**LATAR BELAKANG MASALAH**

Kemajuan zaman dan teknologi saat ini telah merubah pola dalam perusahaan atau pun organisasi-organisasi. Data-data yang besar dan telah terkumpul dalam waktu yang lama dapat diolah menjadi sumber informasi yang dapat membantu menganalisis eksistensi dari sebuah perusahaan atau organisasi. Analisa otomatis dari data yang berjumlah besar atau kompleks yang terbentuk sebagai *data mining*, dengan tujuan untuk menemukan pola atau kecenderungan yang penting biasanya tidak disadari keberadaannya saat ini telah banyak dilakukan[1].

Selain itu permasalah optimasi merupakan permasalahan yang sering ditemui dalam kehidupan sehari-hari. Hal ini tidak terlepas dari sifat manusia yang ingin mendapatkan keuntungan maksimum dan menderita kerugian minimum[2].

Optimasi adalah sarana untuk mengekspresikan model yang bertujuan untuk memecahkan masalah dengan cara terbaik. Model optimasi yang ada digunakan untuk menyelesaikan berbagai permasalahan dalam pemerintah, bisnis, teknik ekonomi, ilmu-ilmu fisika dan sosial yang terkait dengan adanya keterbatasan pengalokasian sumber daya[3].



Gambar 1. 1 Optimasi

*(Sumber: Logique Digital Indonesia)*

Dengan adanya optimasi ini diharapkan dapat membantu perusahaan dalam memaksimalkan penjualan untuk mendapatkan keuntungan yang lebih baik. Optimasi dapat ditempuh dengan dua cara yakni maksimisasi dan minimisasi. Maksimisasi adalah optimasi produk dengan menggunakan atau mengalokasikan input yang sudah tertentu untuk mendapatkan keuntungan yang maksimal. Sedangkan minimisasi adalah optimasi produksi untuk menghasilkan tingkat *output* tertentu dengan menggunakan input atau biaya yang paling minimal.

Persoalan optimasi dibagi menjadi dua jenis, yaitu, tanpa kendala dan dengan kendala. Pada optimasi tanpa kendala, faktor-faktor yang menjadi kendala atau keterbatasan-keterbatasan yang ada terhadap fungsi tujuan diabaikan sehingga dalam menentukan nilai maksimum atau minimum tidak terdapat batasan-batasan terhadap berbagai pilihan alternatif yang tersedia. Sedangkan pada optimasi dengan kendala, faktor-faktor yang menjadi kendala terhadap fungsi tujuan diperhatikan dalam menentukan titik maksimum atau minimum fungsi tujuan[7 chapter II].

Optimasi dengan kendala pada dasarnya merupakan persoalan dalam menentukan nilai variabel suatu fungsi menjadi maksimum atau minimum dengan memperhatikan keterbatasan-keterbatasan yang ada. Keterbatasan itu meliputi *input* atau faktor-faktor produksi. Optimasi produksi dengan kendala perlu memperhatikan faktor-faktor yang menjadi kendala pada fungsi tujuan karena kendala menentukan nilai maksimum dan minimum. Fungsi tujuan merupakan suatu pernyataan matematis yang digunakan untuk mempresentasikan kriteria dalam mengevaluasi solusi suatu masalah. Fungsi tujuan dalam teknik optimasi produksi merupakan unsur yang penting karena akan menentukan kondisi optimasi suatu keadaan[11 chapter II].

# **BAB II**

**LANDASAN TEORI**

## Teori Dasar

### **Optimasi**

Optimasi adalah sarana untuk mengeskpresikan model yang bertujuan untuk memecahkan masalah dengan cara terbaik atau suatu proses untuk mencapai hasil yang ideal. Model optimasi yang ada digunakan untuk menyelesaikan berbagai permasalahan dalam pemerintahan, bisnis, teknik ekonomi, ilmu-ilmu fisika dan sosial terkait dengan adanya keterbatasan pengalokasian sumber daya[3].

### **Peramalan**

Peramalan merupakan hasil dari kegiatan memprediksi atau meramal atau memperkirakan. Prediksi bisa berdasarkan metode ilmiah ataupun subjektif belaka. Sebagai contoh, prediksi cuaca selalu berdasarkan data dan informasi terbaru yang didasarkan pengamatan termasuk oleh satelit. Namun, prediksi seperti pertandingan sepakbola, olahraga, dan lain-lain umumnya berdasarkan pandangan subjektif dengan sudut pandang sendiri yang memprediksinya[5].

### **Data Mining**

*Data mining* adalah sebagai proses untuk mendapatkan informasi yang berguna dari gudang basis data yang besar. *Data mining* juga dapat diartikan sebagai pengekstrakan informasi baru yang diambil dari bongkahan data besar yang membantu dalam pengambilan keputusan. *Data mining* sering juga disebut dengan *Knowledge Discovery in Database* atau disingkat KDD, adalah kegiatan yang meliputi pengumpulan, pemakaian data histori untuk menelusuri data yang ada untuk membangun sebuah model agar dapat mengenali pola data yang lain berukuran besar[6].

*Data mining* dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu sebagai berikut[7].

1. Deskripsi

Untuk mencari cara yang menggambarkan pola dan kecenderungan yang terdapat dalam data. Deskripsi dari pola dan kecenderungan yang sering memberikan kemungkinan penjelasan untuk suatu pola atau kecenderungan.

1. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih kearah numerik daripada kearah kategori. Model dibangun menggunakan *record* lengkap yang menyediakan nilai dari variabel target sebagai nilai prediksi.

1. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang.

1. Klasifikasi

Dalam klasifikasi, terdapat target variabel kategori. Sebagai contoh, penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu pendapatan tinggi, sedang dan rendah.

1. Pengklusteran

Pengklusteran merupakan pengelompokkan *record*, pengamatan atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. *Cluster* adalah kumpulan *record* yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidak miripan dengan *record* dalam *cluster* lain. Algoritma *clustering* mencoba untuk melakukan pembagian terhadap keseluruhan data menjadi kelompok-kelompok yang memiliki kemiripan (homogen), yang mana kemiripan *record* dalam satu kelompok akan bernilai maksimal, sedangkan kemiripan dengan *record* dalam kelompok lain akan bernilai minimal.

1. Asosiasi

Tugas asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam suatu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja (*Market Baskaet Analysis*).

## Bahasa Pemrograman

### **R**

Pemrograman R adalah salah satu paket statistika yang menjadi pilihan baik untuk kepentingan melengkapi pengenalan konsep-konsep statistika, pemrograman maupun sebagai analisis data yang membutuhkan ilustrasi grafik[8]. Bahasa pemrograman R digunakan oleh akedemisi dan peneliti untuk analisis data dan analisis statistik, dan popularitas pemrograman R telah meningkat seiring berjalannya waktu. R diciptakan oleh Ross Ihaka dan Robert Gentlemen. Saat ini R dikembangkan oleh *R Development Core Team*. Munculnya R terinspirasi oleh bahasa statistika dengan S (*statistical language S*). Perintah dasar dalam bahasa R telah menyediakan berbagai *tool* untuk pemodelan statistik linear dan nonlinear, analisis *time-series*, klasifikasi, analisis klaster, dan analisis grafis[9].

### **R-Shiny**

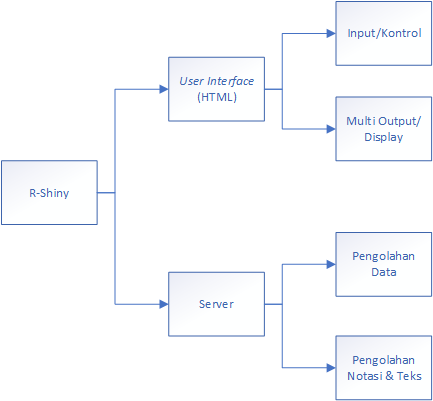
R-Shiny merupakan salah satu versi R berbasis web. Secara umum komponen program Shiny dibedakan menjadi dua kelompok yaitu:

1. *User Interface*, bagian ini bermanfaat untuk[10]:
2. Panel kontrol, ialah panel untuk mengkontrol *input* berupa data, variabel, model, tergantung kompleksitas modul. Tampilan kontrol dapat berupa *slider, radio button, check-box,* dan lain-lain.
3. Pemasukan permintaan nilai *input* (data dengan berbagai jenis variabel yang diperlukan, pemilihan model, jenis dan kriteria uji statistika).
4. Penyajian *output* terkait hasil analisis/uji. Hasil *output* dapat berupa; grafik (histogram, diagram pencar, dan lain-lain). Bentuk angka/teks dapat berbentuk asli (verbatim) maupun berbentuk tabel. Teks khusus dengan notasi matematika dengan format LaTeX.

R-Shiny menyediakan berbagai format *output* seperti *plotOutput, textOutput, verbatimTextOutput, tabelOutput,* dan lain-lain untuk mengakomodasi berbagai luaran tersebut.

Bagian *user interface* dapat disajikan pada file khusus ui.r, dapat juga disajikan penuh melalui file HTML; index.html.

1. *Server*, bagian ini merupakan otak dari program yang bertugas melakukan simulasi, berbagai analisis data sesuai pilihan penggunaan dan selanjutnya mengirim hasilnya ke bagian *output*. *Server* didukung oleh berbagai prosedur dan analisis data yang pada umumnya telah tersedia pada berbagai paket R. Bagian ini disimpan dalam file server.r. Untuk struktur umum komponen pemrograman dengan R-Shiny dapat dilihat pada gambar berikut.



Gambar 2. 1 Struktur Komponen Pemrograman R-Shiny.

## Metode yang Digunakan

### **Regresi Linier**

Metode regresi linier adalah alat statistik yang dipergunakan untuk mengetahui pengaruh antara satu atau beberapa variabel terhadap satu buah variabel terikat. Manfaat dari regresi linier diantaranya analisis lebih akurat dalam melakukan analisis korelasi, karena analisis itu kesulitan dalam menunjukan tingkat perubahan suatu variabel terhadap variabel lainnya (*slope*) dapat ditentukan. Dengan analisis regresi peramalan atau perikiraan nilai variabel terikat pada nilai variabel bebas lebih akurat[11]. Tujuannya untuk mengestimasi serta memprediksi rata-rata populasi atau nilai rata-rata variabel dependen berdasarkan nilai variabel independen yang diketahui. Hasil analisis regresi berupa koefisien pada masing-masing variabel X (indpenden)[12].

Analisis yang memiliki variabel bebas lebih dari satu disebut dengan analisis regresi linier berganda[13]. Regresi berganda merupakan regresi dengan dua atau lebih variabel X1, X2, X3,…,Xn sebagai variabel bebas dan variabel Y sebagai variabel tak bebas, nilai-nilai koefisien atau taksiran parameter regresi berganda dapat diperoleh dengan model regresi linier berganda[14], seperti persamaan berikut:



Keterangan:

Y = Variabel terikat atau *response*.

X = Variabel bebas atau *predictor*.

α = Konstanta.

β = *Slope* atau Koefisien estimate.

Pada penelitian ini, untuk variabel bebasnya (X) terdiri dari jumlah pelanggan dan jumlah produk untuk setiap bulannya. Lalu, untuk variabel terikatnya (Y) adalah pendapatan.

### **Algoritma Genetika**

Algoritma Genetika pertama kali dikenalkan oleh John Holland pada tahun 1960. Bersama murid dan teman-temannya, John Holland mempublikasikan Algoritma Genetika dalam buku yang berjudul *Adaption of Natural and Artificial System* pada tahun 1975. Algoritma genetika merupakan algoritma optimasi yang terinspirasi oleh gen dan seleksi alam. Algoritma ini mengkodekan solusi-solusi yang mungkin ke dalam struktur data dalam bentuk kromosom-kromosom dan mengaplikasikan operasi rekombinasi genetika ke struktur data tersebut[15].

Algoritma ini dimulai dengan pembentukan himpunan individu yang diwakili oleh kromosom. Himpunan kromosom ini disebut populasi awal. Populasi awal dapat dibentuk secara acak ataupun dengan metode heuristik. Sebelum membentuk populasi awal, dibutuhkan representasi solusi ke dalam kromosom.

1. Teknik Pengkodean (*Encoding*)

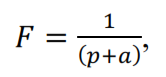
Teknik pengkodean adalah proses pengkodean gen dari kromosom. Gen merupakan bagian dari kromosom, satu gen biasanya akan mewakili satu variabel. Gen dapat direpresentasikan dalam bentuk bit, bilangan ril, daftar aturan, elemen permutasi, elemen program atau representasi lainnya yang dapat diimplementasikan dalam operator genetika.

