

Sales Forecasting Menggunakan Random Forest Regression Dengan Particle Swarm Optimization Pada Penjualan Superstore

Gede Gery Sastrawan^{a1}, I Putu Gede Hendra Suputra^{a2}, Ida Bagus Gede Dwidasmara^{a3}, Ida Ayu Gde Suwiprabayanti Putra^{a4}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana

Kuta Selatan, Badung, Bali, Indonesia

¹gerysastrawan123@gmail.com

²hendra.suputra@unud.ac.id

³dwidasmara@unud.ac.id

⁴iagsuwiprabayantiputra@unud.ac.id

Abstract

In the current digital era, the ever-increasing volume of data highlights the significance of Big Data, encompassing data with large scale, variety, and complexity that poses challenges in storage, analysis, and visualization. Accurate sales forecasting, crucial in a competitive and dynamic business environment, provides critical insights for companies across various sectors. Tree-based machine learning algorithms, such as Random Forest and Gradient Boosting, are popularly employed for this purpose. Method optimization becomes essential to enhance the quality of results and relevance to the utilized data. Particle Swarm Optimization (PSO) is one technique that can be employed for this purpose. The Random Forest method is susceptible to overfitting, posing a major challenge in its usage. This research evaluates the performance of the Random Forest Regression algorithm optimized using Particle Swarm Optimization (PSO) for Superstore sales forecasting, compared to Grid Search and Randomized Search. The PSO-optimized model achieved an error value of 187.68 on the entire training data and 254.32 on the entire testing data. Grid Search has an error value that is superior to PSO but is not too significant. Apart from that, PSO has advantages compared to other optimization algorithms in that it has the shortest optimization time, namely 40 minutes 42 seconds, compared to Grid Search which takes up to 710 minutes 31.5 seconds and PSO allows users to easily tune the hyperparameters of the optimization algorithm that allows users to get better optimization results. Black-box testing demonstrates that the system functions effectively according to user requirements.

Keywords: Sales Forecasting, Random Forest Regression, Particle Swarm Optimization, Grid Search, Randomized Search, Overfitting, Black Box Testing.

1. Pendahuluan

Seiring berjalannya waktu, jumlah data yang dikumpulkan oleh sebuah sistem tentunya akan bertambah dan terus bertambah. Data merupakan sebuah hal yang sangat penting di jaman sekarang. Menurut Forbes, 59% dari bisnis yang ada menggunakan analitik data dengan cara yang berbeda-beda untuk mengambil keputusan yang lebih baik dan meningkatkan performa bisnis mereka. Big Data merupakan sebuah istilah untuk kumpulan data yang besar sekali yang memiliki besar, variasi, dan struktur yang kompleks dengan kesulitan untuk menyimpan, menganalisis, dan memvisualisasikan untuk proses lebih lanjut atau hasilnya[1]. Big data merupakan sebuah kombinasi dari data terstruktur, semiterstruktur, tidak terstruktur yang dikumpulkan oleh perusahaan yang dapat digali sebagai

informasi dan digunakan pada proyek machine learning, predictive modeling, dan aplikasi analitik lainnya. Big Data sering dikarakteristikan oleh tiga V yaitu: volume, variety, dan velocity[2]. Dengan adanya banyak karakteristik tersebut, terdapat banyak tantangan dalam memproses suatu Big Data untuk memperoleh wawasan dari Big Data tersebut. Tantangannya meliputi tantangan data, tantangan pemrosesan, dan tantangan manajemen data[3]. Salah satu bidang yang memanfaatkan big data ialah sektor penjualan. Dengan adanya data penjualan yang dikumpulkan oleh perusahaan, perusahaan dapat mengolah data tersebut dan mendapat wawasan yang dapat kita gunakan untuk membantu perusahaan dalam pengambilan keputusan.

Pada lingkungan bisnis saat ini yang sangat kompetitif dan secara konstan berubah, perkiraan penjualan yang akurat, yang disebut juga sebagai sales prediction atau sales forecasting, dan menawarkan pengetahuan yang penting kepada perusahaan baik dalam bidang manufaktur, grosir, maupun eceran produk. Prediksi short-term dapat membantu dalam perencanaan produksi dan manajemen stok barang, di sisi lain prediksi long-term dapat membantu dalam pengambilan keputusan untuk pengembangan bisnis. Sales forecasting biasanya dilakukan sewenang - wenang oleh manajer. Namun, manajer dengan kemampuan yang baik sangatlah sulit ditemukan. Maka dari itu, sales forecasting harus dibantu oleh sistem komputer yang dapat membantu manajer untuk mengambil keputusan yang baik dengan memberikan perkiraan dari penjualan yang akan datang[4]. Saat ini, sudah banyak model machine learning yang dapat digunakan untuk melakukan sales forecasting. Sales forecasting atau sales prediction lebih cocok disebut sebagai masalah regression dibandingkan dengan masalah time series. Praktek menunjukkan bahwa penggunaan pendekatan regression sering memberikan kita hasil yang lebih baik dibandingkan metode time series. Kita dapat menemukan pola yang rumit pada dinamika penjualan menggunakan metode supervised machine learning. Beberapa algoritma machine learning yang populer merupakan algoritma berbasis tree seperti Random Forest, dan Gradient Boosting[5]. Dengan berdasarkan hal itu, kita dapat menggunakan model regresi Random Forest untuk melakukan forecasting khususnya pada masalah ini ialah pada data penjualan.

