 no 1 tahun 2020 | Perbandingan Metode Moving Average (MA) Dan Neural Network Yang Berbasis Algoritma Backpropagation Dalam Prediksi Harga Saham | Metode Moving Average (MA) Dan Neural Network Yang Berbasis Algoritma Backpropagation | Prediksi harga saham dengan metode Moving Average (MA) lebih akurat dibandingkan dengan Neural Network algoritma Backpropagation, dimana tingkat akurasi untuk Moving Average (MA) adalah 80,11% dan untuk Neural Network algoritma Backpropagation adalah 78,91%. |
| 5 | Peneliti (Lestari & Jasuni, 2023)  Siti Sarah Sobariah Lestari & Ana Yuliana Jasuni  Penerbit dan Tahun Terbit  Jurnal Bisnisman; Riset Bisnis dan Manajemen, vol 5 no 1 tahun 2023 | Analisis Rantai Makrov Lima Status Pada Return Harga Saham BBCA | Rantai Makrov | Berdasarkan hasil analisis yang telah dilakukan, Dalam jangka panjang peluang return harga saham harian penutupan Bank Central Asia Tbk. Terbesar terjadi pada status 2 dimana saaat return harga saham bernilai lebih besar dari -1 dan lebih kecil dari 0 dengan peluang sebesar 0,4186. |
| 6 | Peneliti (Fadilah, Agfiannisa, & Azhar, 2020)  Widya Rizka Ulul Fadilah, Dewi Agfiannisa & Yufis Azhar  Penerbit dan Tahun Terbit  Fountain of Informatics Journal, vol 5 no 2 tahun 2020 | Analisis Prediksi Harga Saham PT. Telekomunikasi Indonesia Menggunakan Metode Support Vector Machine | Support Vector Machine | Peneliti menggunakan Split Validation untuk melakukan pembagian data yakni 20% testing dan 80% training. Setelah itu, peneliti memilih karnel SVM yaitu kernel RBF untuk digunakan dalam algoritma SVM berdasarkan hasil pengujian dibandingkan dengan kernel lain seperti polynomial. Hal ini dikarenakan hasil pengujian yang diperoleh mendapatkan nilai akurasi tertinggi yakni 0.9653 dan nilai RMSE sebesar 0.0091. Peneliti juga melakukan pengujian dengan mencoba menggunakan algoritma KNN sebagai pembanding untuk menilai apakah SVM lebih unggul atau tidak dengan menggunakan dataset yang sama. Hasil yang diperoleh menggunakan algoritma KNN yakni nilai akurasi sebesar 0.9456 dan RMSE sebesar 0.1162. Dengan demikian, dapat diketahui bahwa algoritma SVM lebih baik ketimbang algoritma KNN dalam prediksi harga saham PT. Telekomunikasi. Hal ini dikarenakan algoritma SVM hampir medekati nilai yang sebenarnya dan tingkat error yang dihasilkan oleh metode KNN lebih besar dibandingkan oleh metode SVM. |

# **METODE PENELITIAN**

Dengan menggunakan data berupa histori pergerakan harga harian saham Manchester United (MANU) akan dilakukan analisis prediksi harga saham dengan menggunakan metode support vector machine (SVM) sehingga secara garis besar akan menghasilkan gambaran umum yakni sebagai berikut:



Gambar 3. 1 Gambaran umum penelitian

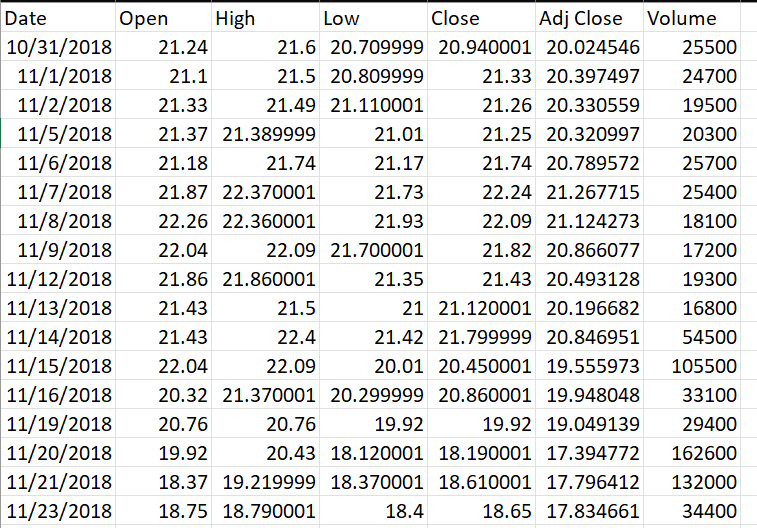
Gambar 3. 1 tersebut merupakan proses atau tahapan yang akan dilakukan dalam penelitian sehingga dapat menjadi peta/alur dalam penelitian. Hal ini tentunya sangat membantu peneliti untuk mencapai pemahaman mendalam tentang langkah-langkah yang perlu diambil dalam mencapai tujuan penelitian. Oleh karena itu, jikalau merujuk pada gambaran umum maka peneliti akan melewati beberapa proses yakni pengumpulan data histori pergerakan harga harian saham Manchester United (MANU), preprocessing data yang bertujuan supaya data yang digunakan dapat lebih efektif dalam proses analisis lebih lanjut, ekstraksi fitur yang bertujuan untuk mengindentifikasi sekaligus mengekstraksi fitur yang relevan, balence data yang bertujuan agar data seimbang sehingga tidak cenderung pada kelas data tertentu, pembagian dataset yang meliputi data traning dan data testing, pemodelan serta evaluasi dan hasil untuk mengetahui sejauh mana model berkerja dengan baik dan optimal dalam menjawab permasalahan yang dirumuskan. Detail dari setiap proses pada gambaran umum tersebut akan dijabarkan sebagai berikut:

## **Pengumpulan Data**

Pada penelitian ini data yang digunakan ialah data pergerakan harian harga saham Manchester United dengan kode saham yang terdaftar pada New York Stock Exchange (NYSC) ialah MANU. Data tersebut diperoleh melalui <https://finance.yahoo.com/> dalam rentang waktu 5 tahun terakhir yakni pada 31 Oktober 2018 sampai 30 Oktober 2023 dengan jumlah data 1257. Pada data tersebut mengandung 7 atribut yang dikemas dalam format file csv yakni sebagai berikut:

1. Date merupakan tanggal transaksi harga saham berlangsung.
2. Open merupakan harga buka saham pada waktu tertentu dengan tipe data bilangan desimal.
3. Close merupakan harga tutup saham pada waktu tertentu dengan tipe data bilangan desimal.
4. High merupakan harga tertinggi saham pada waktu tertentu dengan tipe data bilangan desimal.
5. Low merupakan harga terendah saham pada waktu tertentu dengan tipe data bilangan desimal.
6. Adj Close merupakan harga tutup yang disesuaikan pada waktu tertentu dengan tipe data bilangan desimal.
7. Volume merupakan jumlah saham yang diperdagangkan pada waktu tertentu dengan tipe data bilangan desimal.

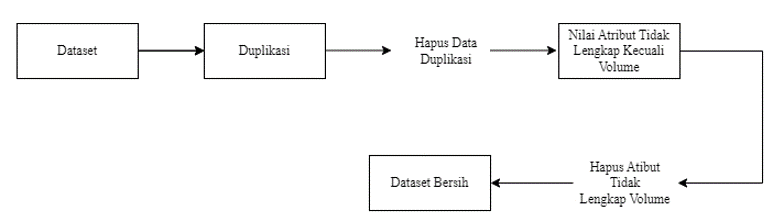
Berikut merupakan Gambar 3. 2 yang menampilkan contoh data dari pergerakan harian harga saham Manchester United (MANU) dalam format CSV :



Gambar 3. 2 Contoh data harga saham Manchester United

## **Preprocessing**

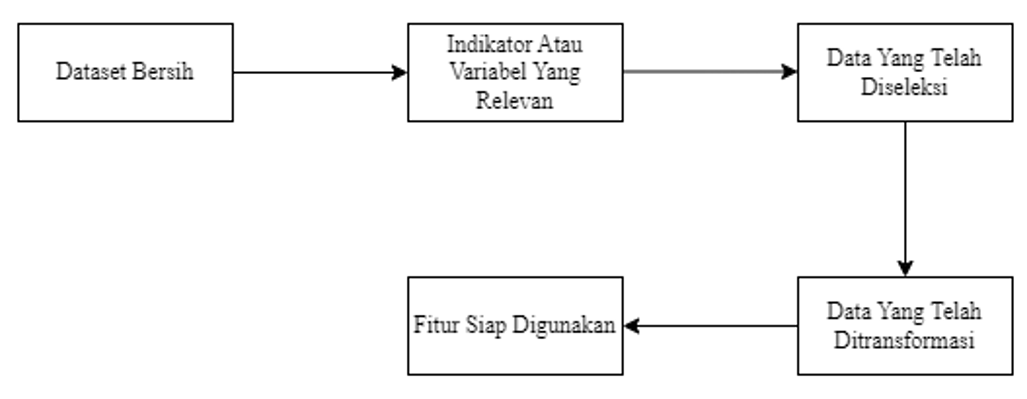
Pada tahap preprocessing akan dilakukan terlebih dahulu pengecekan untuk setiap nilai apakah mengandung duplikasi. Jika terdapat nilai yang mengandung duplikasi maka akan dilakukan pendekatan berupa penghapusan terhadap nilai tersebut. Akan tetapi, dikarenakan data yang dimiliki tidak mengandung duplikasi maka tidak perlu melakukan tindakan tersebut namun tetap dilakukan pengecekan untuk memastikan data tersebut benar-benar tidak mengandung duplikasi. Kemudian setelah itu, dikarenakan nilai pada dataset telah bertipe data float dan integer maka tidak perlu untuk melakukan konversi tipe data sehingga bisa dilakukan cleaning pada beberapa nilai atribut yang tidak lengkap untuk dihapus datanya. Hal ini bertujuan untuk meminimalisir tingkat error dalam proses pengklasifikasian data. Namun, hal ini tidak dilakukan pada atribut volume sebab atribut tersebut menggambarkan jumlah saham yang diperdagangkan sehingga bisa saja pada waktu tertentu volume transaksi bernilai 0 yang artinya tidak adanya penjualan saham yang terjadi pada hari tersebut. Berikut merupakan Gambar 3. 3 dari proses preprocessing:



Gambar 3. 3 Alur proses preprocessing

## **Ekstraksi Fitur**

Ekstraksi fitur bertujuan untuk mengidentifikasi, memilih dan menggunakan berbagai indikator atau variabel yang dianggap relevan dalam mengembangkan model prediksi. Hal ini disebabkan karena tidak semua atribut data memiliki korelasi satu sama lainnya sehingga menyebabkan hadirnya data redundan sehingga bisa dihilangkan dalam kategorisasi atau klasifikasi. Sebagai contoh memilih dan menggunakan atribut close untuk menghitung indikator Relative Strength Index (RSI). Pemilihan penggunaan indikator RSI dalam ekstraksi fitur dikarenakan kemampuannya dalam mengidentifikasi kondisi ekstrem momentum jangka pendek atau dengan kata lain bisa mengukur kecepatan perubahan dalam pergerakan harga saham sehingga dapat memberikan sinyal/gambaran terkait dengan kondisi/fase saham yang sedang mengalamai *overbought* atau *oversold* serta mampu untuk melihat divergence yang terjadi diantara harga dan indikator. Namun, tidak dapat dipungkiri bahwa RSI juga bisa saja menghasilkan *false signal* pada pasar yang sedang trend. Berikut merupakan Gambar 3. 4 dari proses ekstraksi fitur:

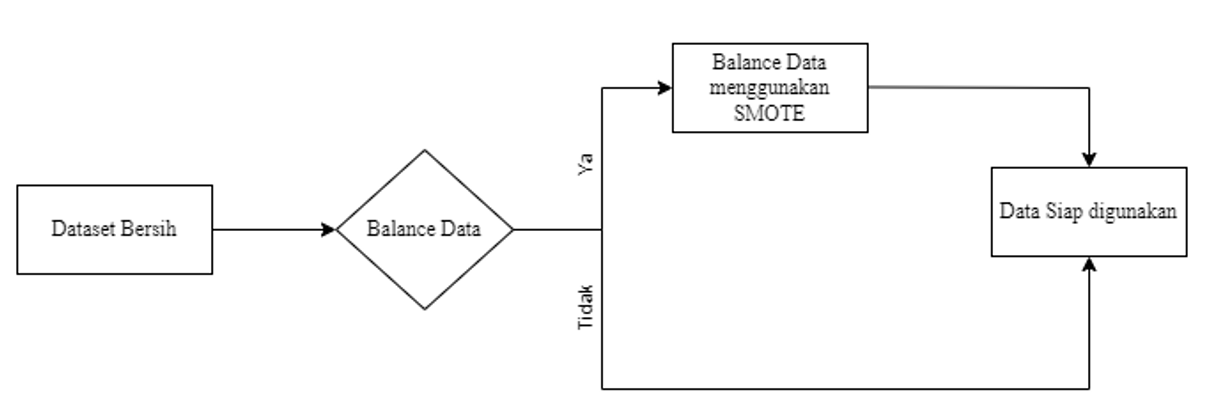


Gambar 3. 4 Alur proses ekstraksi fitur

## **Eksplorasi Data: Balance vs Imbalance**

Ketika data telah dilakukan ekstarksi fitur, maka hal penting yang dilakukan selanjutnya ialah mengecek apakah data tersebut mengandung data yang *balance* atau data yang *imbalance*. Hal ini sangat penting untuk dilakukan supaya mengetahui karakteristik data dalam membangun model yang akurat. Oleh karena itu, dilakukan eksplorasi dengan dua pendekatan yang berbeda.