1. Mengevaluasi Nilai *Fitness*

Mengevaluasi nilai *fitness* berfungsi untuk mengukur kualitas dari sebuah solusi dan memungkinkan tiap solusi untuk dibandingkan. Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran baik tidaknya individu tersebut. Di dalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup, sedangkan individu yang bernilai *fitness* rendah akan mati. Pada masalah optimasi, fungsi *fitness* yang digunakan adalah



Dengan *p* merupakan nilai dari individu, yang artinya semakin nilai *p*, maka semakin besar nilai *fitness*nya. Tetapi hal ini akan menjadi masalah jika *p* bernilai 0, yang mengakibatkan *F* bisa bernilai tak hingga. Untuk mengatasinya, *p* perlu ditambah sebuah bilangan sangat kecil sehingga nilai *fitness*nya menjadi



Dengan *a* adalah bilangan yang dianggap sangat kecil (konstanta) dan bervariasi sesuai dengan masalah yang akan diselesaikan.

1. Seleksi (*Selection*)

Seleksi merupakan pemilihan dua buah kromosom untuk dijadikan sebagai induk yang dilakukan secara proporsional sesuai dengan nilai *fitness*nya.

1. Pindah Silang (*Crossover*)

Pindah silang (*crossover*) adalah operator dari algoritma genetika yang melibatkan dua induk untuk membentuk kromosom baru. Pindah silang menghasilkan keturunan baru dalam ruang pencarian yang siap diuji. Operasi ini tidak selalu dilakukan pada setiap individu yang ada. Individu dipilih secara acak untuk dilakukan *crossover* dengan *Probabilitas Crossover* (PC).

1. Mutasi (*Mutation*)

Mutasi merupakan proses untuk mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Operasi mutasi yang dilakukan pada kromosom dengan tujuan untuk memperoleh kromosom-kromosom baru sebagai kandidat solusi pada generasi mendatang dengan *fitness* yang lebih baik, dan lama-kelamaan menuju solusi optimum yang diinginkan. Akan tetapi, untuk mencapai hal ini, penekanan selektif juga memegang peranan yang penting. Jika dalam pemilihan kromosom-kromosom cenderung terus pada kromosom yang memiliki *fitness* tinggi saja, konvergensi prematur akan sangat mudah terjadi.

1. Pembentukan Populasi Baru

Proses membangkitkan populasi baru bertujuan untuk membentuk populasi baru yang berbeda dengan populasi awal. Pembentukan populasi baru ini didasarkan pada keturunan-keturunan baru hasil mutasi ditambah dengan individu terbaik setelah dipertahankan dengan proses *elitism*. Setalah populasi baru terbentuk, kemudian mengulangi langkah-langkah evaluasi nilai *fitness*, proses seleksi, proses pindah silang, proses mutasi pada populasi baru untuk membentuk populasi baru selanjutnya.

Algoritma genetika (*Genetic Algorithm, GA*) adalah algoritma pencarian yang didasarkan atas mekanisme seleksi alami dan evolusi biologis. Algoritma genetika mengkombinasikan antara deretan struktur dengan pertukaran informasi acak ke bentuk algoritma pencarian dengan beberapa perubahan bakat pada manusia. Pada setiap generasi, himpunan baru dari deretan individu dibuat berdasarkan kecocokan pada generasi sebelumnya[16].

Satu siklus iterasi algoritma genetika (sering disebut sebagai generasi) terdapat dua proses, yakni proses seleksi dan rekombinasi. Proses seleksi adalah proses evaluasi kualitas setiap *string* didalam populasi untuk memperoleh peringkat calon solusi. Berdasarkan hasil evaluasi, dipilih *string-string* yang akan mengalami proses rekombinasi. Proses pemilihan biasanya dilakukan secara acak, *string* dengan kualitas yang lebih baik akan memiliki peluang besar untuk terpilih sebagai calon-calon *string* generasi berikutnya[16].

Cara kerja algoritma genetika sangat sederhana, hanya mencakup proses penduplikasian *string-string* dan pertukaran bagian-bagian dari *string*. Meskipun cukup sederhana, tetapi mempunyai kemampuan untuk menyelesaikan persoalan optimasi. Kemampuan ini didukung oleh tiga operator genetik yaitu reproduksi, rekombinasi, dan mutasi. Pada reproduksi terjadi proses penduplikasian *string* berdasarkan nilai fungsi objektifnya. Nilai objektif ini dapat dilihat sebagai suatu keuntungan yang ingin dicapai atau dimaksimalkan. Sementara proses pertukaran bagian-bagian *string* dilakukan oleh operator rekombinasi dan mutasi[16].

## *Package* yang Digunakan

### **Genetic Algorithm (GA)**

Paket GA adalah kumpulan fungsi tujuan umum yang menyediakan seperangkat alat yang fleksibel untuk menerapkan berbagai metode algoritma genetik. Algoritma genetika adalah algoritma pencarian stokastik yang diilhami oleh prinsip dasar evolusi biologis dan seleksi alam. GA mensimulasikan evolusi organisme hidup, dimana individu yang paling cocok mendominasi diatas yang lemah, dengan meniru mekanisme biologis evolusi, seperti seleksi, persilangan, dan mutasi. GA dapat menentukan operator genetik baru dan mudah mengevaluasi kinerja tersebut[17].

### **Shiny**

*Shiny* merupakan package yang berfungsi untuk membuat tampilan berupa *web* yang isinya terdapat diagram ataupun chart mengenai data yang telah diolah sebelumnya. Aplikasi *shiny* ini dibangun menggunakan dua R skrip yang berkomunikasi satu sama lain, diantaranya: skrip *User-Interface* (UI.R) untuk mengontrol tata letak dan penampilan, dan *Server-Script* (server.R) untuk memasukkan petunjuk *input* pengguna, pengolahan data, dan *output* dengan memanfaatkan bahasa R dan fungsi dari paket yang sudah terinstall[18].

### **Shinydashboard**

Shinydashboard merupakan suatu *package* yang disediakan oleh R yang membantu dalam memberi tampilan *dashboard* pada R shiny, selain menambah nilai estetika dan juga menambah fleksibilitas dalam mengakses fitur dalam aplikasi web, sehingga membantu *developer* dalam mengembangkan aplikasi dengan *user experience* yang lebih dibanding hanya dengan menggunakan *package* R shiny sendiri[19].

### **Foreach**

Paket *foreach* menyediakan pembuatan *looping* baru untuk mengeksekusi kode R berulang kali. Salah satu alasan menggunakan paket *foreach* adalah bahwa paket ini mendukung eksekusi paralel, yaitu, dapat mengeksekusi operasi yang diulang pada beberapa prosesor yang terdapat pada komputer, sehingga jika menjalankan operasi secara ratusan kali dapat dieksekusi secara paralel pada prosesor komputer tersebut[20].

## Studi Literatur

Penelitian sebelumnya berguna bagi penulis untuk dapat menjadi pedoman serta pegangan penelitian yang akan penulis lakukan, sehingga dengan adanya penelitian sebelumnya dapat membantu dan memudahkan penulis dalam melakukan penelitian sesuai dengan tema. Beberapa penelitian terdahulu dengan topik yang sama serta menggunakan metode yang sama dengan penelitian penulis, diantaranya sebagai berikut.

1. *Efficient Genetic Algorithm for Feature Selection for Early Time Series Classification*.

Penelitian ini bertujuan untuk memaksimalkan kinerja klasifikasi dan meminimalkan waktu mulai dan waktu eksekusi klasifikasi. Penelitian ini menetapkan model matematis multi-objektif untuk pemilihan fitur mengingat kerja, waktu eksekusi, dan waktu mulai klasifikasi. Eksekusi dan waktu mulai klasifikasi diukur dengan cara vektor fitur yang dipilih.

Untuk mengembangkan algoritma genetika efisien untuk memecahkan model yang mapan. Algoritma genetika dipilih karena cara untuk menangani probabilitas cocok untuk masalah tersebut. Artinya, probabilitas untuk memilih fitur tertentu selama proses algoritma harus disesuaikan dengan indeks fitur kandidat, yang relatif lebih mudah di algoritma genetika daripada metode heuristik lainnya[21].

1. *Evolving RBF Neural Networks For Rainfall Prediction Using Hybrid Particle Swarm Optimization and Genetic Algorithm*

Penelitan ini bertujuan untuk menentukan parameter radial fungsi jaringan saraf (jumlah *neuron*, masing-masing pusat dan radii) secara otomatis. Prediksi curah hujan yang akurat dan tepat waktu sangat penting untuk perencanaan dan pengelolaan sumber daya air, khususnya untuk sistem peringatan banjir, karena dapat memberikan informasi yang efektif untuk membantu mencegah korban dan kerusakan yang disebabkan oleh bencana alam. Prediksi curah hujan sangat sulit untuk model sejak kompleksitas dari proses atomosfer melibatkan pola nonlinier yang agak kompleks, misalnya tekanan, suhu, kecepatan angin dan arahnya, karakteristik meteorologi dari area curah hujan dan sebagainya[22].

1. *Assigning dispatching rules using a genetic algorithm to solve a hybrid flow shop scheduling problem.*

Penelitian ini bertujuan untuk menunjukkan bagaimana menetapkan aturan pengiriman standar yang berbeda untuk jumlah yang telah ditetapkan pada setiap titik waktu dengan menggunakan algoritma genetika. Sebuah penjadwalan yang baik, penting bagi perusahaan manufaktur di Indonesia karena untuk memenuhi tanggal pengiriman pelanggan dan mencapai pemanfaatan sumber daya produksi yang baik.

Untuk menemukan solusi yang optimal, peneliti mengusulkan menggunakan algoritma genetika untuk menetapkan aturan pengiriman standar yang berbeda untuk jumlah yang ditetapkan pada setiap titik waktu. Algoritma genetika tidak hanya dapat menentukan urutan aturan pengiriman yang diterapkan di toko, tetapi juga jangka periode dimana aturan pengiriman dipilih untuk selanjutnya diterapkan[23].

1. *Improved Genetic Algorithm for Solving Flexible Job Shop Scheduling Problem.*

Penelitian ini bertujuan untuk memecahkan masalah penjadwalan pekerjaan yang fleksibel. Sebuah algoritma genetik ditingkatkan diusulkan untuk mengatasi kekurangan algoritma genetik tradisional, seperti lemah mencari kemampuan dan waktu yang lama berjalan ketika memecahkan FJSP. FJSP itu sendiri adalah salah satu masalah optimasi kombinatorial yang paling sulit, dan algoritma perencanaan tradisional telah terbukti sulit untuk mendapatkan solusi yang memuaskan dalam waktu yang dapat diterima.

Dalam rangka untuk memecahkan masalah ini dan lebih meningkatkan kecepatan kovergensi algoritma genetik dalam berurusan dengan FJSP, secara khusus, algoritma tersebut menggunakan metode inisialisasi baru untuk meningkatkan kualitas awal penduduk dan menciptakan cara baru mutasi, yang tidak memerlukan rekonstruksi kromosom setelah mutasi (yaitu, tidak ada solusi ilegal yang dihasilkan). Hasil simulasi menunjukkan bahwa algoritma yang diusulkan memiliki keuntungan dari kemampuan pencarian yang lebih kuat dan waktu berjalan lebih pendek, dibandingkan dengan beberapa algoritma lainnya[24].

1. *Efficiency of the Rail Sections in Brazilian Railway System, Using TOPSIS and a Genetic Algorithm to Analyse Optimized Scenarios*.

Penelitian ini bertujuan untuk menyelidiki kinerja ekstrem BRCS (Sistem kargo kereta api di Brazil) melalui model *hybrid* baru yang menggabungkan TOPSIS dengan algoritma genetika untuk memperkirakan bobot dalam skenario yang dioptimalkan. Brazil memiliki jaringan modal transportasi kargo yang tidak seimbang jika dibandingkan dengan negara-negara dengan dimensi wilayah besar. Penyisipan transportasi kereta api rendah hanya sebesar 15% sementara transportasi jalan yang tinggi dengan pangsa pasar 65%, termasuk untuk perjalanan jauh.

Kebijakan publik harus berusahan untuk mengubah kenyataan ini dalam jangka panjang untuk menyeimbangkan kembali jaringan transportasi Brazil, mengurangi biaya transportasi dan logistik, dan emisi polutan yang dihasilkan dari pembakaran bahan bakar di sektor transportasi Brazil. Metodologi yang diusulkan dapat diterapkan ke berbagai sektor ekonomi yang diperlukan sebagai jaringan seperti sistem kereta api penumpang dan kargo serta energi atau saluran transmisi telekomunikasi[25].