Pengoptimisasian suatu metode atau algoritma merupakan suatu hal yang wajar. Hal ini dilakukan agar hasil yang kita dapatkan menjadi lebih baik dan metode yang kita gunakan menjadi lebih cocok dengan data yang kita gunakan. Salah satu algoritma yang dapat digunakan untuk pengoptimisasian ialah Particle Swarm Optimization (PSO) yang merupakan salah satu algoritma yang terinspirasi dari makhluk hidup dan sederhana untuk mencari sebuah solusi optimal dalam sebuah ruang solusi. Berbeda dengan algoritma optimisasi lainnya, PSO hanya memerlukan objective function dan tidak bergantung pada gradient atau bentuk diferensial dari objektif nya. PSO juga memiliki sedikit hyperparameter[6].

Seperti yang sebelumnya dinyatakan, Random Forest Regression merupakan salah satu model machine learning yang dapat kita gunakan untuk melakukan sales forecasting. Namun, metode Random Forest memiliki kelemahan yaitu salah satunya adalah rentan terjadinya overfitting, khususnya ketika jumlah tree atau kedalaman dari tree nya terlalu tinggi. Maka dari itu, penelitian ini sangatlah penting untuk dilakukan yaitu dengan tujuan untuk mengetahui apakah metode Random Forest Regression ini dapat dioptimisasi dengan menggunakan suatu algoritma yaitu dalam hal ini menggunakan Particle Swarm Optimization untuk mencari hyperparameter yang lebih baik digunakan pada random Forest Regression untuk melakukan sales forecasting pada data penjualan.

2. Metodologi Penelitian

Penelitian ini penulis bagi ke dalam lima tahap, yaitu analisis permasalahan, pengumpulan dan preprocessing data, analisis kebutuhan sistem, perancangan sistem, dan perancangan pengujian dan evaluasi sistem.

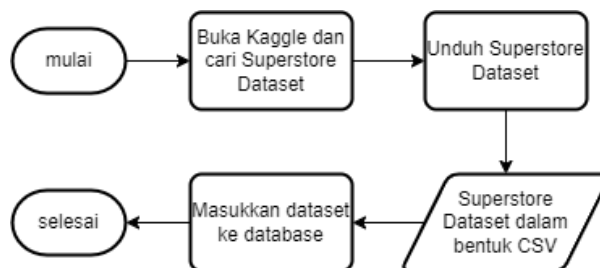
2.1. Analisis Permasalahan

Permasalahan yang ingin diangkat dalam penelitian ini adalah banyaknya data penjualan yang sangat besar dan bervariasi menyebabkan tidak relevannya penggunaan metode sales forecasting tradisional dalam melakukan analisis terhadap data yang sangat besar. Karena kemajuan teknologi, sudah banyak terdapat metode machine learning yang dapat membantu kita dalam melakukan proses sales forecasting terutama pada Big Data, salah satunya ialah metode Random Forest Regression. Metode Random Forest Regression memiliki keunggulan dibanding metode KNN dan Gradient Boosting. Tetapi, metode Random Forest akan rentan mengalami overfitting apabila jumlah tree yang digunakan sangat banyak dan juga memerlukan tuning hyperparameter untuk menentukan suatu hyperparameter

yang dapat memberikan model yang bagus. Maka dari itu, pengoptimalan metode Random Forest sangatlah diperlukan untuk menghindari overfitting.

2.2. Pengumpulan dan Preprocessing Data

Proses pengumpulan dataset dilakukan dengan metode pengambilan data secara manual berdasarkan sumber data. Adapun sumber data yang digunakan adalah dataset Superstore dari Kaggle[7].



Gambar 1. Diagram Alur Pengumpulan Data

Gambar 1 menunjukkan diagram alur pengumpulan data. Pengambilan data dari sumber data dilakukan dengan pengunduhan data dari *repository* kaggle ke *local machine*. Data yang didapat merupakan data dalam bentuk CSV. Kemudian dataset ini disimpan ke dalam database menggunakan *script* Python.

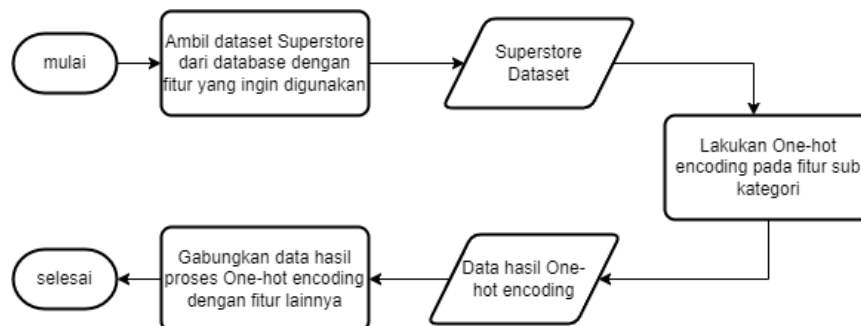
Tabel 1. Spesifikasi Data

Fitur Data	Deskripsi
Row ID	ID unik untuk tiap baris
Order ID	ID Order unik untuk tiap pelanggan
Order Date	Tanggal Order dari produk
Ship Date	Tanggal pengiriman dari produk
Ship Mode	Mode pengiriman yang ditentukan oleh pelanggan
Customer ID	ID unik untuk tiap pelanggan
Customer Name	Nama pelanggan
Segment	Segmen dimana pelanggan diharuskan
Country	Negara dari pelanggan
City	Kota dari pelanggan
State	Negara bagian dari pelanggan
Postal Code	Kode pos dari pelanggan
Region	Wilayah dari pelanggan
Product ID	ID unik produk
Category	Kategori produk yang dipesan
Sub-Category	Sub-kategori produk yang dipesan
Product Name	Nama produk