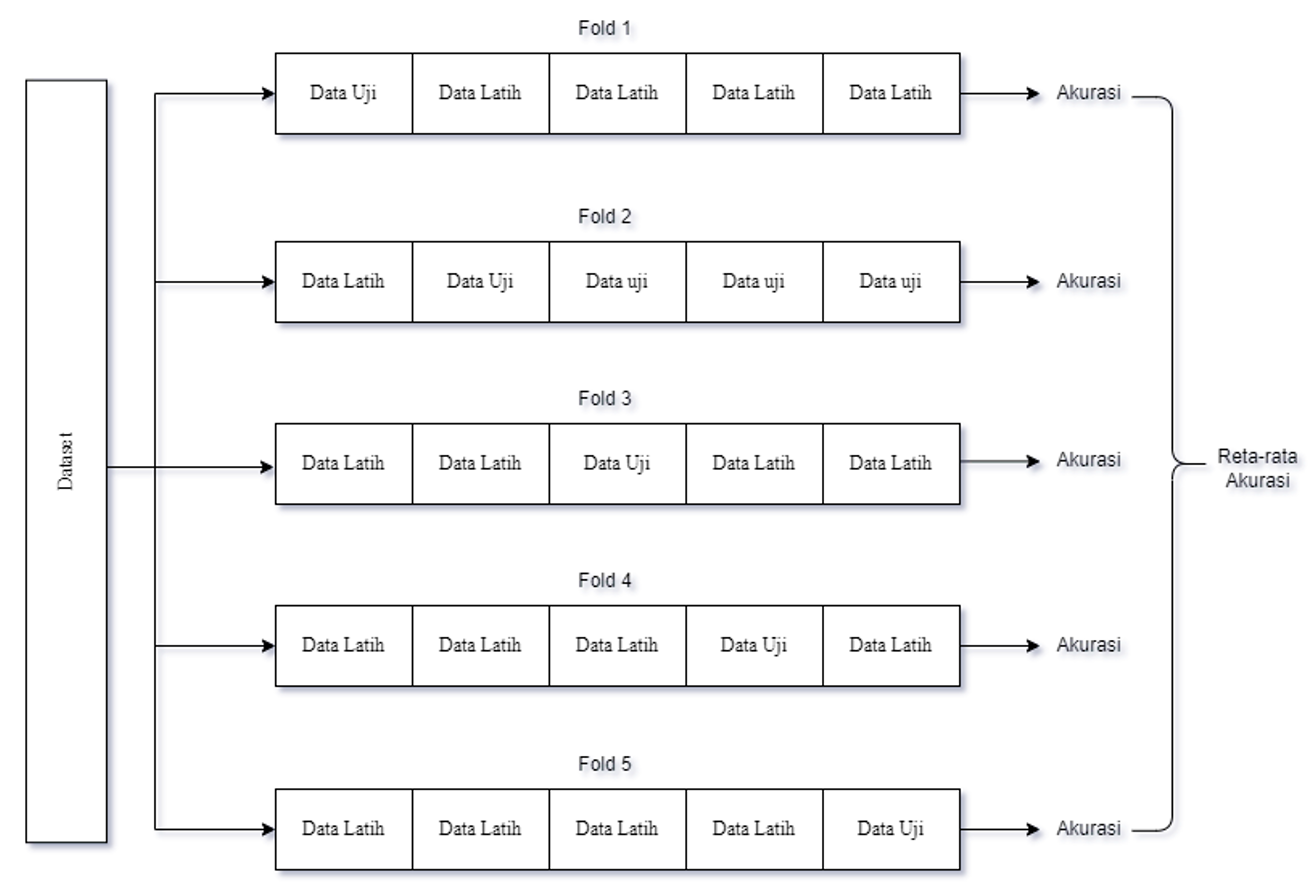
Pertama, mencoba untuk melakukan *balance* data dengan menggunakan teknik *oversampling* atau meningkatkan jumlah kelas minoritas agar seimbang dengan kelas mayoritas. Metode yang digunakan pada *oversampling* yakni SMOTE (Synthetic Minority Over-sampling Technique) yang akan menghasilkan sampel-sampel sintetik untuk kelas minoritas berdasarkan interpolasi atau kombinasi sampel-sampel yang ada pada kelas minoritas. Dengan demikian, data yang dimiliki telah seimbang. Kedua, mencoba untuk mempertahankan data asli tanpa melakukan penyeimbangan data atau dalam artian lain, data tersebut tidak dilakukan proses apapun melainkan langsung digunakan dalam membangun model. Berikut merupakan Gambar 3. 5 dari proses kedua pendekatan diatas yakni *balance* data dan *imbalance* data:



Gambar 3. 5 Alur proses balance data dan imbalance data

## **Pembagian Data**

Tujuan adanya pembagian antara data latih dan data uji yakni untuk mengetahui performa dari proses pelatihan model (Syahputra, Ramadhan, & Burha, 2022). Pada proses ini akan dilakukan pembagian data dengan persentasi yakni 80 % untuk data latih dan 20 % untuk data uji. Pembagian data tersebut menggunakan metode k-fold cross validation dengan nilai k-fold yakni 3, 5, 7 dan 9 sehingga hal ini akan memungkinkan untuk memperoleh akurasi yang lebih baik. Berikut merupakan Gambar 3. 6 dari proses k-fold cross validation:



Gambar 3. 6 Alur proses metode k-flod cross validation

## **Modeling**

Selain mampu bekerja dan menangani kasus yang bersifat linier, SVM juga dapat menangani kasus yang bersifat non-linier dengan memaksimalkan konsep karnel. Karnel merupakan salah satu metode non parametrik yang berfungsi untuk memetakan data dalam ruang berdimensi tinggi (Miraltamirus, Fitri, Vionanda, & Permana, 2023). Dengan adanya pemilihan karnel yang tepat, tentunya dapat meningkatkan kinerja model lebih akurat. Oleh karena itu, pada penelitian ini, karnel yang digunakan ialah kernel polynomial dan karnel RBF yang bertujuan untuk meningkatkan optimasi metode SVM dan diharapkan bisa memperoleh akurasi terbaik dengan membandingkan variabel (Fadilah, Agfiannisa, & Azhar, 2020). Selain itu, nilai parameter C yang digunakan yakni 1, 10, 100 dan 1000 serta parameter gamma yakni 1, 0.1, 0.01, 0.001, dan 0.0001 (Yudhawan & Purwaningsih, 2020) (Fauzi, 2021) sehingga dari kedua karnel tersebut kelak akan dipilih yang menghasilkan hyperplane dengan nilai yang paling optimal.

## **Evaluasi**

Setelah melatih model maka kemudian melakukan evaluasi model dengan melihat hasil dari kinerja model yang telah dibangun. Evaluasi model tersebut dituangkan dalam confusion metrix atau metrix seperti akurasi, presisi, recall dan F1-score untuk melihat kemampuan prediksi model.

## **Skenario Pengujian**

Pada penelitian ini terdapat beberapa skenario pengujian yang akan dilakukan baik pada data yang *balance* maupun data yang *imbalace* yakni seperti pembagian data menggunakan metode k-fold cross validation dengan nilai k-fold yakni 3, 5, 7 dan 9 serta karnel yang digunakan berupa polynomial dan RBF dengan parameter C yakni 1, 10, 100 dan 1000 serta parameter gamma yakni 1, 0.1, 0.01, 0.001, dan 0.0001 (Yudhawan & Purwaningsih, 2020) (Fauzi, 2021). Berikut merupakan rincian terkait perihal diatas:

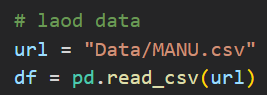
Tabel 3. 1 Skenario pengujian

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **K-Fold** | **Parameter** | | | |
| **Kernel RBF** | | **Kernel Polynomial** | |
| **C** | **Gamma** | **C** | **Gamma** |
| 1 | 3-fold | 1 | 1 | 1 | 1 |
| 2 | 0.1 | 0.1 |
| 3 | 0.01 | 0.01 |
| 4 | 0.001 | 0.001 |
| 5 | 0.0001 | 0.0001 |
| 6 | 10 | 1 | 10 | 1 |
| 7 | 0.1 | 0.1 |
| 8 | 0.01 | 0.01 |
| 9 | 0.001 | 0.001 |
| 10 | 0.0001 | 0.0001 |
| 11 | 100 | 1 | 100 | 1 |
| 12 | 0.1 | 0.1 |
| 13 | 0.01 | 0.01 |
| 14 | 0.001 | 0.001 |
| 15 | 0.0001 | 0.0001 |
| 16 | 1000 | 1 | 1000 | 1 |
| 17 | 0.1 | 0.1 |
| 18 | 0.01 | 0.01 |
| 19 | 0.001 | 0.001 |
| 20 | 0.0001 | 0.0001 |
| 21 | 5-fold | 1 | 1 | 1 | 1 |
| 22 | 0.1 | 0.1 |
| 23 | 0.01 | 0.01 |
| 24 | 0.001 | 0.001 |
| 25 | 0.0001 | 0.0001 |
| 26 | 10 | 1 | 10 | 1 |
| 27 | 0.1 | 0.1 |
| 28 | 0.01 | 0.01 |
| 29 | 0.001 | 0.001 |
| 30 | 0.0001 | 0.0001 |
| 31 | 100 | 1 | 100 | 1 |
| 32 | 0.1 | 0.1 |
| 33 | 0.01 | 0.01 |
| 34 | 0.001 | 0.001 |
| 35 | 0.0001 | 0.0001 |
| 36 | 1000 | 1 | 1000 | 1 |
| 37 | 0.1 | 0.1 |
| 38 | 0.01 | 0.01 |
| 39 | 0.001 | 0.001 |
| 40 | 0.0001 | 0.0001 |
| 41 | 7-fold | 1 | 1 | 1 | 1 |
| 42 | 0.1 | 0.1 |
| 43 | 0.01 | 0.01 |
| 44 | 0.001 | 0.001 |
| 45 | 0.0001 | 0.0001 |
| 46 | 10 | 1 | 10 | 1 |
| 47 | 0.1 | 0.1 |
| 48 | 0.01 | 0.01 |
| 49 | 0.001 | 0.001 |
| 50 | 0.0001 | 0.0001 |
| 51 | 100 | 1 | 100 | 1 |
| 52 | 0.1 | 0.1 |
| 53 | 0.01 | 0.01 |
| 54 | 0.001 | 0.001 |
| 55 | 0.0001 | 0.0001 |
| 56 | 1000 | 1 | 1000 | 1 |
| 57 | 0.1 | 0.1 |
| 58 | 0.01 | 0.01 |
| 59 | 0.001 | 0.001 |
| 60 | 0.0001 | 0.0001 |
| 61 | 9-fold | 1 | 1 | 1 | 1 |
| 62 | 0.1 | 0.1 |
| 63 | 0.01 | 0.01 |
| 64 | 0.001 | 0.001 |
| 65 | 0.0001 | 0.0001 |
| 66 | 10 | 1 | 10 | 1 |
| 67 | 0.1 | 0.1 |
| 68 | 0.01 | 0.01 |
| 69 | 0.001 | 0.001 |
| 70 | 0.0001 | 0.0001 |
| 71 | 100 | 1 | 100 | 1 |
| 72 | 0.1 | 0.1 |
| 73 | 0.01 | 0.01 |
| 74 | 0.001 | 0.001 |
| 75 | 0.0001 | 0.0001 |
| 76 | 1000 | 1 | 1000 | 1 |
| 77 | 0.1 | 0.1 |
| 78 | 0.01 | 0.01 |
| 79 | 0.001 | 0.001 |
| 80 | 0.0001 | 0.0001 |