1. *Applying Social Network Analysis to Genetic Algorithm in Optimizing Project Risk Response Decisions*.

Penelitian ini bertujuan untuk mengoptimalkan dalam membuat keputusan respon risiko (RRD). Risiko adalah entitas yang muncul dihampir semua aspek proyek. Suatu proyek rentan terhadap kegagalan jika tindakan respons risiko tidak cukup memadai untuk menangani risiko yang mungkin terjadi. Namun demikian, karena kompleksitas proyek, fenomena interaksi risiko muncul, yaitu, risiko yang teridentifikasi kemungkinan akan memicu terjadinya satu atau lebih risiko.

Interaksi risiko dapat membuat tindakan respons tidak valid, tindakan yang menghilangkan probabilitas spontan risiko tidak dapat mengurangi frekuensi risiko ini. Namun, beberapa fenomena yang dipimpin oleh interaksi risiko sulit untuk ditangani dengan metode analitis. Dalam penelitian ini, model simulasi menggambarkan jaringan interaksi risiko (RIN) dibangun untuk mengevaluasi RRD.

Model tersebut dapat dianggap sebagai model pemrograman linier. Untuk menyelesaikan model ini, model evaluasi diperlukan untuk menghitung nilai fungsi objektif, dan teknik solusinya meliputi teknik optimasi matematis dan teknik *heuristik*. Algoritma genetika dipilih karena teknik solusi dalam model optimasi mampu menghasilkan solusi berkualitas tinggi dalam masalah optimasi jaringan[26].

1. *Applying Genetic Algorithm With Spectation For Optimization of Grid Template Pattern Detection in Financial Markets*.

Penelitian ini bertujuan untuk mengurangi risiko kegagalan dan meningkatkan laba. Makalah ini menyajikan pendekatan keuangan komputasi baru. Ini dengan menggabungkan teknik pengenalan pola grid yang bersekutu dengan kernel optimasi perhitungan evolusi berdasarkan algoritma genetika, menciptakan cara dinamis untuk menghubungkan skor dengan sinyal yang mempertimbangkan volatilitas dan menormalkan dekeksi pola dengan memperbaiki ukuran grid.

Untuk pencocokan pola, pendekatan dengan menggunakan kisi ukuran bobot yang tetap diadopsi untuk menggambarkan pola perdagangan yang diinginkan, tidak hanya mempertimbangkan harga penutupan, tetapi juga variasi harga dalam interval waktu yang dipertimbangkan dari jarak waktu. Skor yang diberikan ke grid bobot akan dioptimalkan oleh algoritma genetika dan pada saat yang sama, keragaman genetik dari solusi yang mungkin akan dipertahankan menggunakan teknik spesiasi, memberikan waktu bagi individu untuk dioptimalkan.

Pendekatan yang dikembangkan menggunakan grid bobot memiliki 21,3% dari pengembalian rata-rata selama periode pengujian terhadap 10,9% dari pendekatan yang ada dan penggunaan spesiasi meningkatkan beberapa hasil pelatihan karena keragaman genetik dipertimbangkan[27].

1. *Structural Optimization of Concrete Arch Bridges Using Genetic Algorithms.*

Penelitian ini bertujuan untuk menunjukkan metodologi dalam menemukan desain biaya terendah, dalam hal volume material, dengan mencari profil yang optimal. Jembatan beton digunakan untuk jalan raya dan jalan rel. Mereka dicirikan oleh daya tahan, kekuan, ekonomi dan kecantikan. Dalam penelitian ini disajikan studi optimasi struktur geometri untuk jembatan lengkung beton dengan teknik algoritma genetika.

Metodologi optimasi yang diusulkan terbukti merupakan teknik yang berhasil untuk proses optimal struktural yang diselidiki. Itu dapat mencapai cukup solusi optimal dengan biaya perhitungan yang masuk akal. Meskipun, hasil menunjukkan penurunan yang wajar dalam berat untuk solusi yang optimal. Desain optimal yang dihasilkan, setelah peningkatan geometris, memiliki rentang presentase pengurangan dari 30% menjadi 35% dibandingkan dengan desain tradisional[28].

1. *Hybrid Genetic Algorithm for Optimization of Food Composition on Hypertensive Patient*.

Penelitian ini bertujuan untuk mengurangi kemungkinan perubahan hipertensi menjadi penyakit berbahaya. Penyakit tidak menular pada dekade tahun ini menjadi penyakit terdepan yang menyebabkan kematian di Indonesia Sebagai contoh salah satu penyakit tidak menular yang banyak dideria pasien adalah hipertensi. Ini terkait dengan pola hidup manusia sehari-hari seperti makanan yang dikonsumsi, status gizi, pola makan dan faktor-faktor lain seperti tekanan darah, obesitas, serta resistensi kolesterol dan insulin.

Batas wajar untuk masalah ini ialah jumlah *natrium* maksimum yang harus ada dalam makanan adalah 0,8 gram. Hasil dari metode yang diusulkan dalam penelitian ini memberikan hasil yang lebih menarik, yaitu perbedaan jumlah nutrisi antara hasil optimasi *hybrid* algoritma genetika dan VNS dengan jumlah nutrisi yang dibutuhkan oleh pasien lebih sedikit (dalam persen)[29].

1. *Genetic Algorithm for Lecturing Schedule Optimization (Case Study: University of Boyolali*).

Penelitian ini bertujuan untuk menyelesaikan permasalahan penjadwalan kuliah. Penyelesaian yang akan dilakukan pada masalah tersebut didekati dengan membangun sebuah *library* dengan konsep Algoritma Genetika (AG). AG adalah sebuah metode komputasi yang terinspirasi oleh proses seleksi alam. *Library* yang dibangun tersusun oleh beberapa operator seperti *Turnamen Selection, Uniform Crossover, Weak Parent Replacement* dan dua metode mutasi yaitu *Interchanging Mutation* dan *Violated Directed Mutation* (VDM).

Kedua metode mutasi dibandingkan untuk mendapatkan metode mutasi yang ketentuan penjadwalan yang belum diakomodasi didalam *library* tanpa modeifikasi inti program. Hasil uji coba menunjukkan bahwa mutasi dengan VDM lebih menjanjikan solusi yang optimal daripada menggunakan *Interchanging Mutation*[30]*.*

1. Implementasi Algoritma Genetika Dengan Variasi *Crossover* Dalam Penyelesaian *Capacitated Vehicle Routing Problem With Time Windows* (CVRPTW) Pada Pendistribusian Air Mineral.

Penelitian ini bertujuan untuk mengatasi keterlambatan dalam mendistribusikan galon air mineral karena pada kasus tersebut PT Artha Envirotama sebagai salah satu perusahaan yang bergerak dalam pelayanan ketersediaan air minum memerlukan strategi pendistribusian yang tepat agar dapat mendistribusikan air mineral secara efektif sehingga mampu meminimalkan waktu pendistribusian air mineral ke pelanggan-pelanggannya dan semua pelanggan mendapatkan pelayanan yang baik.

Saat ini, pendistribusian galon air mineral oleh Evita ke seluruh pelanggan-pelanggannya masih sering terjadi keterlambatan dan waktu distribusi masih lebih dari batas waktu yang diharapkan, yaitu 480 menit untuk setiap kendaraan. Artinya, setiap kendaraan yang mendistribusikan galon air mineral harus selesai melakukan pendistribusan tidak lebih dari 480 menit.

Untuk mengatasi keterlambatan distribusi air mineral ke masing-masing pelanggan agar distribusi dapat selesai tepat waktu, diperlukan rute distribusi yang optimal sehingga galon air mineral lebih cepat sampai pada pelanggan. Oleh karena itu dibutuhkan metode optimasi yang dapat diterapkan untuk menentukan rute terpendek[15].

1. Penerapan Algoritma Genetika Untuk Penentuan Batasan Fungsi Keanggotaan Fuzzy Tsukamoto Pada Kasus Peramalan Permintaan Barang.

Penelitian ini bertujuan untuk mengetahui peramalan permintaan barang produksi semen. Peramalan permintaan diharapkan akan terealisir untuk jangka waktu tertentu pada masa yang akan datang, peramalan perimintaan tersebut akan menjadi masukan yang sangat penting dalam mengambil keputusan perencanaan dan pengendalian suatu perusahaan.

Dalam kegiatan peramalan memerlukan penerapan metode yang optimal dalam menghasilkan suatu peramalan permintaan. Hal tersebut bertujuan agar dapat mengetahui tingkat permintaan yang akan datang dan meminimumkan kesalahan pada perhitungan peramalan. Apabila nilai dalam peramalan yang kurang tepat, maka akan menyebabkan ketidaksesuaian kuantitas dan kualitas produk dengan permintaan pasar.

Sistem yang akan dibangun pada penelitian ini mampu menghasilkan nilai akurasi yang tepat dalam meramalkan suatu permintaan barang Sistem inferensi Fuzzy Tsukamoto bisa diimplementasikan untuk peramalan. Salah satu permalahan dalam penerapan metode Fuzzy adalah sulitnya menentukan batasan fungsi keanggotaan yang tepat. Karena itu pada tulisan ini diusulkan penggunaan algoritma genetika untuk memperbaiki batasan fungsi keanggotaan fuzzy sehingga didapatkan hasil peramalan yang lebih akurat[31].

1. Optimasi Pemodelan Regresi Linier Berganda Pada Prediksi Jumlah Kecelakaan Sepeda Motor Dengan Algoritma Genetika.

Penelitian ini bertujuan untuk memprediksi daerah yang sering terjadi kecelakaan lalu lintas. Karena angka kecelakaan lalu lintas yang terus meningkat membuat berbagai pihak mencari solusi untuk mencegah terjadinya kecelakaan lalu lintas. Agar terjadi relevan antara solusi yang dihasilkan dengan masalah yang ada, maka diperlukan informasi yang dapat menunjang dari data kecelakaan lalu lintas yang selama ini terjadi.

Sistem memprediksi daerah yang sering terjadi kecelakaan lalu lintas berdasarkan parameter yang digunakan antara lain panjang jalan, lebar badan volume, kecepatan, jumlah lajur, jumlah arah, pembatas/median, akses persil dan lebar bahu. Parameter tersebut digunakan untuk memberikan informasi, sehingga dapat dicari solusi pencegahan kecelakaan lalu lintas.

Dari permasalan yang ada makan dibuat model dengan persamaan regresi linier yang terdiri dari variabel terikat (Y) dan variabel bebas (X) yang dioptimasi dengan menggunakan algoritma genetika[32].

1. Penerapan Data Mining Menggunakan Perbandingan Algoritma Greedy Dengan Algoritma Genetika Pada Prediksi Rentet Waktu Harga Crude Palm Oil.

Penelitian ini bertujuan untuk membuat sebuah model yang menggambarkan bagaimana sifat informasi harga *Crude Palm Oil* (CPO) tersebut dan informasi harga CPO dapat terbentuk sedemikian rupa sampai dengan informasi harga CPO pada saat ini (*present*).

Dalam hal ini *neural network* dipilih karena merupakan model non-linear yang dapat dilatih untuk dapat memetakan data historikal dan data masa depan dari data *time series* dengan cara demikian ekstrak struktur hidden dan hubungannya yang dapat menentukan data yang diramalkan.

Algoritma yang termasuk penggunaannya dalam permasalahan optimasi adalah algoritma genetika. Salah satu daya tarik algoritma genetika terletak pada kesederhanaan dan kemampuan untuk mencari solusi yang baik dan cepat untuk masalah yang kompleks[33].

1. Optimasi Persediaan Barang Dalam Produksi Jilbab Menggunakan Algoritma Genetika.

Penelitian ini bertujuan untuk mengoptimasi persediaan barang dalam produksi jilbab. Pentingnya persediaan berpengaruh pada kelancaran kegiatan operasional dalam tujuannya memproduksi suatu barang. Dengan persediaan, perusahaan dapat melayani kebutuhan konsumen akan produk yang ditawarkan. Persediaan barang menjadi hal penting untuk penentuan keuntungan bagi perusahaan.

Dengan sistem persediaan yang baik, suatu perusahaan akan mendapatkan keuntungan yang optimal. Banyak sediktinya persediaan akan berpengaruh terhadap biaya-biaya yang harus dikeluarkan oleh perusahaan.