Sales	Penjualan dari produk
Quantity	Jumlah produk
Discount	Diskon produk
Profit	Keuntungan/kerugian

Tabel 1 menunjukkan spesifikasi dari data yang digunakan. Data yang diambil merupakan data *time series* yang terdiri dari 9994 sampel data penjualan dari tahun 2014 hingga tahun 2017. Pada dataset ini, terdapat 21 fitur yang berkaitan dengan penjualan seperti yang ditunjukkan pada tabel 1. Data ini terdiri dari 17 sub kategori, 3 kategori, dan 1862 produk yang akan digunakan untuk mewakili sebuah big data penjualan.

Setelah data dikumpulkan, data tersebut akan diproses menggunakan teknik One-hot encoding. One-hot encoding merupakan salah satu teknik yang sering digunakan dalam transformasi data, khususnya untuk menangani data kategorikal dalam machine learning. Teknik ini pada dasarnya mengubah data kategorikal ke dalam format yang dapat dipahami oleh algoritma machine learning. Algoritma machine learning biasanya bekerja dengan data numerik. One-hot encoding mencapai hal ini dengan membuat kolom biner baru untuk setiap kategori dalam variabel kategorikal asli. Setiap kolom biner baru mewakili satu kategori. Nilai 1 pada kolom menunjukkan keberadaan dari kategori tertentu sedangkan nilai 0 menunjukkan ketidakhadirannya. One-hot encoding tidak mengartikan sebuah hirarki atau urutan diantara kategori yang dimana hal ini sangat penting karena kategori kategori yang dipakai mungkin saja tidak memiliki keterkaitan urutan.



Gambar 2. Diagram Alur One-Hot Encoding

Gambar 2 menunjukkan alur dari pengimplementasian One-Hot Encoding pada penelitian ini. Pertama tama, fitur fitur yang diperlukan pada dataset yang telah disimpan akan diambil. Adapun fitur yang diambil adalah *Sub Category*, *Sales*, dan *Order Date*. Kemudian, dilakukan algoritma One-Hot Encoding pada fitur sub kategori untuk mengubahnya dari data kategorikal menjadi data numerik. Hasil dari proses One-Hot Encoding ini akan digabungkan kembali dengan fitur lainnya yang telah diambil.

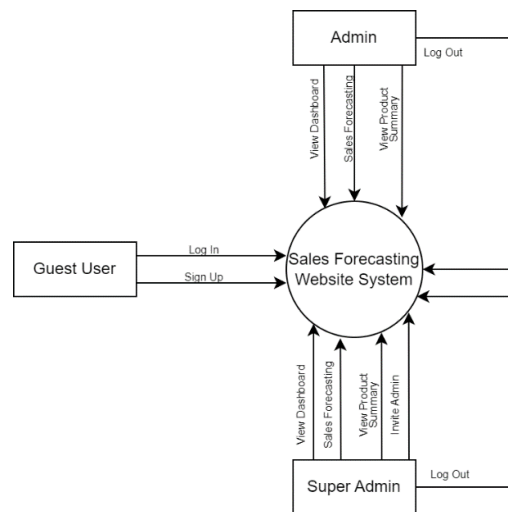
2.3. Analisis Kebutuhan Sistem

Pada tahapan ini, peneliti akan mengumpulkan seluruh informasi mengenai kebutuhan pengguna dengan cara diskusi, observasi, dan atau wawancara. Adapun kebutuhan sistem akan dibagi menjadi dua bagian yaitu kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional mendefinisikan apa yang seharusnya dilakukan oleh aplikasi. Kebutuhan ini berfokus pada fitur dan alur kerja aplikasi yang akan digunakan oleh pengguna. Sedangkan, kebutuhan non fungsional mendefinisikan karakteristik dan kualitas aplikasi secara keseluruhan. Kebutuhan ini tidak berfokus pada fitur, tetapi pada bagaimana aplikasi seharusnya berperilaku.

2.4. Perancangan Sistem

Pada penelitian ini, aplikasi akan dibangun berbasis web. Adapun aplikasi yang dibangun dibagi menjadi dua bagian yaitu bagian Frontend dan bagian Backend. Rancangan sistem yang dibuat pada penelitian ini adalah dalam bentuk Context Diagram, Data Flow Diagram, dan Entity Relationship Diagram.