# **PEMBAHASAN**

## **Data**

Pada tahap ini, dilakukan *load* data menggunakan data pergerakan harian harga saham Manchester United pada bursa efek New York Stock Exchange (NYSC) dalam rentang waktu 5 tahun terkahir (31 Oktober 2018 sampai 30 Oktober 2023) serta dikemas dalam file csv. Perintah yang digunakan untuk melakukan load data tersebut yaitu *pd.read\_csv(url)*. Berikut merupakan Gambar 4. 1 yang menampilkan code pada program yang digunakan:



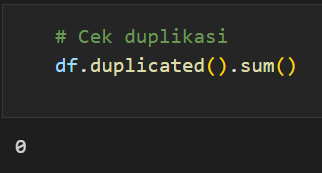
Gambar 4. 1 Code untuk load data

## **Preprocessing Data**

Preprosessing data merupakan proses untuk melakukan pembersihan data sehingga data yang digunakan kelak dapat terhindar dari noise dan hal-hal yang tidak penting. Berikut beberapa tahapan pada proses preprocessing yang dilakukan terhadap data:

### **Menghapus Duplikasi**

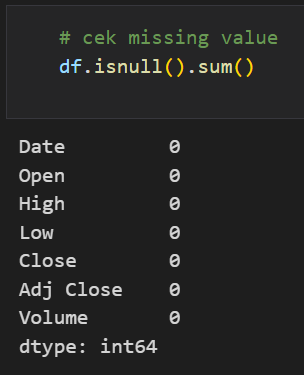
Pada tahap ini, dilakukan pengecekan terhadap data untuk memastikan apakah terdapat nilai yang mengandung duplikasi atau tidak. Jika terdapat nilai yang menggandung duplikasi maka akan dihapus. Namun, jika tidak tidak terdapat nilai yang yang menggandung duplikasimaka akan dilakukan analisa lebih lanjut tanpa melewati proses penghapusan. Perintah yang digunakan untuk melakukan pengecekan duplikasi tersebut yaitu *df.duplicated ().sum()*. Berikut merupakan Gambar 4. 2 yang menampilkan code pada program yang digunakan:



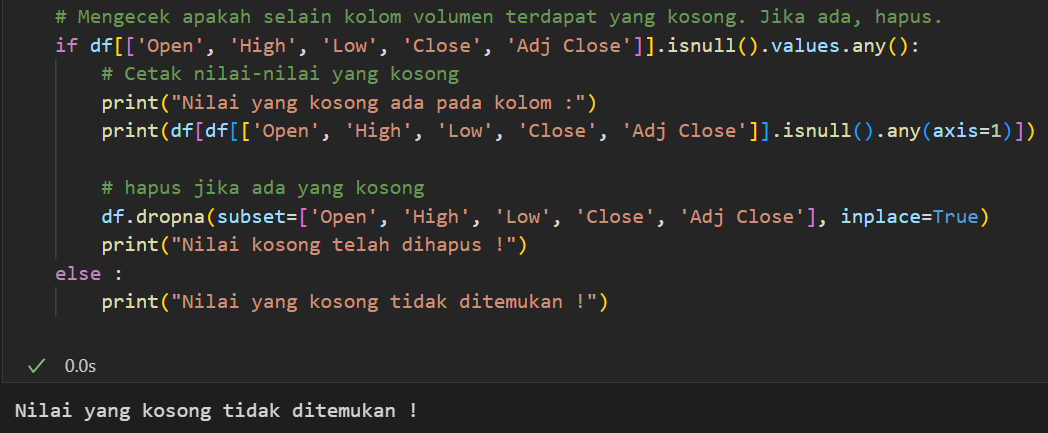
Gambar 4. 2 Code untuk cek duplikasi nilai

### **Menghapus Atribut Tidak Lengkap**

Pada tahap ini, dilakukan pengecekan terhadap nilai dari tiap-tiap atribut untuk memastikan apakah terdapat nilai yang kosong. Untuk melakukan pengecekan tersebut bisa menggunakan perintah yaitu *df.isnull().sum()*. Jika terdapat nilai yang kosong dari tiap-tiap atribut kecuali nilai dari atribut volume maka akan dihapus. Pengecualian penghapusan terhadap nilai pada atribut volume dikarenakan atribut ini menggambarkan jumlah saham yang diperdagangkan sehingga bisa saja pada waktu tertentu volume transaksi bernilai 0 yang artinya tidak adanya penjualan saham yang terjadi pada hari tersebut. Berikut merupakan Gambar 4. 3 yang menampilkan code pada program yang digunakan untuk mengetahui nilai yang kosong dari tiap-tiap atribut dan Gambar 4. 4 yang menampilkan code pada program yang digunakan untuk mengahpus nilai atribut yang kosong selain atritbut volume:



Gambar 4. 3 Code untuk mengetahui nilai yang kosong dari tiap-tiap atribut



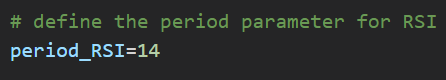
Gambar 4. 4 Code untuk cek nilai atribut yang tidak lengkap

## **Ekstraksi Fitur**

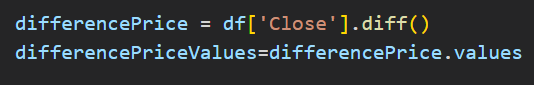
Ekstraksi fitur bertujuan untuk mengidentifikasi dan menemukan informasi yang relevan untuk diolah sehingga menghasilkan fitur yang bermakna. Selain itu juga, dapat memperkaya informasi yang dapat memudahkan model untuk lebih mengenali pola-pola penting sehingga dapat meningkatkan kemampuan model untuk menghasilkan prediksi yang akurat. Berikut merupakan tahapan pada proses ekstraksi fitur yang dilakukan terhadap data:

### **Menemukan Nilai RSI (Relative Strength Index)**

Pada tahap ini, hal pertama yang perlu dilakukan adalah menentukan jumlah periode yang digunakan untuk mencari nilai Relative Strength Index (RSI) yakni 14 hari. Kemudian setelah itu, menghitung nilai selisih antara atribut *Close* hari ini dengan nilai atribut *Close* hari sebelumnya. Berikut merupakan Gambar 4. 5 dan Gambar 4. 6 yang menampilkan code pada program untuk menentukan jumlah periode yang digunakan sekaligus menghitung nilai selisih antara atribut *Close* hari ini dengan nilai atribut *Close* hari sebelumnya:

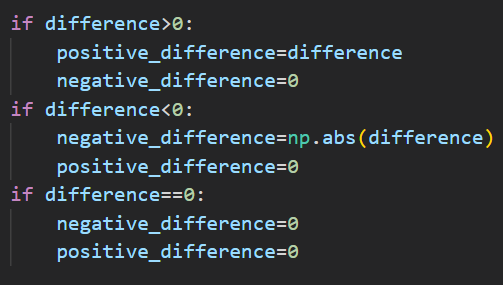


Gambar 4. 5 Code untuk jumlah periode yang digunakan



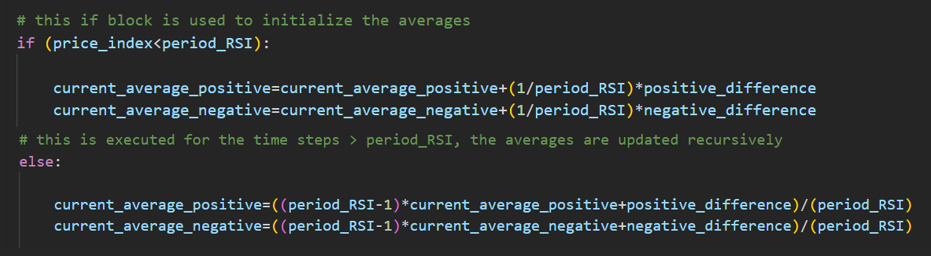
Gambar 4. 6 code untuk menghitung nilai selisih antara atribut Close hari ini dengan nilai atribut Close hari sebelumnya

Setelah berhasil menentukan nilai selisih antara atribut *Close* hari ini dengan nilai atribut *Close* hari sebelumnya, maka kemudian menentukan atau mengelompokkan nilai penutupan naik/*up* (*up close* yakni harga penutupan yang lebih tinggi dibandingkan dengan harga penutupan sebelumnya) dan nilai penutupan turun/*down* (*down close* yakni harga penutupan yang lebih rendah dibandingkan dengan harga penutupan sebelumnya). Berikut merupakan Gambar 4. 7 yang menampilkan code pada program yang digunakan:



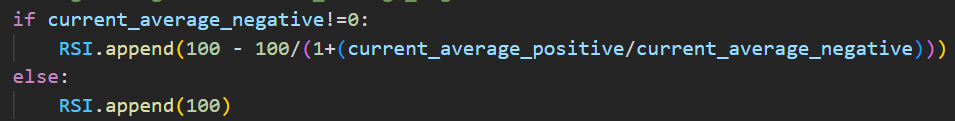
Gambar 4. 7 Code untuk menentukan atau mengelompokkan nilai penutupan naik/up dan nilai penutupan turun/down

Ketika telah berhasil mengelompokkan nilai atribut penutupan naik/*up* dan nilai atribut penutupan turun/*down* maka langkah selanjutnya yakni menghitung nilai rata-rata penutupan naik/*up* (*up close* yakni harga penutupan yang lebih tinggi dibandingkan dengan harga penutupan sebelumnya) dan nilai rata-rata penutupan turun/*down* (*down close* yakni harga penutupan yang lebih rendah dibandingkan dengan harga penutupan sebelumnya) pada hari sebelumnya dalam periode waktu tertentu. Misalnya sebagai contoh menggunakan 14 hari harga penutupan sebelumnya. Setelah itu, lakukan pembagian dari hasil nilai rata-rata penutupan naik/*up* dengan nilai rata-rata penutupan turun/*down* sehingga diperoleh nilai RS. Namun perlu diperhatikan bahwa untuk mendapatkan nilai rata-rata penutupan naik/*up* pada proses berikutnya yakni dengan melakukan perkalian pada nilai rata-rata penutupan naik/*up* sebelumnya dengan 13, lalu ditambahkan dengan jumlah penutupan naik/*up* hari ini (jika ada) dan bagi totalnya dengan 14. Begitu pula untuk mendapatkan nilai rata-rata penutupan turun/*down* pada prosesberikutnya. Berikut merupakan Gambar 4. 8 yang menampilkan code pada program yang digunakan:

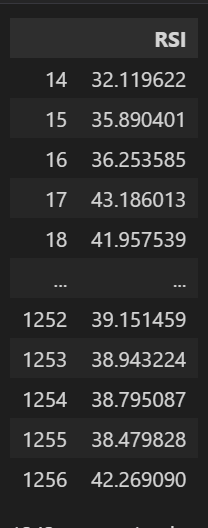


Gambar 4. 8 Code untuk menghitung nilai rata-rata penutupan naik/up dan rata-rata penutupan turun/down

Ketika proses perhitungan nilai rata-rata penutupan naik/*up* dan nilai rata-rata penutupan turun/*down* berhasil dilakukan. Maka barulah kemudian, menentukan nilai RS yakni melakukan pembagian dari hasil nilai rata-rata penutupan naik/*up* dengan nilai rata-rata penutupan turun/*down* sehingga diperoleh nilai RS atau lihat rumus RS pada (2.1). Kemudian, setelah itu, barulah bisa menemukan nilai RSI dengan melakukan perhitungan menggunakan rumus/persamaan pada (2.2). Berikut merupakan Gambar 4. 9 dan Gambar 4. 10 yang menampilkan code pada program sekaligus hasil dari perhitungan nilai RSI tersebut:



Gambar 4. 9 Code untuk menghitung nilai RSI



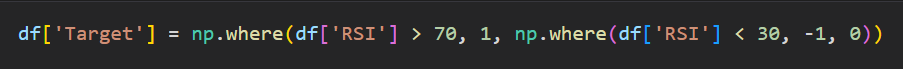
Gambar 4. 10 Output dari hasil perhitungan RSI

### **Menentukan Target/Label**

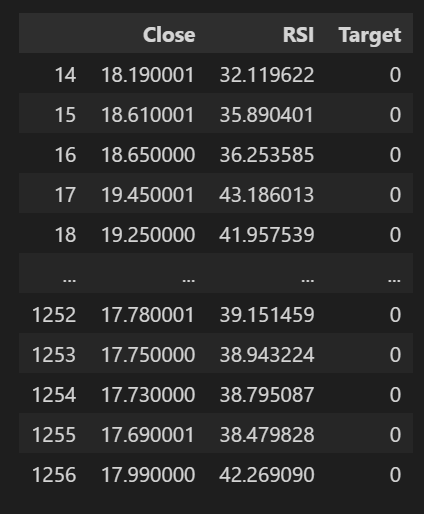
Pada tahap ini akan akan dilakukan penentuan target atau label setelah berhasil mendapatkan nilai dari perhitungan Relative Strength Index (RSI). Penentuan target/label didasarkan dengan kriteria sebagai berikut:

1. Jika nilai Relative Strength Index (RSI) kurang dari 30 maka mengindikasikan terjadi *oversold* (jenuh beli) atau waktunya untuk beli. Oleh karena itu, jika mencapai kondisi tersebut maka beri target/label -1.
2. Jika nilai Relative Strength Index (RSI) lebih dari 70 maka mengindikasikan terjadi *overbought* (jenuh jual) atau waktunya untuk jual. Oleh karena itu, jika mencapai kondisi tersebut maka beri target/label 1.
3. Jika nilai Relative Strength Index (RSI) berada pada rentang 30 sampai 70 maka mengindikasikan bahwa tidak terjadinya overbought dan oversold atau dengan kata lain berada dalam kondisi netral tanpa tekanan jual atau beli yang ekstrim. Oleh karena itu, jika mencapai kondisi tersebut maka beri target/label 0.