Algoritma genetika diterapkan pada kasus optimasi barang untuk mendapatkan keuntungan yang maksimal dengan penghematan biaya persediaan. Optimasi persediaan barang menggunakan teknik *one-cut* *point crossover*, mutasi dengan *exchange mutation*, dan seleksi menggunakan *elitism selection*[34].

# **BAB IV**

**METODOLOGI PENELITIAN**

## Diagram Alur Metodologi Penelitian

Metodologi penelitian merupakan pendekatan yang sistematis untuk keseluruhan kegiatan penelitian. Permasalahan atau pertanyaan penelitian dapat diselesaikan dengan pendekatan metodologi tertentu. Dalam metodologi penelitian dipelajari bagaimana proses dan tahapan suatu kegiatan penelitian. Metodologi penelitian bertujuan untuk membuat berbagai tahap penelitian, mulai dari rencana kerja suatu penelitian sampai dengan publikasi[32]. Metodologi penelitian yang digunakan adalah CRISP-DM (*Cross Standart Industries for Data Mining*). Metodologi CRISP-DM adalah standarisasi *data mining* yang disusun oleh tiga penggagas *data mining* *market* yaitu Daimcler Chrysler (Daimler Benz), SPSS, dan NCR. Kemudian dikembangkan pada berbagai *workshop* (antara tahun 1997-1999). Lebih dari 30 organisasi yang berkontribusi dalam proses *modelling* ini dan akhirnya CRISP-DM dipublikasikan pada tahun 1999[33].



Gambar 4. 1 Diagram Alur CRISP-DM.

## Tahapan-Tahapan Diagram Alur Metodologi Penelitian

Terdapat beberapa tahapan dalam CRISP-DM (*Cross Industries Process for Data Mining*), namun pada penelitian ini tahapan yang dilakukan sampai pada tahapan *deployment* dikarenakan tujuan dari penelitian ini adalah untuk mengetahui pengaruh jumlah pelanggan dan jumlah produk terhadap nilai pendapatan serta untuk mengoptimasi nilai pendapatan tersebut[34].

### **Bussiness Understanding**

Fase ini meliputi penentuan tujuan bisnis, menilai situasi saat ini, menetapkan tujuan data mining, dan mengembangkan rencana penelitian. Tujuan bisnis yang akan dilakukan pada penelitian ini adalah bagaimana mengetahui pengaruh jumlah pelanggan dan jumlah produk terhadap nilai pendapatan menggunakan Regresi Linier Berganda dan Algoritma Genetika untuk melakukan optimasi nilai tersebut.

### **Data Understanding**

Fase ini dikenal sebagai fase pemahaman terhadap data yang diperoleh dan kemudian data awal yang dikumpulkan melalui observasi langsung, dengan melakukan analisa terhadap berkas data pelanggan pada objek penelitian. Kajian perlu dilakukan sehingga diketahui data yang akan digunakan. Data yang digunakan dalam penelitian berjumlah 2000 *records* sebagai populasi, dimana data tersebut diambil dari periode bulan Januari sampai dengan Oktober berdasarkan berkas yang ditunjukan oleh objek penelitian. Jika data yang diperoleh semakin banyak, maka hasil dari akurasi datanya juga akan maksimal. Dalam tahapan ini, perlu identifikasi terhadap masalah kualitas data yang baik sehingga diperoleh nilai subset yang sesuai, menarik dalam pembuatan hipotesa awal.

### **Data Preparation**

Tahapan ini merupakan tahapan pengolahan data atau dapat juga dikatakan sebagai tahapan persiapan data. Banyak persiapan yang dilakukan pada tahapan ini sehingga tak jarang fase ini juga disebut sebagai fase padat karya. Beberapa kegiatan seperti pemilihan tabel dan *field* terjadi pada fase ini. Pemilihan tabel dan *field* tersebut akan dimasukan atau ditransformasikan kedalam *database* yang lain atau *database* baru sebagai bahan atau *data mining* mentah.

### **Modelling**

Pada fase pemodelan dilakukan dengan menggunakan aplikasi R Studio, dan dimasukkan juga metode regresi linier berganda dan algoritma genetika. Data-data yang telah dipilih pada fase *data preparation* digunakan sebagai parameter untuk melakukan peramalan serta optimasi.

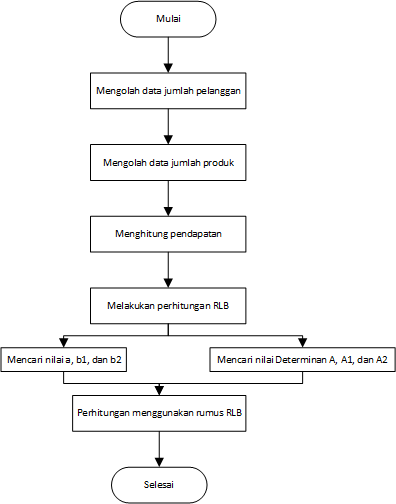
### **Evaluation**

Fase ini merupakan tahapan analisa yang merupakan hasil dari pengolahan fase sebelumnya dengan menginterpretasikan data yang kemudian diperoleh peramalan dan optimasi dari nilai tersebut.

### **Deployment**

Fase ini merupakan pengetahuan atau informasi yang telah diperoleh akan diatur dan dipresentasikan dalam bentuk khusus sehingga dapat digunakan oleh pengguna. Fase *deployment* dapat berupa pembuatan laporan sederhana atau mengimplementasikan proses *data mining* yang berulang dalam perusahaan.

## Diagram Alur Metode Regresi Linier Berganda

****

Gambar 4. 2 Diagram Alur Metode Regresi Linier Berganda.

## Tahapan – Tahapan Diagram Alur Metode Regresi Linier Berganda

Tahapan – tahapan diagram alur metode regresi linier berganda yang digunakan peneliti adalah sebagai berikut:

### **Mengolah Data Jumlah Pelanggana**

Untuk mengolah data jumlah pelanggan ini dapat dilakukan oleh pegawai maupun admin, cara yang dapat dilakukan untuk mengolah data jumlah pelanggan ini dengan mengakumulasikan jumlah pelanggan untuk setiap bulannya dari setiap produk yang berhasil dijual kepada pelanggan. Lalu hasil dari setiap bulannya dijumlahkan kembali untuk mengetahui keseluruhan pelanggan.

### **Mengolah Data Jumlah Produk**

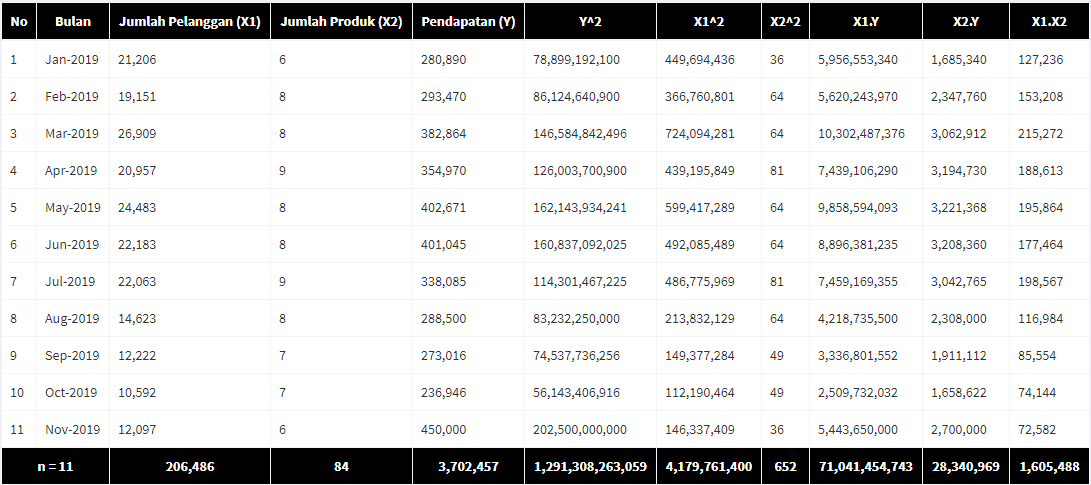
Dalam mengolah jumlah produk, langkah yang dapat dilakukan yakni dengan cara menghitung setiap produk yang terjual setiap bulannya, terhitung bulan Januari.

### **Menghitung Pendapatan**

Untuk melakukan perhitungan pendapatan hal yang perlu diketahui adalah harga untuk setiap produknya. Jika harga sudah diketahui maka tinggal dikalikan saja antara harga produk dikali dengan jumlah pelanggan pada bulan tersebut. Hasilnya diakumulasikan dengan bulan-bulan yang lain, yang selanjutnya digunakan untuk melakukan prediksi pada bulan berikutnya.

### **Melakukan Perhitungan Regresi Linier Berganda**

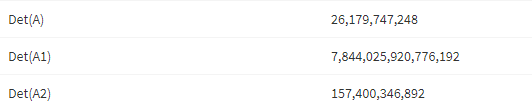
Perhitungan ini dilakukan dengan cara kuadaratkan, dan kalikan dengan nilai pada tabel regresi linier berganda atau dapat dilihat seperti berikut.



Gambar 4. 3 Hasil Perhitungan Tabel RLB.

### **Mencari Nilai Determinan A, A1, dan A2**

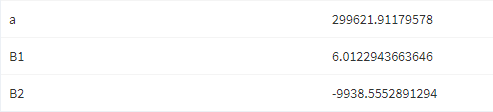
Setelah melakukan perhitungan regresi linier berganda, hal selanjutnya yang perlu kita lakukan ialah mencari determinan A, A1, dan A2 agar mendapatkan nilai untuk melakukan prediksi. Berikut adalah hasil dari determinan A, A1, dan A2.



Gambar 4. 4 Hasil Nilai Determinan.

### **Mencari Nilai A, B1, dan B2**

Untuk mencari nilai a, b1, dan b2. Persyaratan yang perlu dipenuhi adalah nilai determinan A, A1, dan A2 sudah didapatkan, nilai a, b1, dan b2 ini akan digunakan untuk tahap selanjutnya dalam menentukan prediksi atau peramalan, berikut merupakan hasil nilai dari a, b1, dan b2.

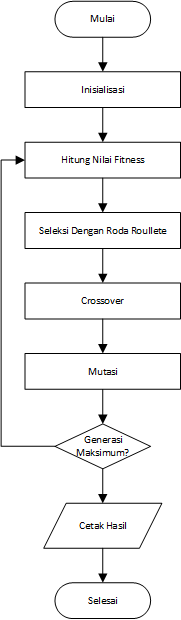


Gambar 4. 5 Hasil Nilai A, B1, dan B2.

### **Perhitungan Menggunakan Rumus Regresi Linier Berganda**

Proses perhitungan menggunakan rumus regresi linier berganda dilakukan jika nilai a, b1, dan b2 sudah didapatkan, karena rumus yang digunakan pada metode regresi linier berganda ini memerlukan nilai a, b1, dan b2 dalam proses peramalan.

## Diagram Alur Algoritma Genetika



Gambar 4. 6 Diagram Alur Algoritma Genetika.

## Tahapan – Tahapan Diagram Alur Algoritma Genetika

Tahapan – tahapan diagram alur algoritma genetika yang digunakan peneliti adalah sebagai berikut:

### **Inisialisasi**

Proses inisialisasi dilakukan dengan cara memberikan nilai awal gen-gen dengan nilai acak sesuai batasan yang telah ditentukan. Sebagai contoh dalam tahap inisialisasi ini jumlah produk dan jumlah pelanggan diberikan inisialisasi identitas atau ID berupa angka 1 sampai jumlah data yang akan diprediksi, sehingga sebuah inisialisasi identitas atau ID berupa angka 1 sampai dengan jumlah data yang akan digunakan adalah gen.

ID untuk nilai X1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 6 | 8 | 2 | 3 | 5 | 7 | 9 | 1 | 4 |

ID untuk nilai X2

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 6 | 10 | 8 | 1 | 3 | 9 | 7 | 5 | 4 |

**Hitung Nilai *Fitness***

Fungsi dari nilai *fitness* suatu kromosom adalah nilai kecocokan kromosom terhadap permasalahan. Semakin tinggi nilai *fitness*, seharusnya solusi tersebut semakin optimal. Berikut merupakan faktor-faktor penunjang nilai *fitness* pada penelitian ini, diantaranya:

1. Jumlah pelanggan pada periode bulan tertentu.
2. Jumlah produk pada periode bulan tertentu.

Pada proses seleksi ini dilakukan dengan cara membuat kromosom yang mempunyai nilai evaluasi kecil, mendapatkan kemungkinan terpilih lebih besar dengan nilai probabilitasnya yang tinggi. Untuk mencari *fitness* rumus yang digunakan sebagai berikut: F = 1 / hasil\_evaluasi.