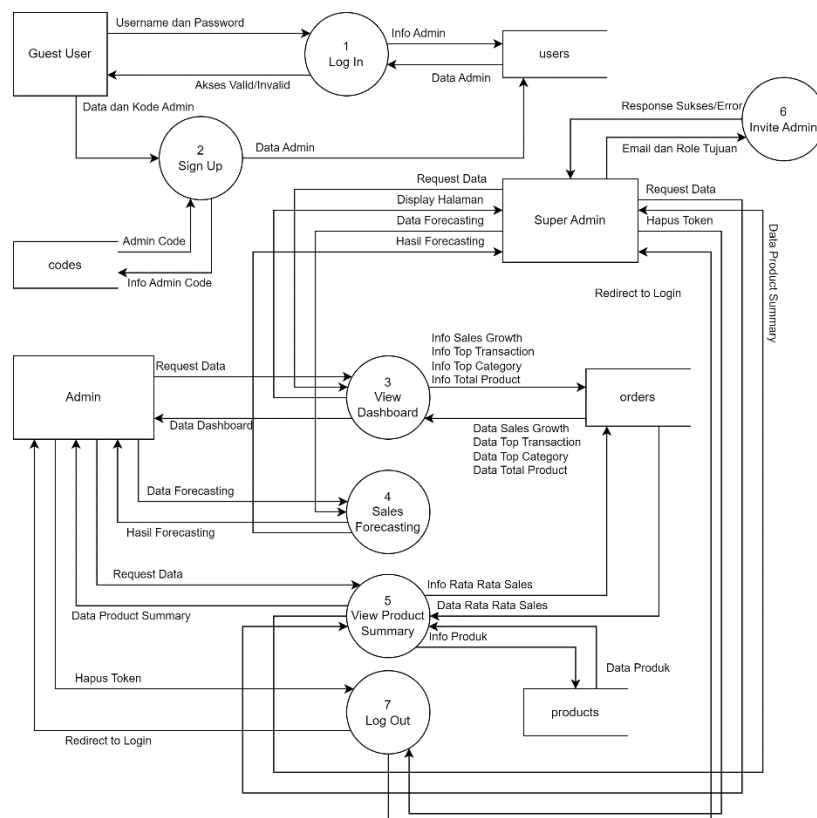
2.4.1. Context Diagram



Gambar 3. Context Diagram

Gambar 3 merupakan diagram konteks dari sistem yang dibuat. Terlihat terdapat tiga entitas eksternal yang berinteraksi dengan sistem yaitu Guest User, Admin, dan Super Admin. Guest User merupakan pengunjung website yang belum melakukan login sehingga tidak atau belum dapat mengakses fitur utama. Admin merupakan seorang pengguna yang telah melakukan login dengan akun yang memiliki *role* Admin. Admin memiliki akses terhadap fitur halaman dashboard, melakukan *sales forecasting*, halaman *product summary*, dan melakukan log out. Sedangkan Super Admin merupakan seorang pengguna yang telah melakukan login dengan akun yang memiliki *role* Super Admin yang dimana Super Admin memiliki akses terhadap fitur yang dapat diakses oleh Admin namun dengan tambahan Super Admin dapat melakukan pengundangan Admin/Super Admin.

2.4.2. Data Flow Diagram (Level 0)



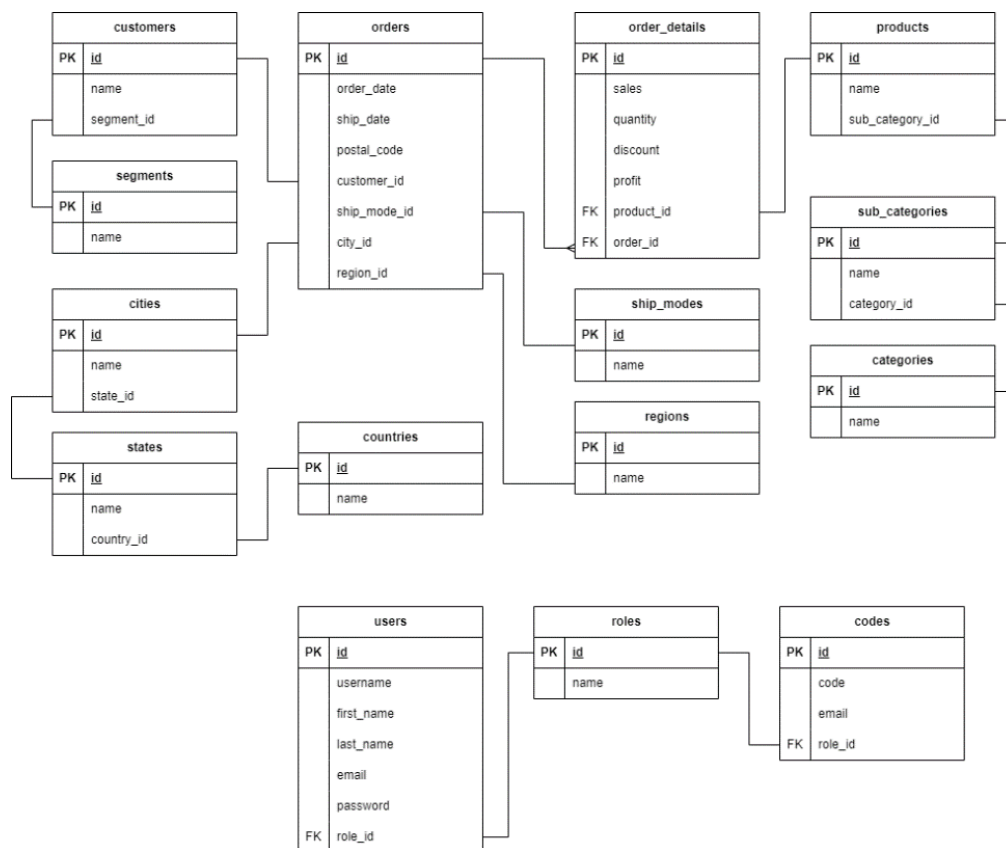
Gambar 4. Data Flow Diagram (Level 0)