Berdasarkan kriteria diatas maka, berikut merupakan Gambar 4. 11 yang menampilkan code pada program yang digunakan beserta output nilai label//target yang diperoleh pada Gambar 4. 12 :



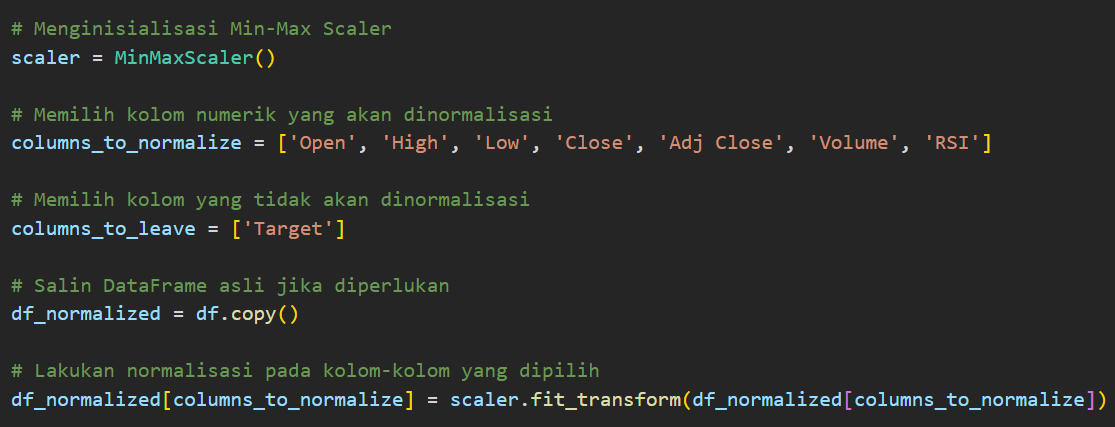
Gambar 4. 11 Code untuk menentukan target/label



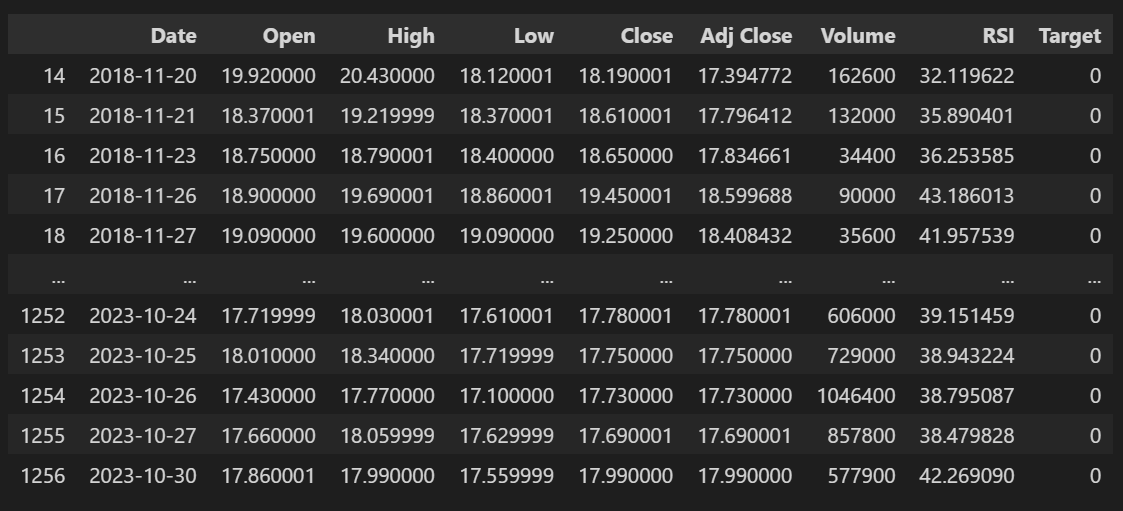
Gambar 4. 12 Output setelah menentukan nilai pada label/target

### **Normalisasi Data**

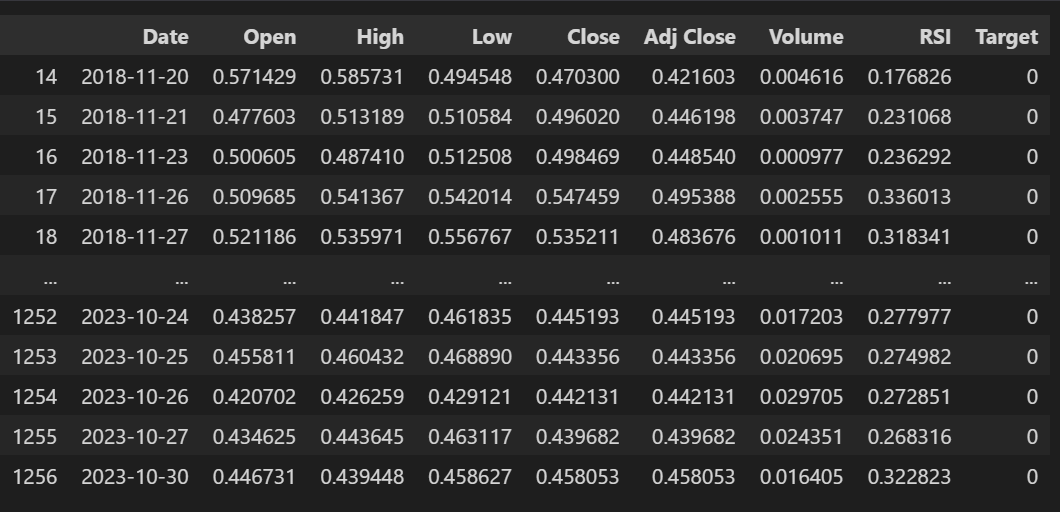
Pada tahap ini akan dilakukan normalisasi nilai dengan tujuan agar memiliki rentang skala yang sama pada tiap-tiap atribut sehingga tidak terjadi kesenjangan nilai antar tiap-tiap atribut yang akhirnya berpengaruh pada kinerja model. Oleh karena itu, pada tahap ini akan dilakukan normalisasi pada semua atribut data kecuali atribut target menggunakan metode *MinMaxScaler()* sehingga akan menghasilakan nilai dengan rentang 0 sampai 1. Berikut merupakan Gambar 4. 13 yang menampilkan code pada program yang digunakan, Gambar 4. 14 yang menampilkan output pada program sebelum nilai dilakukan normalisasi dan Gambar 4. 15 yang menampilkan output pada program setelah nilai dilakukan normalisasi:



Gambar 4. 13 Code untuk normalisasi nilai pada data



Gambar 4. 14 Output sebelum nilai pada data di normalisasi

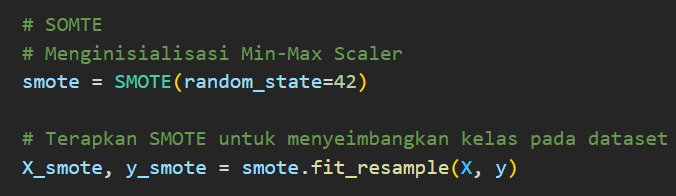


Gambar 4. 15 Output setelah nilai pada data di normalisasi

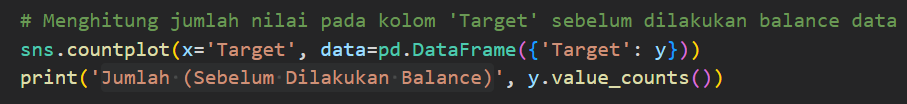
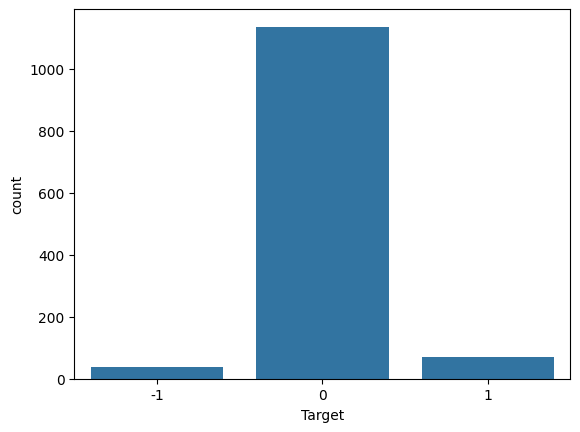
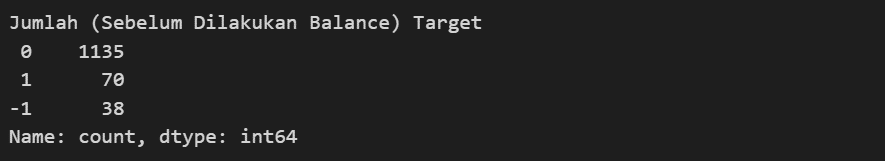
## **Balence Data**

Adanya data yang bersifat *imbalance* dapat memberikan dampak yang cukup buruk pada kinerja model. Hal ini mendorong model untuk cenderung lebih mudah dalam memperediksi kelas mayoritas dibandingkan kelas minoritas dikarenakan dominasi kelas mayoritas. Oleh karena itu, akurasi keseluruhan yang diperoleh tentu tidak mencerminkan kinerja yang sebenarnya pada semua kelas. Lantas, diperlukannya *belence* data agar kinerja model mampu secara akurat dan adil dalam memprediksi semua kelas.

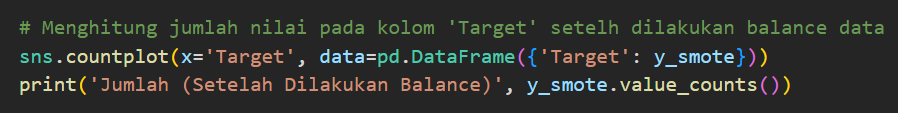
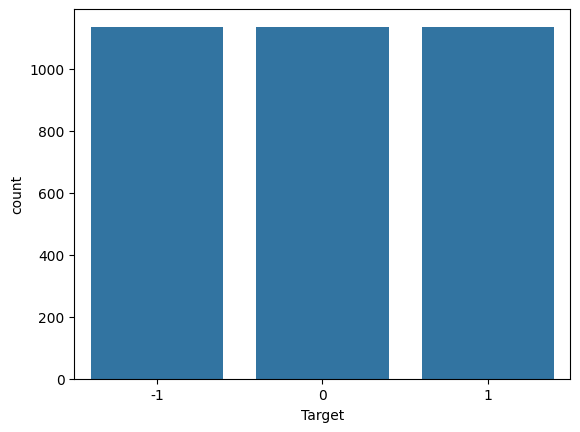
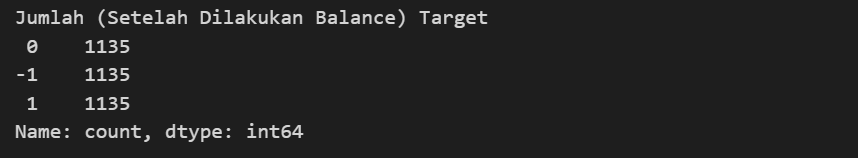
Pada tahap ini, *belence* data yang diterapkan yakni *oversampling* atau dalam artian meningkatkan jumlah kelas minoritas agar seimbang dengan kelas mayoritas. Metode yang digunakan pada *oversampling* yakni SMOTE (Synthetic Minority Over-sampling Technique) yang akan menghasilkan sampel-sampel sintetik untuk kelas minoritas berdasarkan interpolasi atau kombinasi sampel-sampel yang ada pada kelas minoritas. Berikut merupakan Gambar 4. 16 yang menampilkan code pada program yang digunakan untuk melakukan *balance* data, Gambar 4. 17 yang menampilkan code serta output pada program sebelum dilakukan belence data dan Gambar 4. 18 yang menampilkan code serta output pada program setelah dilakukan balence data :