### **Seleksi Dengan Roda *Roullete***

Seleksi dengan *roullete wheel* memilih anggota populasi tertentu untuk menjadi *parents* dengan probabilitas sama dengan *fitness* dibagi dengan total *fitness* populasi. Pada seleksi ini, *parents* dipilih berdasarkan *fitness*. Jika suatu individu lebih baik, maka akan lebih besar kesempatan terpilih. Probabilitas suatu individu terpilih untuk *crossover* sebanding dengan *fitness*-nya.

Seleksi dengan menggunakan metode *roullete wheel* dilakukan dengan tahapan-tahapan sebagai berikut:

1. Menghitung nilai *fitness* tiap individu.
2. Menghitung nilai probabilitas seleksi setiap individu.

### **Crossover**

Proses *crossover* adalah proses persilangan, yaitu membentuk dua *offspring* (kromosom anak) baru dari dua *parent*. Hanya *offspring* yang memenuhi syarat pada permasalahan yang akan ditambahkan ke populasi yang ada. Proses mutasi yaitu mengganti suatu gen dengan gen yang baru. Proses *crossover* dan mutasi biasanya dilakukan secara acak.

### **Mutasi**

Proses mutasi ini dilakukan setelah proses persilangan atau *crossover* dengan cara memilih kromosom yang akan dimutasi secara acak, dan kemudian menentukan titik mutasi pada kromosom tersebut secara acak. Mutasi diperlukan untuk mengembalikan informasi bit yang hilang akibat pindah silang atau *crossover*. Mutasi diterapkan dengan probabilitas yang sangat kecil. Jika mutasi dilakukan terlalu sering, maka akan menghasilkan individu yang lemah karena konfigurasi gen pada individu yang unggul akan dirusak. Untuk melakukan proses mutasi dapat dilakukan dengan cara sebagai berikut:

Menghitung panjang total gen

Panjang total gen = jumlah gen dalam 1 kromosom \* jumlah kromosom

= 2 \* 6 = 12

Untuk memilih posisi gen yang mengalami mutasi dilakukan dengan memberikan bilangan acak 1 – 12. Misal kita tentukan nilai nm = 20%, maka jumlah gen yang dimutasi adalah 0.2 \* 12 = 2.4 = 2. 2 buah posisi gen yang akan dimutasi, setelah diacak adalah posisi gen 4 dan 7. Lalu gen pada posisi tersebut nilainya diganti dengan ID yang telah ditentukan sebelumnya.

### **Generasi Maksimum**

Proses generasi maksimum ini akan melakukan pengecekan terhadap populasi baru yang telah dibentuk dengan melalui tahapan-tahapan yang cukup panjang. Jika populasi baru yang dihasilkan terbaik, maka selanjutnya akan mengulangi tahapan-tahapan dalam pembentukan populasi baru seperti evaluasi nilai *fitness*, proses seleksi, proses *crossover*, dan mutasi untuk membentuk populasi baru selanjutnya.

### **Cetak Hasil**

Setelah melewati tahapan-tahapan yang cukup panjang, langkah selanjutnya adalah melakukan cetak hasil dari populasi baru yang terbentuk dengan solusi yang optimal untuk menyelesaikan permasalahan optimasi nilai pendapatan pada penlitian ini.

**BAB V**

**IMPLEMENTASI DAN HASIL**

* 1. **Implementasi**

1. **Analisis Permasalahan**

Analisis permasalahan dilakukan berdasarkan penelitian di PT Telkom Indonesia. Dimana terdapat beberapa permasalahan, yaitu diantaranya mengenai strategi pemasaran, peramalan nilai pendapatan per bulannya, dan optimasi nilai pendapatan tersebut. Hal ini dirasa belum efektif dikarenakan masih ada beberapa produk yang kurang diminati oleh pelanggan. Maka dari itu perlunya mengetahui jumlah produk yang terjual tiap bulannya, dan jumlah pelanggan yang berlangganan tiap bulannya sebagai langkah awal untuk menganalisa, memprediksi serta mengoptimasi nilai pendapatan tersebut. Untuk mengetahui nilai hasil peramalan dan nilai optimasi tersebut digunakanlah teknik *data mining* dengan menggunakan Regresi Linier Berganda untuk meramalkan nilai, dan Algoritma Genetika untuk mengoptimasi nilai. Data yang dianalisis merupakan data pelanggan yang didapat dari arsip perusahaan. Kemudian dari data tersebut diambil 3 atribut, yaitu jumlah pelanggan yang berlangganan, jumlah produk, dan harga dari setiap produk. Data tersebut mewakili nilai untuk peramalan dan optimasi pendapatan.

1. **Pengumpulan Data**

Pengumpulan data dilakukan dengan mengambil data dari arsip perusahaan yang berupa data pelanggan yang berada dalam wilayah operasional witel Bandung. Data tersebut berisikan informasi yang diperoleh perusahaan mengenai pelanggan selama berlangganan layanan dari mulai bulan Januari hingga Oktober 2019. Informasi tersebut beberapa diantaranya nomor pelanggan, alamat, tanggal registrasi, tanggal layanan dapat digunakan, tanggal berlangganan, layanan *addOn* yang digunakan, serta harga dari masing-masing layanan *addOn*.

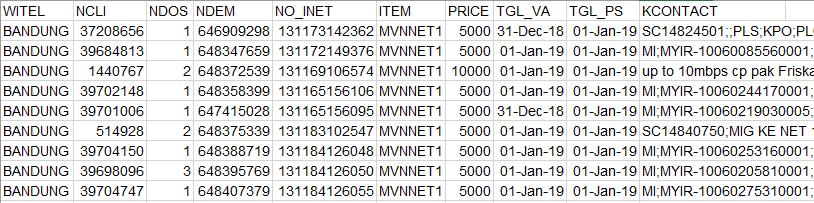
1. **Pengolahan Data Menggunakan Regresi Linier Berganda**
2. **Data Pelanggan**

Pada tahap ini dilakukannya pengumpulan dan penyiapan data pelanggan yang akan dilakukan peramalan dan optimasi. Berikut merupakan data pelanggan yang akan digunakan.

1. Data Pelanggan Movin:

Data pelanggan movin merupakan data pelanggan yang berlangganan *addOn* movin.

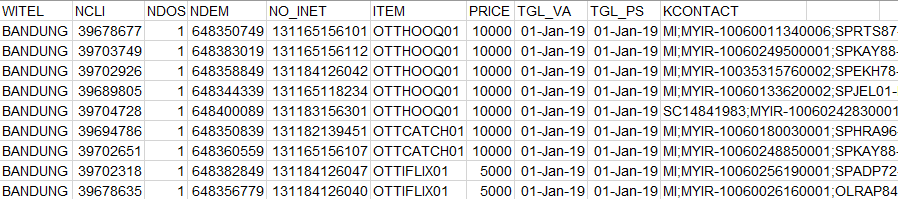
Tabel 5. 1 Data Pelanggan Movin



1. Data Pelanggan OTT

Data pelanggan ott merupakan data pelanggan yang berlangganan *addOn* ott.

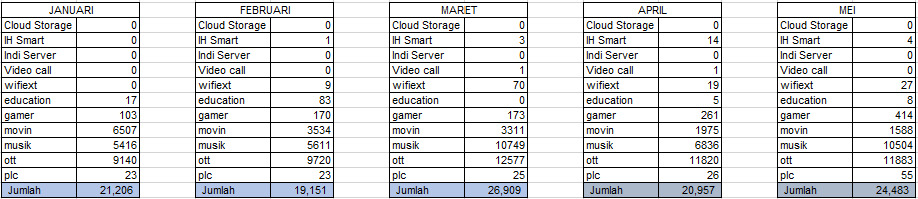
Tabel 5. 2 Data Pelanggan OTT



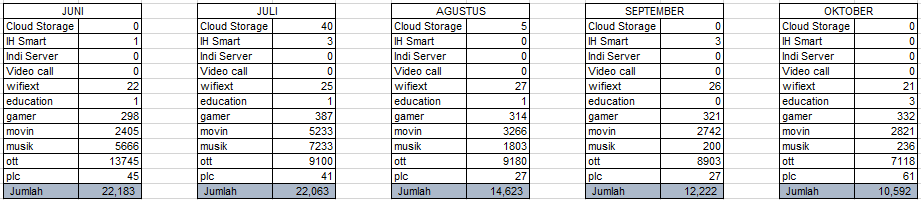
1. **Menjumlahkan Total Pelanggan**

Untuk menjumlahkan total pelanggan, kita harus mengetahui jumlah pelanggan yang berlangganan dari setiap produknya. Berikut merupakan cara yang digunakan untuk menjumlahkan total pelanggan dari setiap produknya.

Tabel 5. 3 Penjumlahan Total Pelanggan

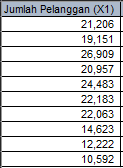


Tabel 5. 4 Penjumlahan Total Pelanggan



Setelah dilakukan penjumlahan seperti tabel tersebut, maka langkah selanjutnya tinggal menyatukan masing-masing jumlah tersebut ke dalam kolom tabel jumlah pelanggan (X1). Berikut hasilnya.

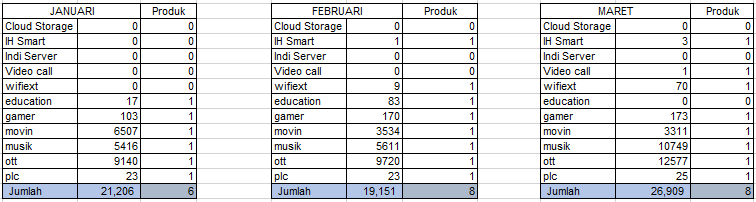
Tabel 5. 5 Jumlah Pelanggan



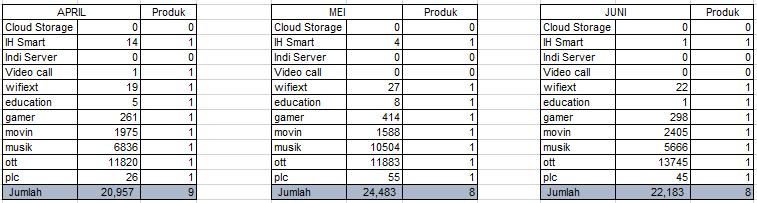
1. **Menjumlahkan Total Produk**

Dalam menentukan jumlah produk pada setiap bulannya, saya melakukan penjumlahan sesuai dengan pelanggan yang berlangganan di setiap bulannya. Berikut hasil dari total produk dari bulan Januari – Oktober.

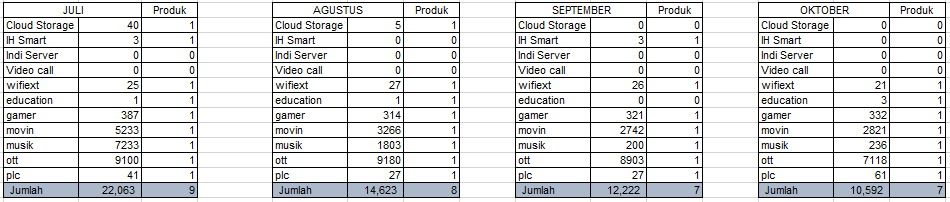
Tabel 5. 6 Penjumlahan Total Produk



Tabel 5. 7 Penjumlahan Total Produk

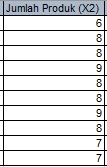


Tabel 5. 8 Penjumlahan Total Produk



Setelah menjumlahkan total produk pada setiap bulannya, selanjutnya kita jadikan menjadi satu kolom tabel total produk (X2). Berikut hasilnya.