Sales Forecasting Menggunakan Random Forest Regression Dengan Particle Swarm Optimization Pada Penjualan Superstore

Gambar 4 menunjukkan data flow diagram yang dibuat berdasarkan diagram konteks yang sebelumnya telah dibuat pada gambar 3. Data flow diagram ini berisi gambaran aliran data yang lebih detail dari diagram konteks untuk menggambarkan rancangan sebuah sistem. Data flow diagram ini terdiri dari tiga entitas eksternal, tujuh proses, dan empat data store. Berikut penjelasan komponen - komponennya:

- a. Entitas Eksternal
 - Guest User: pengguna yang belum melakukan login
 - Admin: pengguna yang sudah melakukan login dengan akun yang memiliki *role* Admin yang dapat mengakses halaman dashboard, melakukan peramalan penjualan, mengakses halaman product summary, dan melakukan log out.
 - Super Admin: pengguna yang sudah melakukan login dengan akun yang memiliki *role* Super Admin yang dapat mengakses fitur yang dapat diakses oleh Admin dengan tambahan dapat melakukan pengundangan Admin atau Super Admin.
- b. Proses
 - Log In: mengotentikasi pengguna agar dapat mengakses fitur sesuai role.
 - Sign Up: melakukan pendaftaran dengan menggunakan data diri pengguna dan kode admin yang telah dikirim di email.
 - View Dashboard: melihat halaman dashboard yang berisikan informasi seperti pertumbuhan penjualan, kategori barang yang paling banyak dibeli, transaksi yang paling banyak, dan juga total produk yang ada.
 - Sales Forecasting: melakukan peramalan penjualan sesuai bulan, tahun, dan sub kategori yang dimasukkan oleh pengguna.
 - View Product Summary: melihat halaman product summary yang berisikan daftar barang dengan data yang ringkas.
 - Invite Admin: melakukan pengundangan Admin atau Super Admin kepada email yang dimasukkan oleh pengguna yang dimana hanya dapat diakses oleh Super Admin.
 - Log Out: melakukan log out pada akun sehingga kembali ke halaman login dan menjadi Guest User.
- c. Data Store/ Penyimpanan Data
 - Users: menyimpan informasi mengenai pengguna yaitu Admin dan Super Admin.
 - Codes: menyimpan informasi mengenai kode admin dan juga email yang terkait dengan kode tersebut.
 - Orders: menyimpan informasi mengenai pesanan produk yang pernah dilakukan oleh customer.
 - Products: menyimpan informasi mengenai data produk yang ada.

2.4.3. Rancangan Entity Relationship Diagram



Gambar 5. Entity Relationship Diagram

Gambar 5 menunjukkan Entity Relationship Diagram dari basis data yang dirancang pada sistem yang dibuat. Sistem yang akan dibangun menggunakan basis data relasional untuk penyimpanan data. Berdasarkan Entity Relationship Diagram pada gambar 5, terdapat 15 entitas yang diantaranya akan menyimpan data dari dataset superstore dan juga data pengguna. Terdapat tabel customers, segments, orders, cities, states, countries, order_details, ship_modes, regions, products, sub_categories, categories, users, roles, dan codes.

2.5. Perancangan Pengujian dan Evaluasi Sistem

Pengujian untuk model Random Forest Regression akan dilakukan dengan menghitung nilai Mean Absolute Error masing masing model yang telah dibuat menggunakan masing masing algoritma optimisasi. Kemudian, hasil evaluasi akan dibandingkan antara model satu dengan lainnya. Hasil perbandingan tersebut akan digunakan peneliti untuk penarikan kesimpulan.

Sedangkan, pengujian untuk sistem dan tampilan disini akan menggunakan Black Box Testing yang dimana peneliti akan melakukan pengujian secara fungsionalitasnya tanpa memerhatikan kode dari sistem yang dibangun. Skenario pengujian pada penelitian ini dapat dilihat pada tabel berikut.

Tabel 2. Skenario Pengujian

Kelas Uji	Butir Uji	Pengujian
Model Random Forest Regression dengan tuning hyperparameter menggunakan Particle Swarm Optimization, Grid Search, dan Randomized Search	Besar error pada masing - masing model peramalan	MAE
Pengujian Login	Pengujian login admin	Black Box Testing

Pengujian Sign Up	Pengujian Signup Admin	Black Box Testing
Pengujian Dashboard	Pengujian Halaman Dashboard	Black Box Testing
Pengujian Peramalan Penjualan	Pengujian pada halaman peramalan	Black Box Testing
Pengujian Product Summary	Pengujian pada halaman Product Summary	Black Box Testing
Pengujian Undang Admin	Pengujian pada halaman undangan admin	Black Box Testing

Tabel 2 menunjukkan skenario pengujian yang dilakukan pada penelitian ini baik untuk pengujian model maupun pengujian sistem. Model yang diujikan menggunakan MAE merupakan model hasil *training* menggunakan hyperparameter hasil optimisasi dari masing masing algoritma optimisasi. Sedangkan untuk pengujian black box, yang diujikan adalah fungsionalitas dari sistem melalui tampilan antarmuka yang dibuat.