Gambar 4. 16 Code untuk belance data

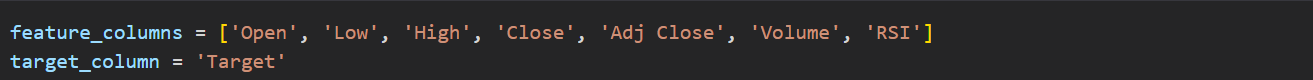
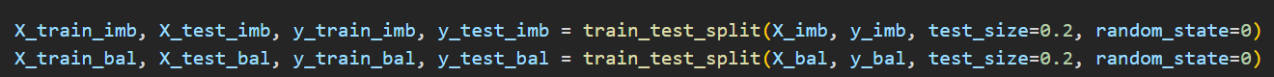
Gambar 4. 17 code serta output pada program sebelum dilakukan belence data

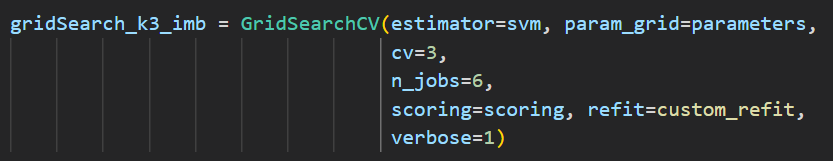
Gambar 4. 18 code serta output pada program setelah dilakukan belence data

## **Pembagian Data**

Pada tahap ini akan dilakukan terlebih dahulu pembagaian data berupa data *traning* dan data *testing* menggunakan *train\_test\_split* dengan persentasi yakni 80% untuk data *traning* dan 20% untuk data *testing* pada masing-masing dataset *imbalance* maupun *balance*. Kemudian setelah itu, data *traning* akan dilakukan pembagian data secara silang dengan nilai k-fold yakni 3, 5, 7, 9 dan 11 menggunakan *GridSearchCV* dengan parameter *cv* sama dengan nilai k-fold. Berikut merupakan Gambar 4. 19 yang menampilkan code *train\_test\_split* dan Gambar 4. 20 yang menampilkan code *GridSearchCV* pada program yang digunakan:

Gambar 4. 19 Pembagian data traning dan testing menggunakan train\_test\_split pada dataset imbalance dan balance



Gambar 4. 20 Data traning imbalance maupun balance akan dilakukan pembagian data secara silang berdasarkan nilai k-fold menggunakan GridSearchCV

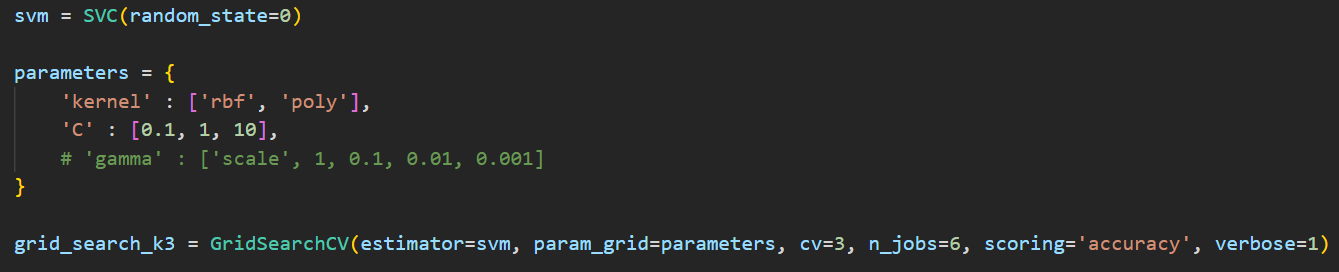
## **Modeling**

### **Pemodelan SVM (Support Vector Machine)**

Pemodelan pada tahap ini menggunakan SVM (Support Vector Machine) dengan kernel berupa kernel RBF dan kernel polynomial. Parameter yang digunakan ialah berupa nilai C yakni 1, 10, 100 dan 1000 serta nilai gamma yakni 1, 0.1, 0.01, 0.001, dan 0.0001 menggunakan pustaka dari library *scikit-learn*. Oleh karena itu, terlebih dahulu melakukan inisialisasi SVM (Support Vector Machine) dengan parameter berupa *random\_state=0* untuk mengatur *seed* dari generator acak agar hasil yang diperoleh tetap konsisten tiap kali dijalankan dan parameter *decision\_function\_shape='ovr'* untuk menentukan fungsi keputusan yang digunakan untuk menanggani masalah *multi class classifier*. Kemudian barulah, menyetel parameter yang digunakan seperti yang telah dijabarkan sebelumnya. Berikut merupakan Gambar 4. 21 yang menampilkan code inisialisasi model SVM (Support Vector Machine) dan Gambar 4. 22 yang menampilkan parameter model SVM (Support Vector Machine):



Gambar 4. 21 Code untuk inisialisasi model SVM (Support Vector Machine)



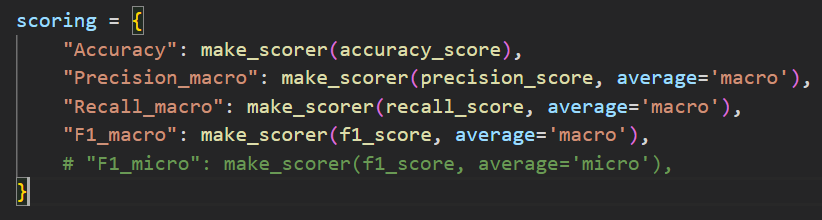
Gambar 4. 22 Code untuk parameter model SVM (Support Vector Machine)

### **Mengatur Parameter Nilai Pada GridSearchCV**

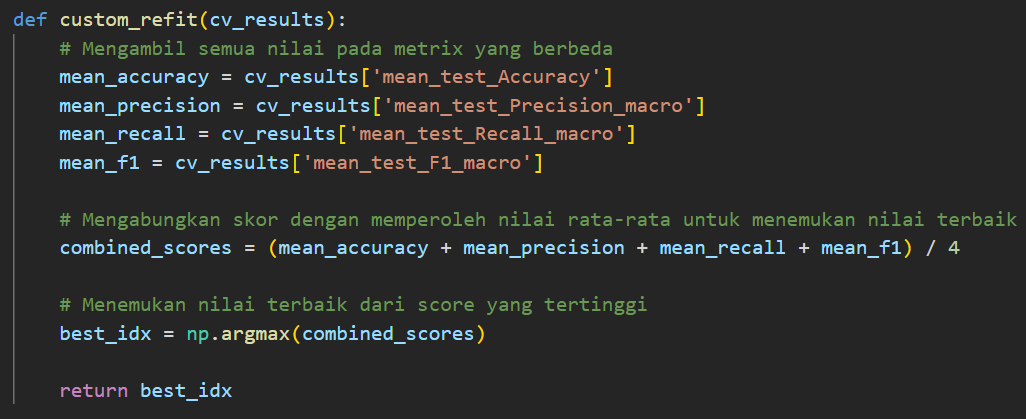
Sebelum melakukan proses pencarian parameter terbaik menggunakan *GridSearchCV*, terdapat beberapa parameter tambahan yang dibutuhkan agar pencarian tersebut dapat berjalan secara optimal. Parameter tambahan tersebut yakni:

1. Parameter *estimator* berfungsi untuk menentukan model yang akan digunakan. Dalam hal ini, model yang digunakan ialah SVM (Support Vector Machine) yang telah diinisialisasikan sebelumnya pada Gambar 4. 21
2. Parameter *param\_grid* merujuk pada daftar kumpulan nilai-nilai parameter yang dibungkus dalam bentuk *dictionary*, yang akan dicoba selama proses pencarian. Nilai dari *param\_grid* telah dideklarasikan sebelumnya pada variabel *paramters* yakni pada Gambar 4. 22
3. Parameter *cv* digunakan untuk menentukan jumalah lipatan atau nilai *fold* yang digunakan dalam proses *cross validation*. Nilai *fold* yang digunakan ialah 3, 5, 7 dan 9.
4. Parameter *n\_job* digunakan untuk mengatur jumlah pekerjaan paralel yang akan dijalankan selama proses pencarian grid. Misal, menerapkan *n\_job=6* untuk melakukan pemrosesan paralel menggunakan 6 inti CPU sehingga mempercepat proses pencarian.
5. Parameter *scorcing* digunakan untuk menilai peforma model menggunakan metrix evaluasi berdasarkan variabel *scorcing* yang telah didefinisikan sebelumnya seperti akurasi, presisi, recall dan F1-score. Lihat Gambar 4. 23 yang mendefinisikan nilai dari variabel *scorcing*.
6. Parameter *refit* digunakan untuk memilih model terbaik berdasarkan metrix evaluasi yang ditentukan. Misal, pemilihan model terbaik berdasarkan hasil akurasi tertinggi dari akurasi, presisi, recall dan F1-score yang telah didefinisikan pada *function* *custom\_refit*. Lihat Gambar 4. 24 yang mendefinisikan nilai dari *function* *custom\_refit*.
7. Parameter *verbose* digunakan untuk mengatur tingkat keluaran log selama proses pencarian sehingga informasi mengenai kemajuan pencarian akan ditampilkan dan dapat memantau secara jelas proses yang sedang berlangsung. Misal, menerapkan *verbose=1* untuk menampilkan informasi dasar tentang proses pencarian parameter.

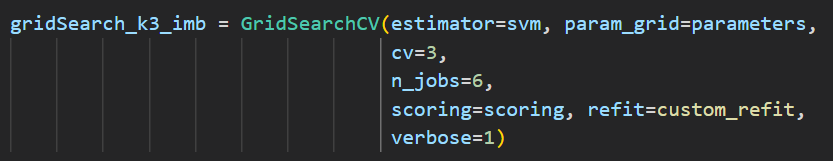
Dengan memahami serta menerapakan parameter tambahan diatas secara tepat, maka proses pencarian parameter terbaik menggunakan *GridSearchCV* bisa berjalan secara optimal serta menghasilkan model dengan performa terbaik sesuai dengan kebutuhan. Berikut merupakan Gambar 4. 25 yang menampilkan code secara utuh terkait dengan penggunaan *GridSearchCV*:



Gambar 4. 23 Code yang mendefinisikan nilai dari variabel scorcing



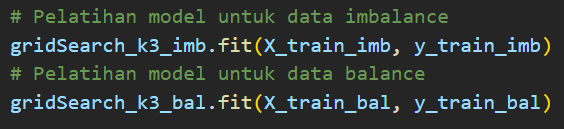
Gambar 4. 24 Code yang mendefinisikan nilai dari function custom\_refit



Gambar 4. 25 Code yang menerapkan GridSearchCV

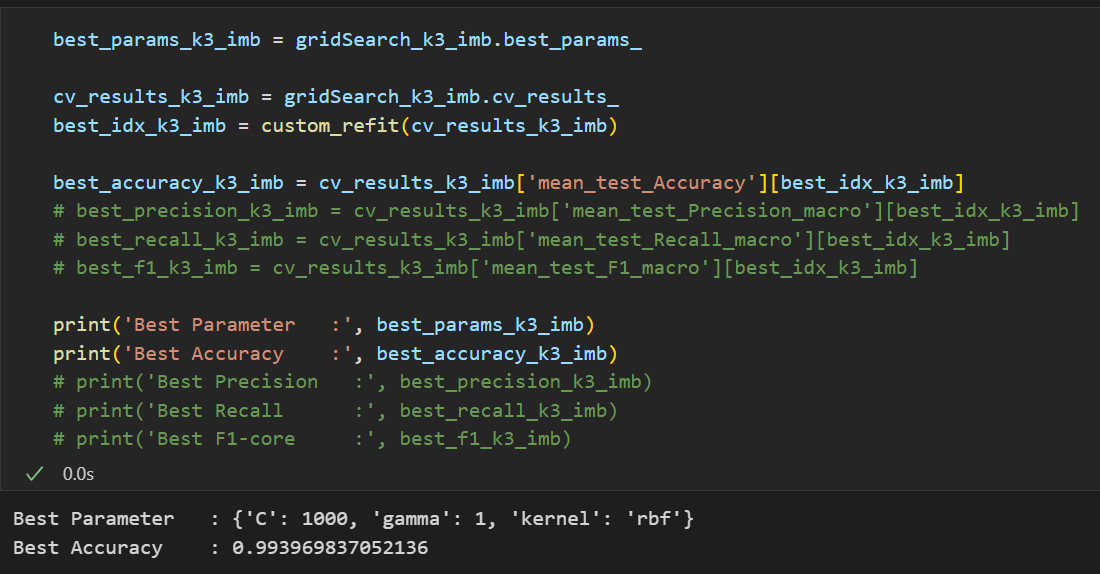
### **Pencarian Parameter Terbaik**

Setelah melakukan pengaturan parameter nilai pada *GridSearchCV* maka langkah selanjutnya ialah melakukan pencarian parameter terbaik menggunakan *GridSearchCV*. Oleh karena itu, terlebih dahulu perlu untuk melakukan pelatihan model baik pada data yang *imbalance* maupun *balance*. Pelatihan model tersebut menggunakan data *traning* dari masing-masing dataset baik pada data *imbalance* maupun data *balance* yang telah dibagi pada proses pembagian data sebelumnya. Berikut Gambar 4. 26 yang merupakan *code* yang digunakan dalam proses pelatihan model:



Gambar 4. 26 Code yang digunakan untuk melakukan pelatihan model

Setelah berhasil melakukan pelatihan model pada data *balance* maupun *imbalance*, maka barulah dapat diketahui hasil parameter model terbaik selama proses pelatihan untuk tiap-tiap fold yang diuji. Berikut merupakan Gambar 4. 27 yang menampilkan code beserta output dari hasil parameter terbaik serta akurasinya:

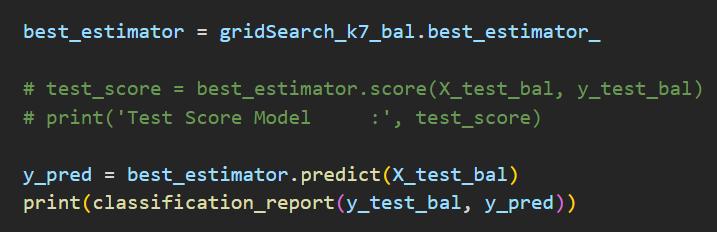


Gambar 4. 27 Code yang digunakan untuk menemukan hasil parameter terbaik serta akurasinya

Berdasarkan hasil yang telah diperoleh selama proses pelatihan model, maka akan dilakukan pemilihan parameter paling optimal diantara nilai *fold* yang diuji baik pada data *imbalance* maupun *balance*. Pemilihan parameter paling optimal ini bertujuan untuk mengetahui kinerja model yang terbaik sehingga dapat dipilih dan digunakan dalam melakukan prediksi model kedepannya. Oleh karena itu, parameter paling optimal akan digunakan untuk melakukan prediksi pada model.

### **Pengujian Menggunakan Paramter Terbaik**

Ketika telah menemukan parameter paling optimal, maka langkah selanjutnya ialah melakukan prediksi model atau pengujian model. Pegujian model ini menggunakan data *testing* yang telah dibagi pada proses pembagian data sebelumnya. Data *testing* tersebut merupakan data tunggal yang tidak digunakan atau tidak dilibatkan selama proses pelatihan model. Oleh karena itu, data *testing* bertujuan untuk mengevaluasi model untuk megetahui kinerja model yang telah dilatih sebelumnya. Berikut Gambar 4. 28 yang merupakan *code* yang digunakan untuk melakukan prediksi atau pengujian model:



Gambar 4. 28 Code untuk melakukan prediksi atau pengujian model

## **Hasil Dan Evaluasi**

Pada tahap ini, akan dilakukan pengujian menggunakan data yang diproses sebelumnya pada pergerakan harian harga saham Manchester United (MANU) pada bursa efek New York Stock Exchange (NYSC) dengan rentang waktu selama 5 tahun terkahir yakni pada 31 Oktober 2018 sampai 30 Oktober 2023. Pengujian ini dilakukan menggunakan cross validation dengan tiap-tiap nilai k-fold yakni 3, 5, 7 dan 9 serta tidak dilakukanya *balance* data dan dilakukanya *balance* data. Berikut merupakan hasil dari pengujian tersebut:

### **Hasil Pengujian Parameter Pada 3-Fold**

Tahap pertama yang perlu dilakukan ialah menemukan nilai terbaik dari tiap-tipa parameter kernel, nilai C dan nilai gamma. Berikut merupakan hasil dari tiap-tiap parameter kernel, nilai C dan nilai gamma pada Table 4. 1 untuk data *imbalance* dan Table 4. 2 untuk data *balance*:

Table 4. 1 Hasil pengujian kernel, nilai C dan nilai menggunakan 3-fold pada data imbalance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3-fold | | | | | | |
| Paramater | | | Data Imbalance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.937624 | 0.645275 | 0.497076 | 0.541382 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.984912 | 0.988387 | 0.900747 | 0.939069 |
| 0.1 | 0.937630 | 0.756396 | 0.501329 | 0.552498 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.990952 | 0.973430 | 0.955301 | 0.963537 |
| 0.1 | 0.987934 | 0.985900 | 0.922545 | 0.950993 |
| 0.01 | 0.938637 | 0.756717 | 0.507177 | 0.558685 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | **0.993970** | **0.981434** | **0.971619** | **0.975170** |
| 0.1 | 0.992966 | 0.987706 | 0.960291 | 0.973085 |
| 0.01 | 0.989942 | 0.986619 | 0.934241 | 0.958255 |
| 0.001 | 0.938637 | 0.756717 | 0.507177 | 0.558685 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| Poly | 1 | 1 | 0.952723 | 0.642578 | 0.590275 | 0.612050 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.958760 | 0.645588 | 0.625363 | 0.634221 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.979886 | 0.916373 | 0.911087 | 0.910049 |
| 0.1 | 0.929573 | 0.642712 | 0.450292 | 0.493419 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.982895 | 0.899031 | 0.977294 | 0.932673 |
| 0.1 | 0.952723 | 0.642578 | 0.590275 | 0.612050 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |

Table 4. 2 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 3-fold pada data balance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 3-fold | | | | | | |
| Paramater | | | Data Balance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.975404 | 0.976375 | 0.975268 | 0.975095 |
| 0.1 | 0.937225 | 0.942831 | 0.936877 | 0.935176 |
| 0.01 | 0.866006 | 0.885765 | 0.865339 | 0.855755 |
| 0.001 | 0.366373 | 0.214972 | 0.360650 | 0.224566 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 10 | 1 | 0.990822 | 0.990961 | 0.990772 | 0.990750 |
| 0.1 | 0.973568 | 0.974681 | 0.973422 | 0.973200 |
| 0.01 | 0.938326 | 0.943723 | 0.937984 | 0.936363 |
| 0.001 | 0.865639 | 0.885482 | 0.864970 | 0.855316 |
| 0.0001 | 0.366740 | 0.215876 | 0.361019 | 0.225145 |
| 100 | 1 | 0.997063 | 0.997071 | 0.997047 | 0.997047 |
| 0.1 | 0.991924 | 0.992087 | 0.991879 | 0.991860 |
| 0.01 | 0.974302 | 0.975352 | 0.974160 | 0.973948 |
| 0.001 | 0.937959 | 0.943429 | 0.937615 | 0.935967 |
| 0.0001 | 0.865639 | 0.885482 | 0.864970 | 0.855316 |
| 1000 | 1 | **0.997063** | **0.997079** | **0.997047** | **0.997050** |
| 0.1 | 0.997063 | 0.997076 | 0.997047 | 0.997047 |
| 0.01 | 0.991924 | 0.992052 | 0.991879 | 0.991860 |
| 0.001 | 0.973935 | 0.975005 | 0.973791 | 0.973572 |
| 0.0001 | 0.937959 | 0.943429 | 0.937615 | 0.935967 |
| Poly | 1 | 1 | 0.943098 | 0.949981 | 0.942783 | 0.942152 |
| 0.1 | 0.578928 | 0.687339 | 0.575683 | 0.528114 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 10 | 1 | 0.977974 | 0.979250 | 0.977852 | 0.977911 |
| 0.1 | 0.796623 | 0.860329 | 0.795213 | 0.789644 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 100 | 1 | 0.988620 | 0.989077 | 0.988557 | 0.988625 |
| 0.1 | 0.899046 | 0.915807 | 0.898512 | 0.894796 |
| 0.01 | 0.339207 | 0.113111 | 0.333333 | 0.168906 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 1000 | 1 | 0.993025 | 0.993236 | 0.992986 | 0.993038 |
| 0.1 | 0.943098 | 0.949981 | 0.942783 | 0.942152 |
| 0.01 | 0.578928 | 0.687339 | 0.575683 | 0.528114 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |

Berdasarkan hasil yang telah diperoleh pada tabel diatas, maka dapat diketahui bahwa parameter terbaik pada data yang *imbalance* ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nilai gamma sama dengan 1** dengan akurasi **99.40%,** presisi **98.14%,** recall **97.16%** dan F1-score **97.52%**. Sedangkan pada data yang *balance*, parameter terbaik ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nila gamma sama dengan 1** dengan akurasi **99.71%,** presisi **99.71%,** recall **99.70%**dan F1-score **99.71%**. Oleh karena itu, jika dibandingkan hasil yang diperoleh maka dapat disimpulkan bahwa parameter terbaik ialah pada data yang dilakukan *balance*.

### **Hasil Pengujian Parameter Pada 5-Fold**

Tahap pertama yang perlu dilakukan ialah menemukan nilai terbaik dari tiap-tipa parameter kernel, nilai C dan nilai gamma. Berikut merupakan hasil dari tiap-tiap parameter kernel, nilai C dan nilai gamma pada Table 4. 3 untuk data *imbalance* dan Table 4. 4 untuk data *balance*:

Table 4. 3 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 5-fold pada data imbalance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5-fold | | | | | | |
| Paramater | | | Data Imbalance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.946683 | 0.648219 | 0.550000 | 0.584771 |
| 0.1 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.984920 | 0.979709 | 0.910519 | 0.939010 |
| 0.1 | 0.942653 | 0.713580 | 0.530808 | 0.579120 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.990950 | 0.977171 | 0.954523 | 0.964467 |
| 0.1 | 0.991955 | 0.997091 | 0.946104 | 0.968914 |
| 0.01 | 0.940627 | 0.712951 | 0.518182 | 0.564101 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.992960 | 0.976311 | 0.975158 | 0.974183 |
| 0.1 | **0.994975** | **0.989017** | **0.972431** | **0.979995** |
| 0.01 | 0.989945 | 0.996382 | 0.928932 | 0.958633 |
| 0.001 | 0.940627 | 0.712951 | 0.518182 | 0.564101 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| Poly | 1 | 1 | 0.952713 | 0.650204 | 0.584343 | 0.610413 |
| 0.1 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.959758 | 0.646202 | 0.630945 | 0.637775 |
| 0.1 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.986919 | 0.932287 | 0.978634 | 0.948593 |
| 0.1 | 0.930572 | 0.643069 | 0.456061 | 0.490197 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.982894 | 0.907886 | 0.982853 | 0.938045 |
| 0.1 | 0.952713 | 0.650204 | 0.584343 | 0.610413 |
| 0.01 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909456 | 0.303152 | 0.333333 | 0.317527 |

Table 4. 4 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 5-fold pada data balance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 5-fold | | | | | | |
| Paramater | | | Data Balance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.977242 | 0.978114 | 0.977107 | 0.976959 |
| 0.1 | 0.943471 | 0.948143 | 0.943149 | 0.941814 |
| 0.01 | 0.870783 | 0.890034 | 0.870125 | 0.861227 |
| 0.001 | 0.387667 | 0.282188 | 0.382056 | 0.263480 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 10 | 1 | 0.991192 | 0.991380 | 0.991144 | 0.991129 |
| 0.1 | 0.975775 | 0.976786 | 0.975625 | 0.975446 |
| 0.01 | 0.943104 | 0.947823 | 0.942777 | 0.941425 |
| 0.001 | 0.870783 | 0.890034 | 0.870125 | 0.861227 |
| 0.0001 | 0.387667 | 0.282188 | 0.382056 | 0.263480 |
| 100 | 1 | 0.997064 | 0.997079 | 0.997043 | 0.997045 |
| 0.1 | 0.991559 | 0.991747 | 0.991510 | 0.991508 |
| 0.01 | 0.976877 | 0.977818 | 0.976732 | 0.976573 |
| 0.001 | 0.942737 | 0.947520 | 0.942408 | 0.941037 |
| 0.0001 | 0.870783 | 0.890034 | 0.870125 | 0.861227 |
| 1000 | 1 | 0.997064 | 0.997093 | 0.997045 | 0.997048 |
| 0.1 | **0.997431** | **0.997443** | **0.997414** | **0.997415** |
| 0.01 | 0.992293 | 0.992434 | 0.992245 | 0.992240 |
| 0.001 | 0.976877 | 0.977827 | 0.976734 | 0.976582 |
| 0.0001 | 0.942737 | 0.947520 | 0.942408 | 0.941037 |
| Poly | 1 | 1 | 0.946403 | 0.952675 | 0.946131 | 0.945535 |
| 0.1 | 0.611598 | 0.734603 | 0.608655 | 0.569811 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 10 | 1 | 0.980548 | 0.981609 | 0.980426 | 0.980487 |
| 0.1 | 0.813512 | 0.864073 | 0.812282 | 0.805974 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 100 | 1 | 0.988990 | 0.989390 | 0.988922 | 0.988972 |
| 0.1 | 0.902719 | 0.919238 | 0.902192 | 0.898689 |
| 0.01 | 0.339207 | 0.113152 | 0.333333 | 0.168952 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 1000 | 1 | 0.993027 | 0.993231 | 0.992981 | 0.993027 |
| 0.1 | 0.946403 | 0.952675 | 0.946131 | 0.945535 |
| 0.01 | 0.611598 | 0.734603 | 0.608655 | 0.569811 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168859 |

Berdasarkan hasil yang telah diperoleh pada tabel diatas, maka dapat diketahui bahwa parameter terbaik pada data yang *imbalance* ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nilai gamma sama dengan 0.1** dengan akurasi **99.50%,** presisi **98.90%,** recall **97.24%** dan F1-score **98.00%**. Sedangkan pada data yang *balance*, parameter terbaik ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nila gamma sama dengan 0.1** dengan akurasi **99.74%,** presisi **99.74%,** recall **99.74%** dan F1-score **99.74%**. Oleh karena itu, jika dibandingkan hasil yang diperoleh maka dapat disimpulkan bahwa parameter terbaik ialah pada data yang dilakukan *balance*.