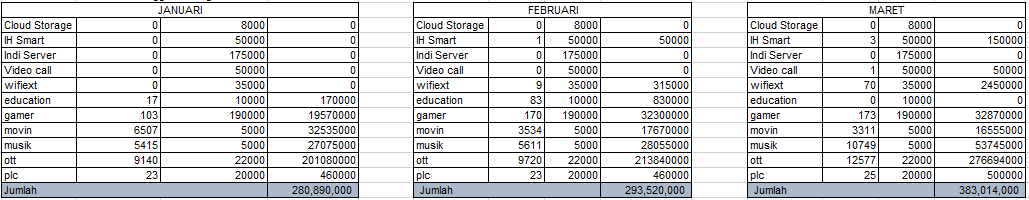
Tabel 5. 9 Jumlah Produk



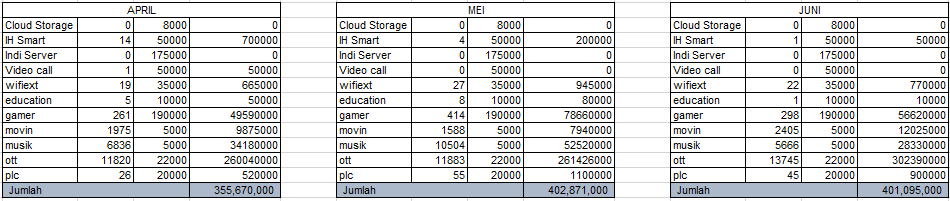
1. **Menjumlahkan Total Pendapatan**

Dalam menentukan jumlah pendapatan ini, hal yang harus diperhatikan ialah jumlah pelanggan dan harga dari masing-masing produk yang dipasarkan. Jika kedua atribut tersebut telah didapatkan, maka selanjutnya kita mengalikan nilai jumlah pelanggan dengan harga dari produk tersebut. Berikut hasilnya.

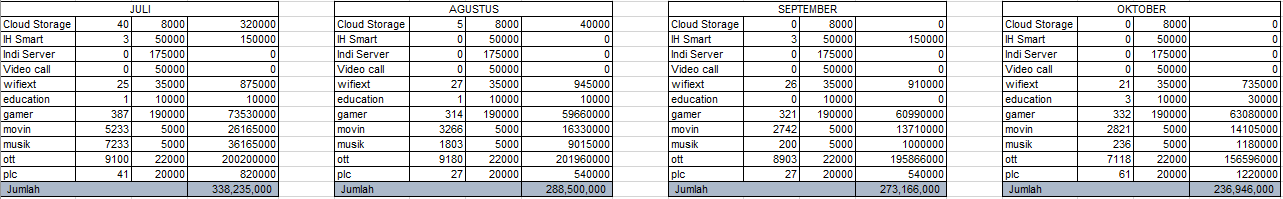
Tabel 5. 10 Penjumlahan Nilai Pendapatan



Tabel 5. 11 Penjumlahan Nilai Pendapatan

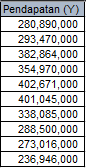


Tabel 5. 12 Penjumlahan Nilai Pendapatan



Setelah menjumlahkan nilai pendapatan dari setiap bulannya, langkah selanjutnya ialah menggabungkan nilai tersebut ke dalam kolom pendapatan (Y).

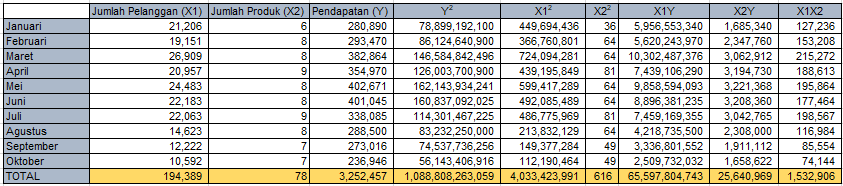
Tabel 5. 13 Hasil Penjumlahan Nilai Pendapatan



1. **Proses Pemangkatan Nilai**

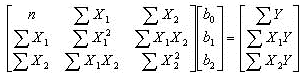
Proses ini dilakukan untuk mencari nilai matrix dan determinan. Proses tersebut diantaranya dengan memangkatkan masing-masing variabel bebas dan variabel terikat. Kemudian, nilai variabel bebas dikalikan dengan variabel terikat. Dan, yang terakhir mengalikan antara variabel bebas.

Tabel 5. 14 Hasil Nilai Setelah Dilakukan Proses Pemangkatan.



1. **Menentukan Nilai Determinan A, B1, dan B2**

Untuk mencari nilai determinan A, B1, dan B2 diperlukan nilai matrix yang didapatkan dari tabel hasil nilai pemangkatan. Proses mencari nilai matrix ini menggunakan rumus sebagai berikut.



Gambar 5. 1 Rumus Matrix

Masukkan nilai hasil pemangkatan pada rumus tersebut untuk mencari nilai determinan A. Sehingga hasilnya akan seperti berikut.

Tabel 5. 15 Nilai Matrix Untuk Mencari Nilai Determinan A



Hasil dari nilai matrix A untuk mencari nilai determinan A ialah sebagai berikut.



Kemudian untuk mencari nilai A, diperlukan nilai matrix A1 dan nilai determinan A1. Dalam menentukan nilai matrix A1 ada langkah yang perlu diperhatikan, diantaranya. Pada kolom pertama (n, ∑X1, dan ∑X2) dirubah menjadi (∑Y, ∑X1Y, dan ∑X2Y). Sehingga hasilnya seperti berikut.

Tabel 5. 16 Nilai Matrix Untuk Mencari Nilai Determinan A1

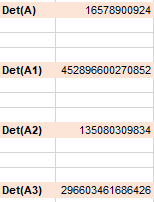


Hasil dari nilai matrix A1 untuk mencari nilai determinan A1 ialah sebagai berikut.



Hal yang sama dilakukan untuk mencari nilai determinan A2, dan A3. Hasil dari nilai determinan tersebut digunakan untuk mencari nilai a, b1, dan b2 yang akan digunakan pada proses perhitungan menggunakan rumus Regresi Linier Berganda. Berikut merupakan hasil nilai determinan yang telah dilakukan.

Tabel 5. 17 Hasil Perhitungan Untuk Mencari Nilai Determinan



Untuk mencari nilai a rumus yang digunakan ialah determinan A1 / determinan A. Begitupun hal yang sama dilakukan untuk mencari nilai b1, dan b2, yakni dengan cara membagi determinan A2 maupun A3 dengan determinan A. Sehingga hasil yang didapatkan ialah sebagai berikut.

Tabel 5. 18 Hasil Untuk Nilai A, B1, dan B2



1. **Menghitung Menggunakan Rumus Regresi Linier Berganda**

Setelah proses-proses tersebut dilewati dan mendapatkan nilai untuk a, b1, dan b2. Selanjutnya tinggal menghitung variabel terikat yang akan diprediksi. Berikut contoh perhitungan dan hasilnya.

Tabel 5. 19 Perhitungan Menggunakan Rumus Regresi Linier Berganda



Disini peneliti mencoba untuk memprediksi nilai pendapatan pada bulan Oktober 2019 dengan nilai X1 = 10,592, dan X2 = 7. Hasil yang didapatkan dari perhitungan tersebut ialah Y = 238,851.

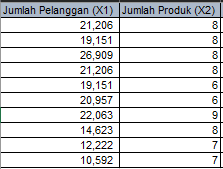
1. **Optimasi Data Menggunakan Algoritma Genetika**

Optimasi ini dilakukan karena nilai *error* yang dihasilkan dari metode Regresi Berganda ini cukup tinggi, sehingga diperlukan optimasi dengan harapan dapat mengurangi nilai *error.* Berikut merupakan tahapan-tahapan yang akan dilakukan dalam proses optimasi menggunakan Algoritma Genetika.

1. **Dataset**

Untuk nilai yang akan dioptimasi diantaranya, jumlah pelanggan dan jumlah produk yang telah diolah pada tahap sebelumnya.

Tabel 5. 20 Dataset Untuk Optimasi



1. **Inisialisasi**

Inisialisasi digunakan untuk membuat kromosom yang dibuat secara *random*, yaitu dengan mengacak nilai gen inisialisasi berupa angka dari jumlah data yang akan digunakan, sehingga didapatkan sebuah kromsom yang urutan gennya acak.

ID untuk nilai X1

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 6 | 8 | 2 | 3 | 5 | 7 | 9 | 1 | 4 |

ID untuk nilai X2

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 6 | 10 | 8 | 1 | 3 | 9 | 7 | 5 | 4 |

Setiap angka tersebut adalah sebuah gen dari kromosom yang telah diberikan identitas sebagai pengganti dari nilai jumlah pelanggan dan jumlah produk. Untuk jumlah gen telah ditentukan sebanyak jumlah nilai yang akan diproses. Kromosom digunakan untuk merepresentasikan suatu solusi yang mungkin dari permasalahan yang akan diselesaikan menggunakan algoritma genetika. Berikut merupakan kromosom yang dibangkitkan atau yang akan digunakan dalam proses optimasi.

Inisialisasi

* Kromosom[1] = [1 4]
* Kromosom[2] = [2 5]
* Kromosom[3] = [3 6]
* Kromosom[4] = [6 3]
* Kromosom[5] = [3 2]
* Kromosom[6] = [4 1]

1. **Evaluasi Nilai Kromosom**

Permasalahan yang ingin diselesaikan adalah nilai variabel X1 dan X2, maka fungsi objektif dapat digunakan untuk mendapatkan solusi. Kita hitung fungsi objektif dari kromsom yang telah dibangkitkan.

* Kromosom[1] = (27 + 8.15\*1 + 17\*4)

= (27 + 8.15 + 68)

= 103.15

* Kromosom[2] = (27 + 8.15\*2 + 17\*5)

= (27 + 17 + 85)

= 129

* Kromosom[3] = (27 + 8.15\*3 + 17\*6)

= (27 + 24.45 + 102)

= 153.45

* Kromosom[4] = (27 + 8.15\*6 + 17\*3)

= (27 + 48.9 + 51)

= 126.9

* Kromosom[5] = (27 + 8.15\*3 + 17\*2)

= (27 + 24.45 + 34)

= 85.45

* Kromosom[6] = (27 + 8.15\*4 + 17\*1)

= (27 + 32.6 + 17)

= 76.6

* Rata-rata nilai dari evaluasi

= (103.15 + 129 + 153.45 + 126.9 + 85.45 + 76.6) / 6

= 112.425

1. **Menghitung Nilai *Fitness***

Pada proses seleksi ini dilakukan dengan cara membuat kromsom yang mempunyai nilai fungsi objektif kecil, mendapatkan kemungkinan terpilih lebih besar dengan nilai probabilitasnya yang tinggi. Untuk itu dapat digunakan fungsi *fitness* = 1 / fungsi\_objektif.

* Fitness[1] = 1 / 104.15

= 0.0096

* Fitness[2] = 1 / 130

= 0.0076

* Fitness[3] = 1 / 154.45

= 0.0064

* Fitness[4] = 1 / 127.9

= 0.0078

* Fitness[5] = 1 / 86.45

= 0.0115

* Fitness[6] = 1 / 77.6

= 0.0128

* Total Nilai *Fitness*

= 0.0096 + 0.0076 + 0.0064 + 0.0078 + 0.0115 + 0.0128

= 0.0557

1. **Mencari Nilai Probabilitas**

Setelah nilai *fitness* ditemukan, selanjutnya mencari nilai probabilitas dengan rumus P[i] = *fitness*[i] / total\_fitness.

* P[1] = 0.0096 / 0.0557

= 0.1723

* P[2] = 0.0076 / 0.0557

= 0.1364

* P[3] = 0.0064 / 0.0557

= 0.1149

* P[4] = 0.0078 / 0.0557

= 0.1400

* P[5] = 0.0115 / 0.0557

= 0.2064

* P[6] = 0.0128 / 0.0557

= 0.2298

Dari probabilitas tersebut terlihat bahwa kromosom ke-6 memiliki *fitness* paling besar, maka kromosom tersebut mempunyai probabilitas untuk terpilih pada generasi selanjutnya lebih besar dari kromosom lainnya.

1. **Mencari Nilai Komutatif**

Untuk proses seleksi digunakan *roulete-wheel*, untuk itu terlebih dahulu mencari nilai komutatif dari probabilitasnya.

C[1] = 0.1723

C[2] = 0.1723 + 0.1364 = 0.3087

C[3] = 0.3087 + 0.1149 = 0.4236

C[4] = 0.4236 + 0.1400 = 0.5636

C[5] = 0.5636 + 0.2064 = 0.77

C[6] = 0.77 + 0.2298 = 1

Proses *roulete-wheel* adalah membangkitkan nilai acak R antara 0-1. Jika R[k] < C[k], maka kromosom ke-k tersebut sebagai induk. Setelah itu putar *roulete-wheel* sebanyak jumlah kromsom.

1. **Membangkitkan Nilai Acak R**

Nilai acak ini dibangkitkan untuk membantu proses terbentuknya populasi baru.