3. Hasil dan Diskusi

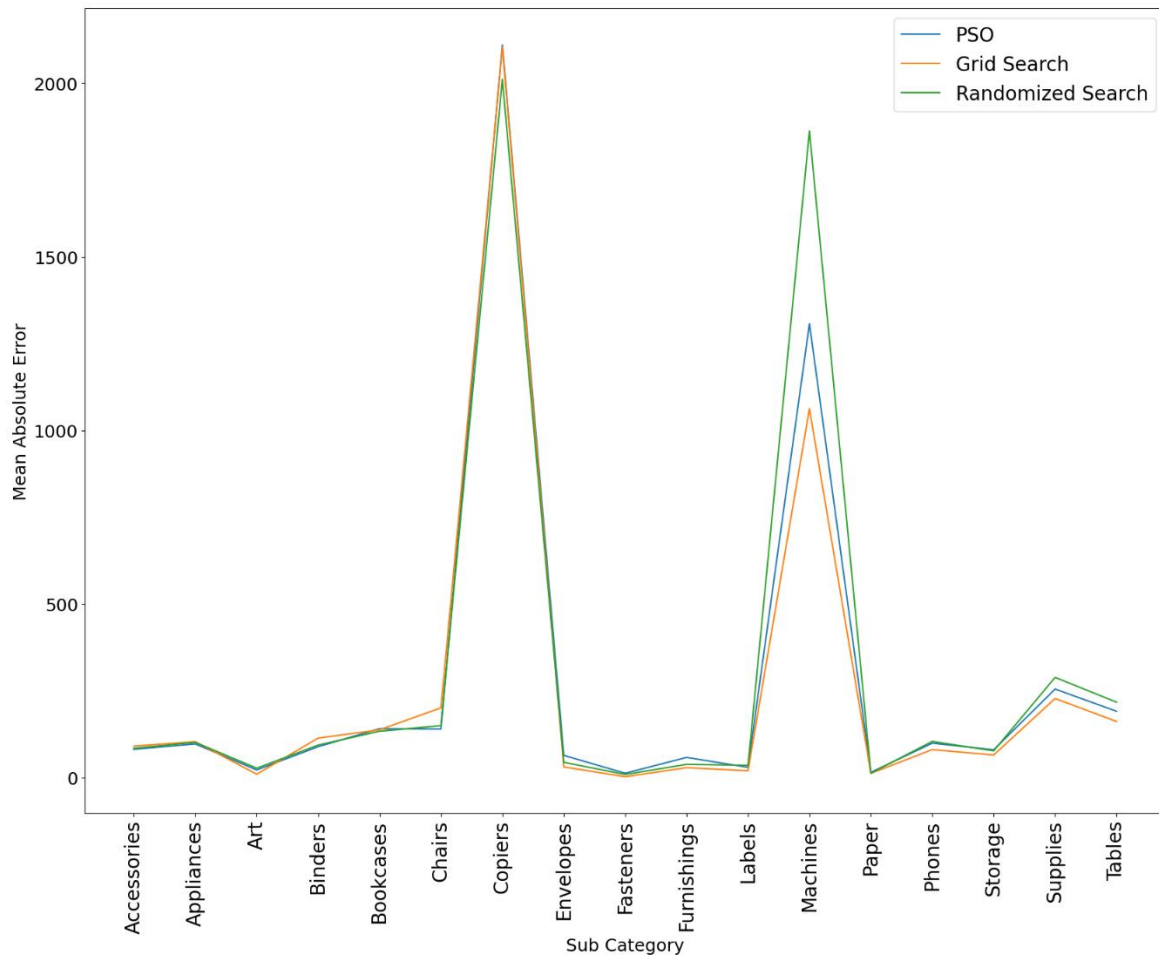
3.1. Hasil Pengujian Error Model

Model yang telah dibuat menggunakan hyperparameter hasil dari optimisasi menggunakan Particle Swarm Optimization, Grid Search, dan Randomized Search dan dilatih menggunakan data training selanjutnya diuji errornya. Pengujian error model dilakukan dengan menggunakan metrik Mean Absolute Error untuk mengetahui rata rata error dari model.

Tabel 3. Hasil Evaluasi Model

Metode Optimisasi Model	Error Pada Data Training	Error Pada Data Testing	Waktu Pengoptimisasian
PSO	187.680487	254.328478	40 menit 42 detik
Grid Search	199.534286	234.099818	710 menit 31.5 detik
Randomized Search	165.270662	285.475800	48 menit 48.1 detik

Tabel 3 menunjukkan hasil evaluasi dari masing masing model. Apabila terlihat sekilas, model dengan optimisasi Randomized Search memiliki nilai error yang paling rendah pada data training namun memiliki nilai error yang paling tinggi pada data testing. Model dengan metode optimisasi Grid Search memiliki error paling rendah pada data testing namun memakan waktu paling lama dibandingkan yang lain. Model dengan metode optimisasi PSO mendapat error yang tidak terlalu tinggi ataupun tidak terlalu rendah dibandingkan yang lainnya. PSO memiliki waktu pengoptimisasian paling cepat dibandingkan Grid Search maupun Randomized Search. Hal ini dikarenakan PSO memiliki fitur yang akan menghentikan proses apabila tidak ada perubahan signifikan saat proses pengoptimisasian berlangsung. Hal ini menjadi keuntungan bagi PSO sehingga pengguna dapat melakukan tuning lagi pada hyperparameter milik PSO untuk mendapat hasil optimisasi yang lebih baik. Sedangkan untuk Grid Search memang dalam hal ini memberikan nilai error yang lebih kecil pada data testing namun Grid Search memiliki waktu pengoptimisasian yang berkali lipat lebih lama dibandingkan algoritma lainnya yaitu selama 710 menit 31,5 detik. Hal ini dapat merugikan pengguna dalam hal waktu apabila pengguna ingin melakukan tuning pada grid milik Grid Search. Performa model tidak dapat secara langsung ditentukan dengan nilai error atau MAE dari rata rata penjualan pada keseluruhan sub kategori. Hal ini memerlukan penelitian lebih lanjut pada masing masing sub kategori. Apabila kita lihat error nilai error dari setiap sub kategori maka akan terlihat seperti grafik berikut.