### **Hasil Pengujian Parameter Pada 7-Fold**

Tahap pertama yang perlu dilakukan ialah menemukan nilai terbaik dari tiap-tipa parameter kernel, nilai C dan nilai gamma. Berikut merupakan hasil dari tiap-tiap parameter kernel, nilai C dan nilai gamma pada Table 4. 5 untuk data *imbalance* dan Table 4. 6 untuk data *balance*:

Table 4. 5 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 7-fold pada data imbalance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 7-fold | | | | | | |
| Paramater | | | Data Imbalance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.946680 | 0.695842 | 0.552513 | 0.593998 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.986922 | 0.981206 | 0.926646 | 0.946299 |
| 0.1 | 0.943662 | 0.694869 | 0.535317 | 0.578796 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.991952 | 0.992153 | 0.950821 | 0.968822 |
| 0.1 | 0.989940 | 0.996386 | 0.933333 | 0.959023 |
| 0.01 | 0.941650 | 0.694215 | 0.524735 | 0.569658 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.993964 | 0.981080 | 0.971884 | 0.974011 |
| 0.1 | **0.995976** | **0.998546** | **0.976190** | **0.986117** |
| 0.01 | 0.992958 | 0.997466 | 0.951190 | 0.970803 |
| 0.001 | 0.940644 | 0.693890 | 0.518783 | 0.564601 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| Poly | 1 | 1 | 0.953722 | 0.650541 | 0.590608 | 0.613690 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.959759 | 0.646073 | 0.630586 | 0.637225 |
| 0.1 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.986922 | 0.939114 | 0.972868 | 0.949004 |
| 0.1 | 0.932596 | 0.643690 | 0.468915 | 0.507820 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.983903 | 0.926273 | 0.966178 | 0.938493 |
| 0.1 | 0.953722 | 0.650541 | 0.590608 | 0.613690 |
| 0.01 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909457 | 0.303152 | 0.333333 | 0.317527 |

Table 4. 6 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 7-fold pada data balance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 7-fold | | | | | | |
| Paramater | | | Data Balance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.978341 | 0.979128 | 0.978221 | 0.978097 |
| 0.1 | 0.943472 | 0.948328 | 0.943152 | 0.941862 |
| 0.01 | 0.872621 | 0.891529 | 0.871967 | 0.863458 |
| 0.001 | 0.411899 | 0.475842 | 0.406469 | 0.299546 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 10 | 1 | 0.991559 | 0.991723 | 0.991510 | 0.991497 |
| 0.1 | 0.977609 | 0.978494 | 0.977482 | 0.977348 |
| 0.01 | 0.943105 | 0.947994 | 0.942783 | 0.941466 |
| 0.001 | 0.872621 | 0.891529 | 0.871967 | 0.863458 |
| 0.0001 | 0.412266 | 0.475949 | 0.406841 | 0.300359 |
| 100 | 1 | 0.997064 | 0.997086 | 0.997047 | 0.997046 |
| 0.1 | 0.993026 | 0.993145 | 0.992986 | 0.992983 |
| 0.01 | 0.977976 | 0.978836 | 0.977852 | 0.977716 |
| 0.001 | 0.942738 | 0.947689 | 0.942414 | 0.941081 |
| 0.0001 | 0.872621 | 0.891529 | 0.871967 | 0.863458 |
| 1000 | 1 | 0.997798 | 0.997824 | 0.997785 | 0.997788 |
| 0.1 | **0.998166** | **0.998193** | **0.998154** | **0.998161** |
| 0.01 | 0.992658 | 0.992776 | 0.992617 | 0.992607 |
| 0.001 | 0.977609 | 0.978509 | 0.977482 | 0.977343 |
| 0.0001 | 0.942738 | 0.947689 | 0.942414 | 0.941081 |
| Poly | 1 | 1 | 0.948247 | 0.954225 | 0.947951 | 0.947500 |
| 0.1 | 0.625184 | 0.770769 | 0.622358 | 0.585156 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 10 | 1 | 0.981646 | 0.982683 | 0.981543 | 0.981618 |
| 0.1 | 0.820487 | 0.864027 | 0.819292 | 0.813255 |
| 0.01 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 100 | 1 | 0.988988 | 0.989408 | 0.988926 | 0.988981 |
| 0.1 | 0.904559 | 0.920607 | 0.904024 | 0.900757 |
| 0.01 | 0.339942 | 0.208515 | 0.334077 | 0.170568 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 1000 | 1 | 0.993762 | 0.993946 | 0.993725 | 0.993767 |
| 0.1 | 0.948247 | 0.954225 | 0.947951 | 0.947500 |
| 0.01 | 0.625184 | 0.770769 | 0.622358 | 0.585156 |
| 0.001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |
| 0.0001 | 0.339207 | 0.113069 | 0.333333 | 0.168860 |

Berdasarkan hasil yang telah diperoleh pada tabel diatas, maka dapat diketahui bahwa parameter terbaik pada data yang *imbalance* ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nilai gamma sama dengan 0.1** dengan akurasi **99.60%,** presisi **99.85%,** recall **97.62%** dan F1-score **98.61%**. Sedangkan pada data yang *balance*, parameter terbaik ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nila gamma sama dengan 0.1** dengan akurasi **99.82%**,presisi **99.82%**,recall **99.82%** dan F1-score **99.82%**. Oleh karena itu, jika dibandingkan hasil yang diperoleh maka dapat disimpulkan bahwa parameter terbaik ialah pada data yang dilakukan *balance*.

### **Hasil Pengujian Parameter Pada 9-Fold**

Tahap pertama yang perlu dilakukan ialah menemukan nilai terbaik dari tiap-tipa parameter kernel, nilai C dan nilai gamma. Berikut merupakan hasil dari tiap-tiap parameter kernel, nilai C dan nilai gamma pada Table 4. 7 untuk data *imbalance* dan Table 4. 8 untuk data *balance*:

Table 4. 7 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 9-fold pada data imbalance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 9-fold | | | | | | |
| Paramater | | | Data Imbalance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.947693 | 0.685617 | 0.558642 | 0.595140 |
| 0.1 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.987933 | 0.982796 | 0.926953 | 0.947253 |
| 0.1 | 0.943680 | 0.721350 | 0.539683 | 0.587757 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.991956 | 0.992167 | 0.948924 | 0.966515 |
| 0.1 | 0.990964 | 0.996750 | 0.937831 | 0.962498 |
| 0.01 | 0.943671 | 0.721359 | 0.538801 | 0.584536 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.993967 | 0.981877 | 0.972371 | 0.973944 |
| 0.1 | **0.995987** | **0.998544** | **0.977072** | **0.986836** |
| 0.01 | 0.992975 | 0.997465 | 0.950176 | 0.970269 |
| 0.001 | 0.944681 | 0.721674 | 0.544974 | 0.592643 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| Poly | 1 | 1 | 0.953726 | 0.650543 | 0.591711 | 0.614362 |
| 0.1 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 10 | 1 | 0.959760 | 0.645472 | 0.631027 | 0.636916 |
| 0.1 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 100 | 1 | 0.989944 | 0.951549 | 0.979782 | 0.959797 |
| 0.1 | 0.932596 | 0.643709 | 0.468254 | 0.504351 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 1000 | 1 | 0.986923 | 0.933658 | 0.984480 | 0.951373 |
| 0.1 | 0.953726 | 0.650543 | 0.591711 | 0.614362 |
| 0.01 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |
| 0.0001 | 0.909455 | 0.303152 | 0.333333 | 0.317527 |

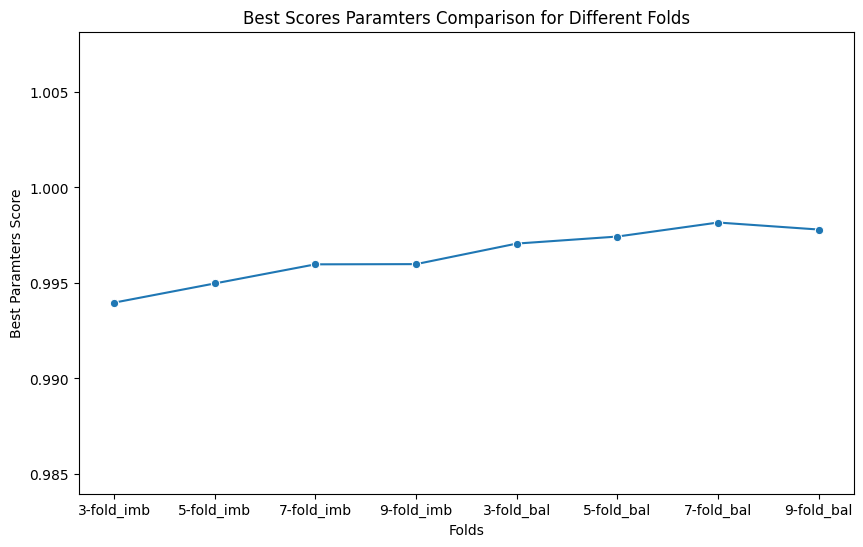
Table 4. 8 Hasil pengujian kernel, nilai C dan nilai γ menggunakan 9-fold pada data balance

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 9-fold | | | | | | |
| Paramater | | | Data Balance | | | |
| Kernel | C |  | Akurasi (%) | Presisi (%) | Recall (%) | F1 (%) |
| RBF | 1 | 1 | 0.977976 | 0.978863 | 0.977844 | 0.977737 |
| 0.1 | 0.943836 | 0.948651 | 0.943509 | 0.942199 |
| 0.01 | 0.872618 | 0.891925 | 0.871950 | 0.863230 |
| 0.001 | 0.446773 | 0.547536 | 0.441752 | 0.357307 |
| 0.0001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 10 | 1 | 0.992661 | 0.992814 | 0.992607 | 0.992608 |
| 0.1 | 0.978346 | 0.979243 | 0.978203 | 0.978098 |
| 0.01 | 0.943835 | 0.948638 | 0.943513 | 0.942197 |
| 0.001 | 0.872983 | 0.892224 | 0.872324 | 0.863658 |
| 0.0001 | 0.447508 | 0.548179 | 0.442496 | 0.358330 |
| 100 | 1 | 0.997064 | 0.997080 | 0.997044 | 0.997046 |
| 0.1 | 0.993394 | 0.993540 | 0.993348 | 0.993351 |
| 0.01 | 0.978714 | 0.979560 | 0.978570 | 0.978459 |
| 0.001 | 0.944570 | 0.949257 | 0.944250 | 0.942987 |
| 0.0001 | 0.872983 | 0.892224 | 0.872324 | 0.863658 |
| 1000 | 1 | 0.997065 | 0.997102 | 0.997041 | 0.997047 |
| 0.1 | **0.997799** | **0.997817** | **0.997781** | **0.997788** |
| 0.01 | 0.992661 | 0.992793 | 0.992607 | 0.992605 |
| 0.001 | 0.978714 | 0.979560 | 0.978570 | 0.978459 |
| 0.0001 | 0.944570 | 0.949257 | 0.944250 | 0.942987 |
| Poly | 1 | 1 | 0.948607 | 0.954556 | 0.948313 | 0.947858 |
| 0.1 | 0.631050 | 0.782400 | 0.628358 | 0.592201 |
| 0.01 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.0001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 10 | 1 | 0.981651 | 0.982732 | 0.981522 | 0.981611 |
| 0.1 | 0.823057 | 0.865217 | 0.821920 | 0.815822 |
| 0.01 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.0001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 100 | 1 | 0.988990 | 0.989454 | 0.988915 | 0.988980 |
| 0.1 | 0.906389 | 0.921961 | 0.905864 | 0.902694 |
| 0.01 | 0.341040 | 0.243032 | 0.335189 | 0.172883 |
| 0.001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.0001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 1000 | 1 | 0.993761 | 0.993923 | 0.993718 | 0.993761 |
| 0.1 | 0.948607 | 0.954556 | 0.948313 | 0.947858 |
| 0.01 | 0.631050 | 0.782400 | 0.628358 | 0.592201 |
| 0.001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |
| 0.0001 | 0.339205 | 0.113068 | 0.333333 | 0.168859 |