R[1] = 0.201

R[2] = 0.284

R[3] = 0.009

R[4] = 0.822

R[5] = 0.398

R[6] = 0.501

1. **Populasi Baru Terbentuk**

Angka acak R[1] nilainya lebih besar dari C[1] dan lebih kecil dari C[2], maka pilih kromsom [2] sebagai kromosom pada populasi baru. Kromosom baru dari hasil proses seleksi, sebagai berikut.

Kromosom[1] = [2] = [2 5]

Kromosom[2] = [2] = [2 5]

Kromosom[3] = [1] = [1 4]

Kromosom[4] = [5] = [3 2]

Kromosom[5] = [2] = [2 5]

Kromosom[6] = [3] = [3 6]

1. ***Crossover***

Kromsom yang dijadikan induk dipilih secara acak dari jumlah kromsom yang di *crossover* dipengaruhi oleh parameter *crossover* probabilitas, misalnya = 25% = 0,25.

1. Membangkitkan nilai acak R

R[1] = 0.191

R[2] = 0.259

R[3] = 0.760

R[4] = 0.006

R[5] = 0.159

R[6] = 0.340

\*kromosom ke-k yang dipilih sebagai induk. Jika R[k] < nc, maka yang akan dijadikan induk adalah kromosom [1], [4], dan [5].

1. Bilangan acak untuk 3 kromosom induk yang akan di *crossover*

C[1] = 1

C[4] = 1

C[5] = 2

1. Proses *Crossover*

Kromosom[1] = kromosom[1] >< kromosom[4]

= [1 4] >< [6 3]

= [1 6]

Kromosom[4] = kromosom[4] >< kromosom[5]

= [6 3] >< [3 2]

= [3 2]

Kromosom[5] = kromosom[5] >< kromosom[1]

= [3 2] >< [1 4]

= [2 1]

Populasi setelah dilakukan *crossover*

* Kromosom[1] = [1 6]
* Kromosom[2] = [2 5]
* Kromosom[3] = [3 6]
* Kromosom[4] = [3 2]
* Kromosom[5] = [2 1]
* Kromosom[6] = [4 1]

1. **Mutasi**

Menghitung total panjang gen.

Panjang total gen = jumlah gen dalam 1 kromsom \* jumlah kromosom = 2 \* 6 = 12.

Untuk memilih posisi gen yang mengalami mutasi dilakukan dengan memberikan bilangan acak 1 – 12. Misal kita tentukan nilai nm = 20%, maka jumlah gen yang dimutasi adalah 0,2 \* 12 = 2,4 = 2. Dua buah posisi gen yang akan dimutasi, setelah diacak adalah posisi gen 4 dan 7. Lalu gen pada posisi tersebut nilainya diganti dengan ID yang telah ditentukan sebelumnya.

1. Proses mutasi

Kromosom[1] = [1 6]

Kromosom[2] = [2 6]

Kromosom[3] = [3 6]

Kromosom[4] = [1 2]

Kromosom[5] = [2 1]

Kromosom[6] = [4 1]

1. **Menghitung Nilai Fungsi Objektif**

Kromosom[1] = (27 + 8.15\*1 + 17\*6)

= (27 + 8.15 + 102)

= 137.15

Kromosom[2] = (27 + 8.15\*2 + 17\*6)

= (27 + 16.3 + 102)

= 145.3

Kromosom[3] = (27 + 8.15\*3 + 17\*6)

= (27 + 24.45 + 102)

= 153.45

Kromosom[4] = (27 + 8.15\*1 + 17\*2)

= (27 + 8.15 + 34)

= 69.15

Kromosom[5] = (27 + 8.15\*2 + 17\*1)

= (27 + 16.3 + 17)

= 60.3

Kromosom[6] = (27 + 8.15\*4 + 17\*1)

= (27 + 32.6 + 17)

= 76.6

Rata-rata nilai dari evaluasi

= (137.15 + 145.3 + 153.45 + 69.15 + 60.3 + 76.6) / 6

= 106.991

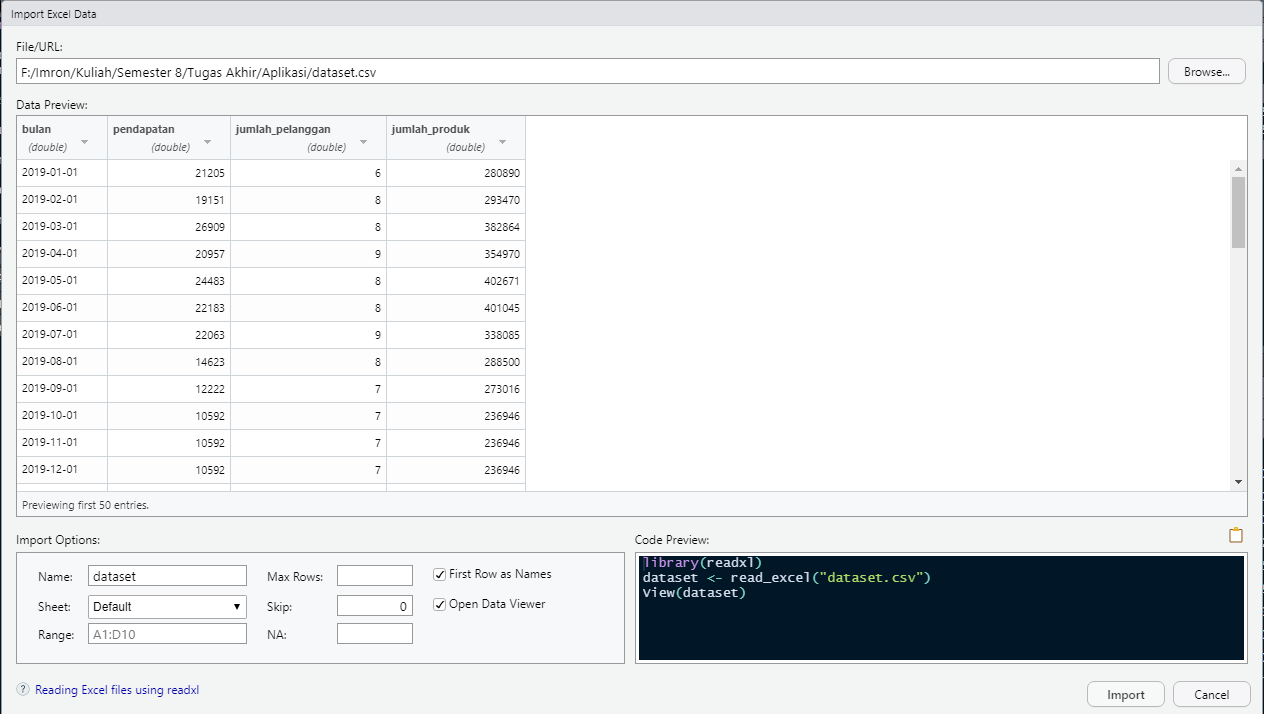
* 1. **Pemodelan Regresi Berganda**

Pemodelan adalah proses untuk membuah model dari sistem. Model simulasi merupakan salah satu bentuk model matematis yang bersifat deskriprif atau prediktif. Simulasi didefinisikan sebagai sekumpulan metode dan aplikasi untuk menirukan atau merepresentasikan perilaku dari suatu sistem nyata, yang biasanya dilakukan pada komputer dengan menggunakan perangkat lunak tertentu. Tujuan suatu pemodelan adalah untuk menganalisa dan memberi prediksi yang dapat mendekati kenyataan sebelum sistem diterapkan di lapangan.

R studio adalah salah satu *Graphic User Interface* (GUI) untuk bahasa pemrograman R, salah satu keunggulan pada R studio ini adalah dapat dijalankan pada browser, maka pengguna tidak memerlukan lagi installasi R, kecuali *package* pemrograman sesuai dengan kebutuhan[1]. Selain itu R studio dapat digunakan untuk merancang, mengembangkan, melaksanakan, dan berkomunikasi dengan kode khusus, R studio mengandung kode komputer dan mempunyai fitur tampilan.

1. ***Import Dataset***

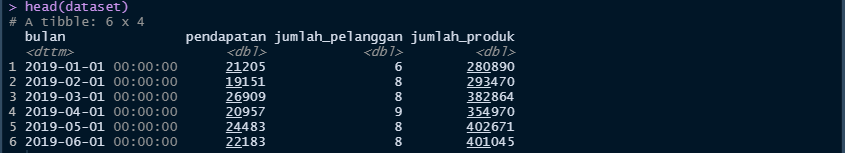
Pada tahap ini penulis melakukan *import dataset* yang akan digunakan dalam proses pemodelan menggunakan metode regresi bergandadan optimasi menggunakan algoritma genetika.



Gambar 5. 2 Import Dataset

1. **Baca Data**

Pada tahap ini melakukan pembacaan data yang telah di *import* pada tahap sebelumnya. Perintah yang digunakan untuk melihat data teratas dengan head(dataset), perintah tersebut dapat menampilkan data dalam bentuk excel dengan format xlsx maupun csv.



Gambar 5. 3 Baca Data Teratas

1. ***Import Module***

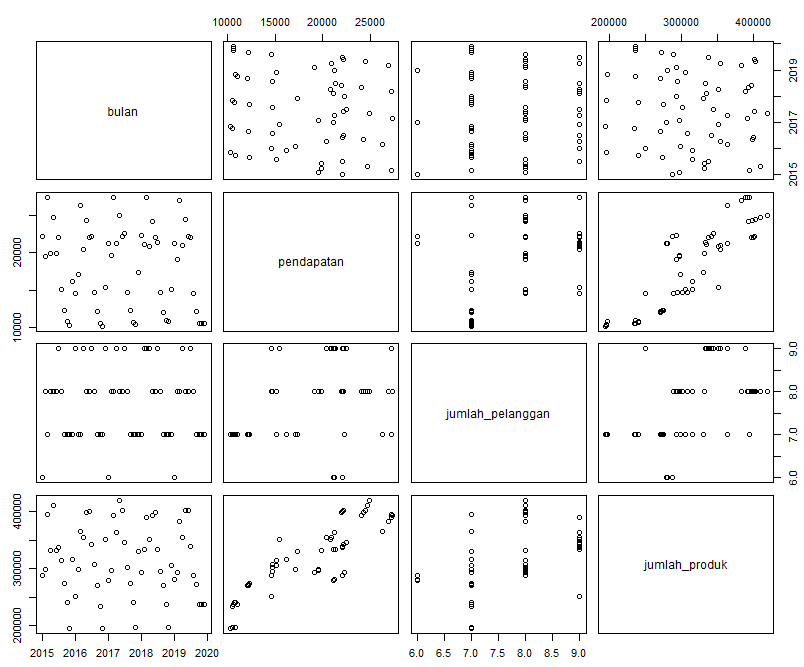
Pada tahap ini penulis melakukan *import module* atau *package* yang digunakan untuk memanggil fungsi yang digunakan selama proses pemodelan, dikarenakan R studio merupakan salah satu aplikasi yang menggunakan bahasa pemrograman R, sehingga *package* yang digunakan merupakan *package* yang berjalan pada bahasa pemrograman R seperti shiny, shinydashboard, lm, ggplot2, DT. *Library* ggplot2 ini berguna untuk menampilkan chart dari proses pengolahan data menggunakan metode regresi berganda.



Gambar 5. 4 Import Module

1. ***Pairs Dataset***

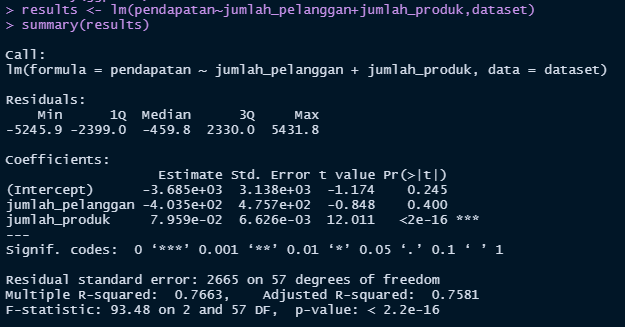
Fungsi *pairs* ini berguna untuk menampilkan korelasi dataset sebelum dilakukan pengolahan. Dengan begitu kita dapat mengetahui variabel mana saja yang saling berkorelasi satu sama lain. Hasil yang ditampilkan berupa *plot chart* sebagai berikut.



Gambar 5. 5 Hasil Pairs Dataset

1. **Perhitungan Menggunakan Metode Regresi Berganda**

Pada tahap ini dataset yang telah di *import* akan dihitung terkait korelasi yang terjadi diantara variabel-variabel tersebut.