Gambar 6. Grafik MAE Setiap Sub Kategori

Gambar 6 merupakan data MAE dari setiap sub kategori yang divisualisasikan dalam bentuk grafik garis. Garis yang berwarna biru merupakan data dari model yang menggunakan PSO. Garis yang berwarna oranye merupakan data dari model yang menggunakan Grid Search. Garis yang berwarna hijau merupakan data dari model yang menggunakan Randomized Search. Berdasarkan grafik tersebut, Grid Search lebih unggul dari PSO namun tidak secara signifikan. Hal ini dapat dilihat dari bagaimana PSO lebih unggul dari Grid Search pada beberapa sub kategori.

Apabila dilihat dalam hal kecepatan pemrosesan, PSO lebih unggul secara signifikan dari Grid Search maupun Randomized Search. Hal ini dapat dilihat dari kecepatan pemrosesan PSO yaitu 40 menit 42 detik yang unggul dari Randomized Search yang dimana selama 48 menit 48,1 detik dan unggul jauh dari Grid Search yang dimana selama 710 menit 31,5 detik. Dengan keunggulan kecepatan pemrosesan ini, pengguna tentunya diuntungkan ketika ingin melakukan tuning hyperparameter dari algoritma optimisasi terkait untuk mendapat hasil optimisasi yang lebih baik.

Pada grafik sebelumnya terlihat bahwa terdapat lonjakan error pada beberapa sub kategori misalnya pada "Copiers" dibandingkan dengan sub kategori lainnya seperti "Fasteners". Hal ini terjadi karena adanya fluktuasi harga yang besar yang menyebabkan besarnya error seperti terlihat pada gambar berikut.

	Accessories	Appliances	Art	Binders	Bookcases	Chairs	Copiers	Envelopes	Fasteners	Furnishings	Labels	Machines	Paper	Phones	Storage	Supplies	Tables	Order Month	Order Year	Sales
50	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1	2017	2999.949951
174	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3	2017	3553.303650
306	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5	2017	3359.951904
435	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7	2017	1159.971313
500	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8	2017	589.987000
568	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9	2017	319.984009
636	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	10	2017	11199.967773
703	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	11	2017	4119.978577
771	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	12	2017	1669.971954

Gambar 7. Data Testing Sub Kategori "Copiers"

Gambar 7 menunjukkan adanya perbedaan penjualan yang jauh pada bulan bulan tertentu yang mempengaruhi error pada model regresi. Setiap nilai error pada setiap sub kategori membutuhkan analisis lebih lanjut tiap sub kategori untuk menentukan pengaruh dari error tersebut untuk tiap sub kategori.

3.2. Pengujian Sistem Menggunakan Black Box Testing

Sistem akan diuji dengan menggunakan teknik pengujian Black Box Testing dimana metode pengujian ini bertujuan untuk memastikan setiap fitur ataupun proses telah berfungsi sesuai dengan kebutuhan dan ekspektasi yang diharapkan. Skenario pengujian akan dibagi menjadi beberapa bagian seperti melakukan login (LI), melakukan sign up (SU), melihat dashboard (DS), melakukan peramalan (FR), melihat product summary (PS), mengundang admin (IA), dan melakukan log out (LO). Hasil dari pengujian dari skenario pengujian diatas akan dijelaskan pada tabel 4.

Tabel 4. Hasil Pengujian Black Box Testing

Id Skenario	Pengujian	Hasil Yang Diinginkan	Hasil Yang Didapat	Keterangan
LI01	Login dengan username dan password yang benar	Berhasil masuk ke dalam sistem	Berhasil masuk ke dalam sistem	Sesuai
LI02	Login dengan username dan password yang salah	Tidak dapat masuk ke sistem dan menampilkan error	Tidak dapat masuk ke sistem dan menampilkan error	Sesuai
SU01	Melakukan pendaftaran akun dengan kode admin yang benar, username yang unik, dan data yang sesuai tanpa ada data yang kosong	Berhasil melakukan pendaftaran dan dialihkan ke halaman login	Berhasil melakukan pendaftaran dan dialihkan ke halaman login	Sesuai
SU02	Melakukan pendaftaran namun dengan username yang pernah dipakai yaitu "admin"	Tidak berhasil melakukan pendaftaran dan muncul error	Tidak berhasil melakukan pendaftaran dan muncul error	Sesuai
SU03	Melakukan pendaftaran namun dengan kode admin yang tidak tepat	Tidak berhasil melakukan pendaftaran dan muncul error	Tidak berhasil melakukan pendaftaran dan muncul error	Sesuai
SU04	Menekan tombol "Register Now" dalam keadaan ada <i>field</i> yang kosong	Muncul pesan error	Muncul pesan error	Sesuai
DS01	Mengakses halaman dashboard setelah melakukan login	Masuk ke halaman dashboard tanpa dibawa ke halaman login	Masuk ke halaman dashboard tanpa dibawa ke halaman login	Sesuai
DS02	Mengakses halaman dashboard melalui URL tanpa melakukan login sebelumnya	Dibawa ke halaman login	Dibawa ke halaman login	Sesuai