Berdasarkan hasil yang telah diperoleh pada tabel diatas, maka dapat diketahui bahwa parameter terbaik pada data yang *imbalance* ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nilai gamma sama dengan 0.1** dengan akurasi **99.60%,** presisi **99.85%,** recall **97.71%** dan F1-score **98.68%**. Sedangkan pada data yang *balance*, parameter terbaik ialah **kernel RBF** dengan **nilai C sama dengan 1000 dan nila gamma sama dengan 0.1** dengan akurasi **99.78%,** presisi **99.78%,** recall **99.78%** dan F1-score **99.78%**. Oleh karena itu, jika dibandingkan hasil yang diperoleh maka dapat disimpulkan bahwa parameter terbaik ialah pada data yang dilakukan *balance*.

### **Akurasi Parameter Terbaik Dari K-Fold Yang Diuji**

Berdasarkan hasil yang diperoleh, dapat diketahui bahwa nilai parameter terbaik terdapat pada data yang dilakukan *balance* dibandingkan dengan data *imbalance* untuk semua fold yang diuji. Hal bisa dilihat dengan adanya peningkatan nilai akurasi, presisi, recall dan F1-score yang diperoleh dari masing-masing fold yang diuji. Berikut Gambar 4. 29 yang menujukan perbandingan hasil parameter terbaik untuk tiap-tiap fold yang diuji pada data *imbalance* maupun data *balance*.



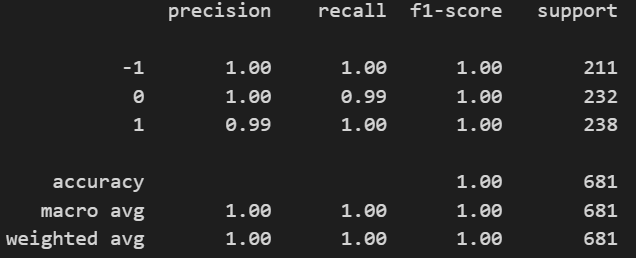
Gambar 4. 29 Perbandingan hasil parameter terbaik untuk tiap-tiap fold yang diuji pada data imbalance dan data balance

Berdasarkan grafik perbandingan hasil parameter terbaik untuk tiap-tiap fold yang diuji pada data *imbalance* dan data *balance* diatas, dapat diketahui bahwa parameter pada **7-fold dengan data yang dilakukan *balance*** merupakan parameter paling optimal. Oleh karena itu, parameter tersebut yaitu **kernel RBF** dengan **nilai C sama dengan 1000** dan **nila gamma sama dengan 0.1** akan digunakan untuk melakukan prediksi model atau pengujian model.

### **Hasil Pengujian Model Menggunakan Parameter Terbaik**

Setelah menemukan kombinasi parameter terbaik selama proses pelatihan, maka kemudian akan dilakukan prediksi model atau pengujian model. Pengujian model tersebut menggunakan data testing yang telah dilakukan pada proses pembagian data sebelumnya. Data testing tersebut merupakan data tunggal yang tidak digunakan atau tidak dilibatkan selama proses pelatihan model. Oleh karena itu, data testing ini berfungsi untuk mengevaluasi kinerja model yang telah dilatih sehingga bisa memastikan bahwa model yang dihasilkan mampu memberikan performa yang baik saat dihadapkan pada data baru dan tidak pernah dilihat sebelumnya.

Proses pengujian model tersebut menggunakan 7-fold dengan dilakukan data *belance* menggunakan parameter berupa kernel RBF sertanilai C sama dengan 1000 dan nila gamma sama dengan 0.1 menghasilkan akurasi **99.70%,** presisi **100%,** recall **100%,** dan F1-score **100%**. Berikut merupakan Gambar 4. 30 yang menampilkan hasil *performance metrics* dari prediksi model:



Gambar 4. 30 Performance metrics dari prediksi model

Dengan demikian, pemilihan dan penggunaan parameter yang optimal mampu memberikan prediksi yang akurat pada model. Hal ini menujukan bahwa pendekatan yang digunakan dapat menghasilkan performa terbaik yang tercermin melalui hasil akurasi, *precesion*, *recall* dan *F1-score* pada hasil prediksi.

# **KESIMPULAN DAN SARAN**

## **Kesimpulan**

Berdasarkan hasil pengujian yang telah dilakukan, maka dapat disimpulkan bahwa:

1. Metode SVM (Support Vector Machine) dapat digunakan untuk memprediksi harga saham yang akurat melalui pemrosesan data yang tepat serta pemilihan parameter yang optimal. Oleh karena itu, penggunaan kernel RBF mampu menjawab permasalahan yang kompleks dari data yang bersifat non-linier serta parameter nilai C sama dengan 1000 dan nilai gamma sama dengan 0.1 yang diperoleh dari proses validasi silang yakni 7-fold sebagai kombinasi yang optimal serta proses *balance* data agar model yang dihasilakn tidak bias terhadap kelas tertentu sehingga cenderung membuat model lebih prediktif dan akurat. Dengan demikian, memungkinkan model SVM (Support Vector Machine) mencapai hasil yang paling optimal dalam memprediksi harga saham.
2. Hasil penelitian menujukan bahwa model SVM (Support Vector Machine) menghasilkan parameter optimal pada 7-fold dengan data yang dilakukan *balance* menggunakan kernel RBF dengannilai C sama dengan 1000 dan nila gamma sama dengan 0.1. Parameter model tersebut menghasilakn akurasi sebesar **99.82%**,presisi sebesar **99.82%**,recall sebesarrecall **99.82%** dan F1-score sebesar **99.82%** sehingga parameter model tersebut akan digunakan untuk melakukan prediksi model atau pengujian model. Kemudian, pada tahap pengujian model atau prediksi model menggunakan data *testing*, diperoleh akurasi sebesar **99.70%,** presisi sebesar **100%,** recall sebesar **100%,** dan F1-score sebesar **100%**. Oleh karena itu, tingkat akurasi yang tinggi pada pengujian model menunjukan bahwa model dapat berkerja dengan baik dan dapat diandalkan untuk melakukan prediksi yang mendekati realitas pasar. Dengan demikian, hal ini sangat membantu bagi para investor dalam menentukan pilihan dan pengambilan keputusan serta menjawab permasalahan pada bagian latar belakang melalui solusi prediktif yang akurat dan dapat diandalkan pada model.

## **Saran**

Terdapat beberapa saran yang dapat penulis berikan untuk penelitian selanjutnya yaitu sebagai berikut:

1. Pertimbangkan untuk menggunakan data selain histori pergerakan harga harian saham Manchester United (MANU) dengan melihat beberapa informasi terkait dengan sektor saham pada industri yang sama atau berbeda. Tidak hanya itu, rentang waktu yang digunakan juga dapat diperluas sehingga dapat memberikan gambaran yang lebih jelas atau nyata terkait dengan kondisi pergerakan pasar saham.
2. Eksplorasi berbagai metode atau algoritma selain Support Vector Machine (SVM) untuk meningkatkan atau memperoleh akurasi yang lebih baik lagi. Bandingkan dan variasikan pendekatan untuk mengetaui sejauh mana model berkerja dan sesuai dengan karakteristik data.
3. Pertimbangkan untuk menerapkan metode *balance* data selain Synthetic Minority Over-sampling Technique (SMOTE) untuk memperoleh akurasi yang lebih baik lagi dalam melakukan prediksi. Pendekatan seperti teknik *oversampling* dan *undersampling* bisa menjadi alternatif lain dalam mengatasi data yang *imbalance* sehingga model yang dihasilkan lebih prediktif. Bandingkan berbagai pendekatan tersebut untuk menemukan solusi paling optimal.
4. Menerapkan indikator teknikal yang beragam selain hanya Relative Strength Index (RSI) bahkan mempertimbangkan untuk menggabungkan beberapa indikator teknikal seperti Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Moving Average (MA), Stochastic Oscillator, Bollinger Bands, Fibonacci Retracement dan sebagainya. Pendekatan ini dapat memperkaya atribut data dan meningkatkan pemahaman terhadap dinamika pasar. Selain itu, tidak ada salahnya untuk mempertimbangkan penggunaan indikator fundamental sebagai tambahan dalam analisis.

# **DAFTAR PUSTAKA**

Dr. Mamduh M. Hanafi, M. (2016). *Manajemen Keuangan.* Yogyakarta: BPFE-YOGYAKARTA.

Fadilah, W. R., Agfiannisa, D., & Azhar, Y. (2020). Analisis Prediksi Harga Saham PT. Telekomunikasi Indonesia Menggunakan Metode Support Vector Machine. *Fountain of Informatics Journal, 5*.

Hidayahtullah, F. S. (2022). *Teknik Rahasia Analisis Teknikal Saham.* Jakarta: PT Elex Media Komputindo.

Lestari, M., & Pangaribuan, J. J. (2020). Perbandingan Metode Moving Average (MA) Dan Neural Network Yang Berbasis Algoritma Backpropagation Dalam Prediksi Harga Saham. *Information System Development, 5*.

Lestari, S. S., & Jasuni, A. Y. (2023). Analisis Rantai Makrov Lima Status Pada Return Harga Saham BBCA. *Jurnal Bisnisman; Riset Bisnis dan Manajemen, 5*.

Manchester United. (n.d.). *Manchester United plc*. (Yahoo Finance) Retrieved 11 25, 2023, from https://finance.yahoo.com/quote/MANU/profile?p=MANU

Maulana, R., & Kumalasari, D. (2019). Analisis Dan Perbandingan Algoritma Data Mining Dalam Prediksi Harga Saham GGRM. *Jurnal Informatika Kaputama (JIK), 3*.

Miraltamirus, I., Fitri, F., Vionanda, D., & Permana, D. (2023). Stock Price Prediction of PT Bank Syariah Indonesia Tbk Using Support Vector Regression. *UNP JOURNAL OF STATISTICS AND DATA SCIENCE, 1*, 112-119.

Monika, N. E., & Yusniar, M. W. (2020). Analisis Teknikal Menggunakan Indikator MACD dan RSI Pada Saham JII. *Jurnal Riset Inspirasi Manajemen dan Kewirausahaan, 4*.

Rusmalawati, V., Furqon, M. T., & Indriati. (2018). Peramalan Harga Saham Menggunakan Metode Support Vector Regression (SVR) Dengan Particle Swarm Optimization (PSO). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 2*, 1980-1990.

Shen, J., Wu, J., Xu, M., Gan, D., An, B., & Liu, F. (2021). A Hybrid Method to Predict Postoperative Survival of Lung Cancer Using Improved SMOTE and Adaptive SVM. *Computational and Mathematical Methods in Medicine*, 15.

Syahputra, J., Ramadhan, R. D., & Burha, A. (2022). Prediksi Harga Saham Bank BRI Menggunakan Algoritma Linier Regression Sebagai Strategi Jual Beli Saham. *Journal of Dinda, 2*.

Wilder, J. W. (1978). *New Concepts in Technical Trading Systems.* Trend Research: Hunter Publishing Company.

Wulandari, R. F., & Anubhakti, D. (2022). Implementasi Algoritma Support Vector Machine (SVM) Dalam Memprediksi Harga Saham PT. Garuda Indonesia Tbk. *Indonesia Journal Information System (IDEALIS), 4*.

Zhao, Z., & Bai, T. (2022). Financial Fraud Detection and Prediction in Listed Companies Using SMOTE and Machine Learning Algorithms. *Entropy*, 1157.