FR01	Mengakses halaman <i>forecasting</i> setelah melakukan login	Masuk ke halaman <i>forecasting</i> tanpa dibawa ke halaman login	Masuk ke halaman <i>forecasting</i> tanpa dibawa ke halaman login	Sesuai
FR02	Mengakses halaman <i>forecasting</i> melalui URL tanpa melakukan login sebelumnya	Dibawa ke halaman login	Dibawa ke halaman login	Sesuai
FR03	Memasukkan nilai nilai yang berbeda pada field	Menghasilkan angka penjualan yang berbeda	Menghasilkan angka penjualan yang berbeda	Sesuai
PS01	Mengakses halaman <i>product summary</i> setelah melakukan login	Masuk ke halaman <i>product summary</i> tanpa dibawa ke halaman login	Masuk ke halaman <i>product summary</i> tanpa dibawa ke halaman login	Sesuai
PS02	Mengakses halaman <i>product summary</i> melalui URL tanpa melakukan login sebelumnya	Dibawa ke halaman login	Dibawa ke halaman login	Sesuai
PS03	Menekan halaman berbeda pada <i>pagination</i>	Menampilkan produk yang berbeda	Menampilkan produk yang berbeda	Sesuai
IA01	Memasukkan email dan role	Muncul loading lalu muncul pesan "Done"	Muncul loading lalu muncul pesan "Done"	Sesuai
LO01	Menekan tombol log out	Diarahkan ke halaman login	Diarahkan ke halaman login	Sesuai
LO02	Mengakses halaman yang terproteksi seperti halaman dashboard setelah melakukan log out	Diarahkan ke halaman login	Diarahkan ke halaman login	Sesuai

Tabel 4 menunjukkan hasil dari pengujian black box yang telah dilakukan. Terlihat bahwa semua skenario pengujian menunjukkan hasil "Sesuai" yang berarti dari pengujian yang dilakukan, peneliti mendapatkan hasil yang diharapkan pada semua skenario pengujian.

4. Kesimpulan

Berdasarkan penelitian yang telah dilaksanakan dan juga hasil pengujian yang diperoleh dari penelitian, maka didapatkan kesimpulan sebagai berikut:

- Model yang dioptimisasi menggunakan Particle Swarm Optimization memiliki nilai error yaitu sebesar 187,68 pada seluruh data training dan 254,32 pada seluruh data testing. Tingginya nilai error ini disebabkan karena adanya sub kategori dengan penjualan yang sangat tinggi yang memungkinkan nilai kesalahan yang lebih besar sehingga dibutuhkan penelitian yang lebih mendalam.
- Model yang dioptimisasi menggunakan Grid Search lebih unggul dibandingkan dengan model yang menggunakan PSO namun tidak secara signifikan dilihat dari nilai error PSO yang unggul pada beberapa sub kategori dibandingkan dengan Grid Search. Disamping itu, PSO unggul pada waktu pemrosesannya dibandingkan yang lainnya yaitu selama 40 menit 42 detik. Hal ini sangat membantu pengguna untuk melakukan tuning pada hyperparameter milik algoritma optimisasi sehingga memungkinkan pengguna untuk mendapat hasil optimisasi yang lebih baik. Hal ini dapat menjadi keunggulan PSO dibandingkan Grid Search yang membutuhkan waktu hingga 710 menit 31,5 detik.
- Melalui pengujian Black Box pada sistem yang dibuat, dapat disimpulkan bahwa fungsionalitas dari sistem yang dibangun dapat berjalan dengan baik. Pengujian yang dilakukan meliputi

pengujian yang menempatkan penguji sebagai user sehingga penguji dapat berinteraksi dengan frontend aplikasi dan melakukan skenario pengujian yang ditentukan.

References

- [1] S. Sagioglu and D. Sinanc, "Big data: A review," in *2013 International Conference on Collaboration Technologies and Systems (CTS)*, 2013, pp. 42–47. doi: 10.1109/CTS.2013.6567202.
- [2] Bridget Botelho and Stephen J. Bigelow, "big data." Accessed: Mar. 26, 2023. [Online]. Available: <https://www.techtarget.com/searchdatamanagement/definition/big-data>
- [3] N. Thabet and T. R. Soomro, "Big Data challenges," 2019, *Reed Business Information (Cahners)*. doi: 10.4172/2324-9307.1000133.
- [4] G. Tsoumakas, "A survey of machine learning techniques for food sales prediction," *Artif Intell Rev*, vol. 52, no. 1, pp. 441–447, Jun. 2019, doi: 10.1007/s10462-018-9637-z.
- [5] B. M. Pavlyshenko, "Machine-learning models for sales time series forecasting," *Data (Basel)*, vol. 4, no. 1, Mar. 2019, doi: 10.3390/data4010015.
- [6] Adrian Tam, "A Gentle Introduction to Particle Swarm Optimization." Accessed: Mar. 26, 2023. [Online]. Available: <https://machinelearningmastery.com/a-gentle-introduction-to-particle-swarm-optimization/>
- [7] Tableau, "Superstore Dataset." Accessed: Feb. 04, 2023. [Online]. Available: <https://www.kaggle.com/datasets/vivek468/superstore-dataset-final>