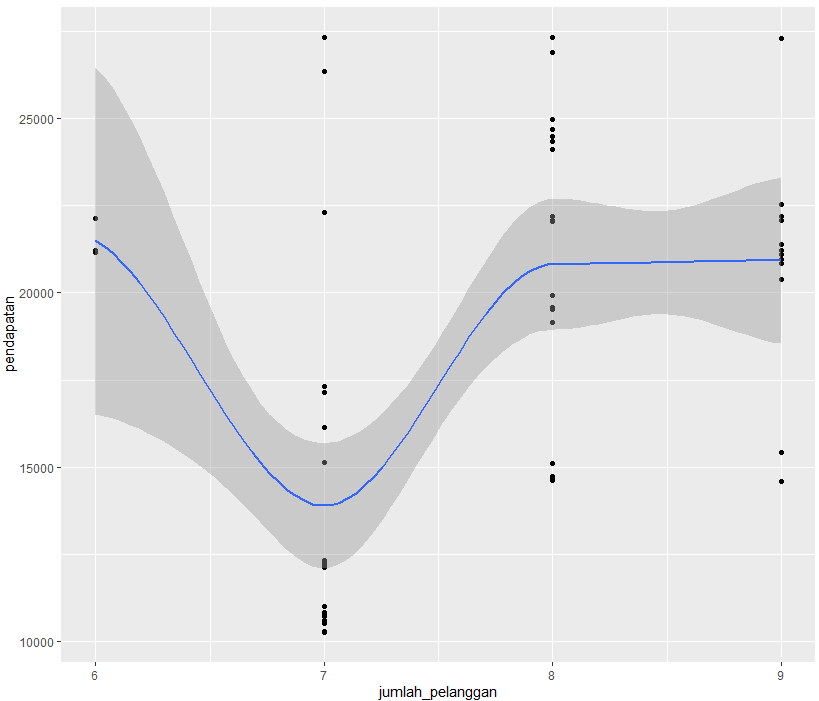


Gambar 5. 6 Perhitungan Metode Regresi Berganda

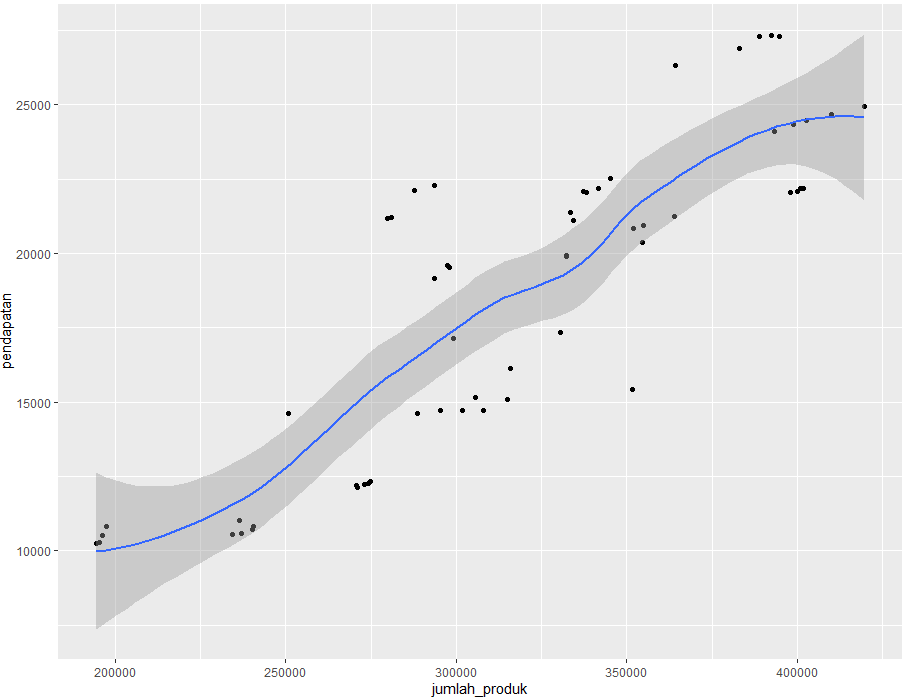
Hasil tersebut cukup baik untuk korelasi antara variabel bebas (jumlah pelanggan dan jumlah produk) terhadap variabel terikat (pendapatan). Karena untuk *Multiple R-squared* yang dihasilkan sebesar 76% dan *Adjusted R-squared* sebesar 75% artinya korelasi tersebut ada, namun belum cukup kuat dikarenakan masih menghasilkan *standard error* yang cukup tinggi.

1. **Plot Hasil Perhitungan**

Tahap ini akan menampilkan plot hasil dari perhitungan yang telah dilakukan pada tahap sebelumnya. Berikut hasil plot untuk jumlah pelanggan dengan pendapatan dan jumlah produk dengan pendapatan.



Gambar 5. 7 Hasil Plot Jumlah Pelanggan dengan Pendapatan



Gambar 5. 8 Hasil Plot Jumlah Produk dengan Pendapatan

* 1. **Pemodelan Algoritma Genetika**

Setelah pemodelan regresi berganda, selanjutnya akan dilakukan pemodelan dengan menggunakan algoritma genetika. Hal ini bertujuan untuk optimasi nilai variabel bebas dari dataset tersebut. Berikut langkah-langkahnya untuk pemodelan menggunakan algoritma genetika.

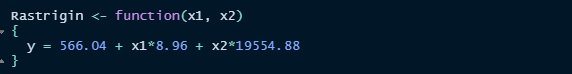
1. ***Import Module***



Gambar 5. 9 Import Module Algoritma Genetika

Pada tahap ini penulis melakukan *import module* atau *library* yang digunakan dalam proses pemodelan menggunakan algoritma genetika. *Library* yang digunakan ini dapat berjalan menggunakan bahasa pemrograman R. *Library* GA ini berfungsi untuk optimasi nilai, namun tidak hanya untuk optimasi, *library* GA ini dapat digunakan sesuai dengan kebutuhan *user* yang menggunakannya.

1. **Mendifinisikan Rumus**

******

Gambar 5. 10 Mendefinisikan Rumus

Tahapan ini digunakan untuk mendefinisikan rumus dan nilai yang akan dilakukan optimasi. Untuk rumusnya menggunakan rumus regresi linier berganda dengan nilai x1 dan x2 yang akan dilakukan optimasi.

1. **Mengurutkan Angka**

****

Gambar 5. 11 Mengurutkan Angka Acak

Fungsi seq() ini akan menghasilkan urutan angka secara acak dengan penjumlahan yang telah ditentukan, pada penelitian ini peneliti membangkitkan urutan angka acak dari -2 sampai 2 dengan penambahan 0.5 untuk setiap angkanya. Penambahan angka ini berfungsi untuk menentukan *values* yang akan dilakukan optimasi pada tahap selanjutnya. Berikut merupakan hasil dari urutan angka yang telah dilakukan.



Gambar 5. 12 Hasil Urutan Angka

1. **Menerapkan Fungsi Array Pada Nilai X1 dan X2**

****

Gambar 5. 13 Function Outer

Fungsi outer() ini berfungsi untuk menerapkan array pada variabel yang akan dilakukan optimasi. Fungsi outer() ini diterapkan pada *values* yang sudah dihasilkan pada tahapan sebelumnya, sehingga hasil dari *function* outer() ini sebagai berikut.



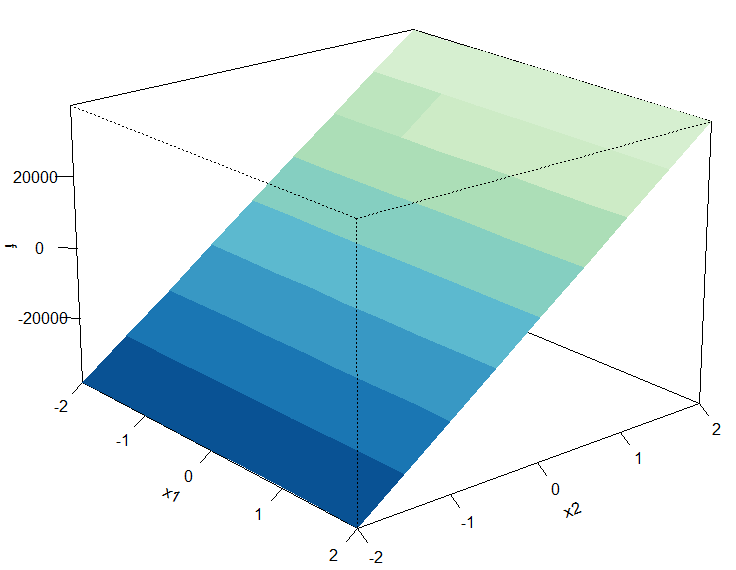
Gambar 5. 14 Hasil Outer

1. **Menampilkan Gambaran Perspektif Nilai X**



Gambar 5. 15 Function Persp

Fungsi persp3D ini berguna untuk menggambarkan plot perspektif nilai X sebelum dilakukannya proses optimasi. Pada tahap ini gambar yang dihasilkan berbentuk 3D. Berikut merupakan gambar yang dihasilkan.



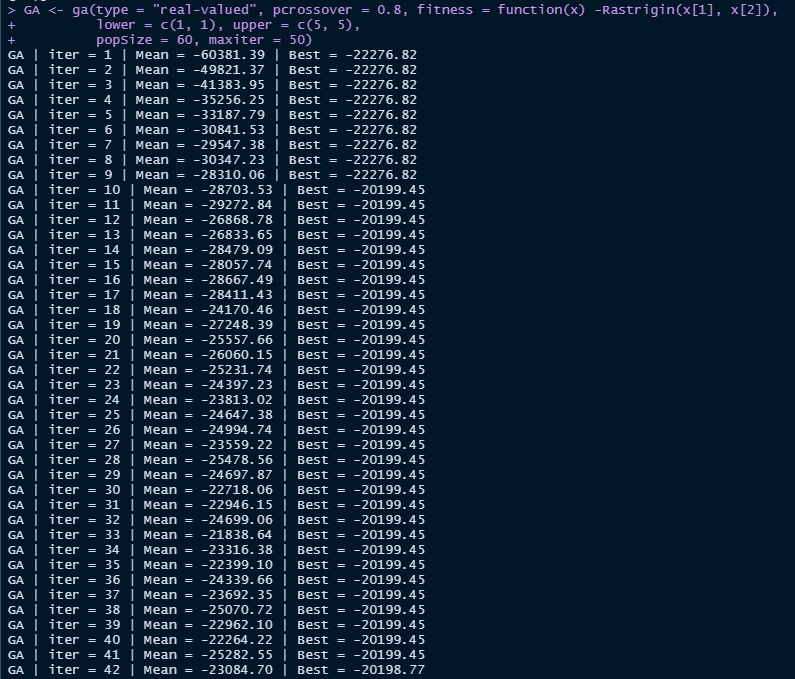
Gambar 5. 16 Hasil Dari Function Persp

1. **Optimasi dan Iterasi**

****

Gambar 5. 17 Proses Optimasi dan Iterasi

Pada penelitian ini jenis pengkodean yang akan digunakan ialah bilangan real karena cocok dengan penelitian ini yakni untuk optimasi nilai. *Pcrossover* berfungsi untuk memberika nilai probabilitas antara kromosom dengan nilai yang diberikan 0.8. *Fitness* ini berguna untuk menentukan solusi terbaik dari sekian solusi yang didapatkan dari proses optimasi. Untuk *lower* dan *upper* ini merupakan nilai batas bawah dan atas dalam proses optimasi. *Popsize* ini merupakan populasi yang dibangkitkan, penelitian ini membangkitkan populasi sebesar 60 dengan maksimal iterasi atau pengulangan sebanyak 50 kali.



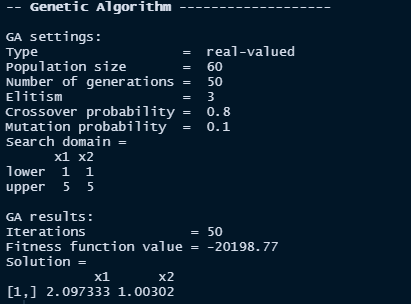
Gambar 5. 18 Proses Iterasi

1. **Summary**

****

Gambar 5. 19 Function Summary

Setelah proses optimasi dan iterasi, maka hasil yang didaptkan ialah sebagai berikut.



Gambar 5. 20 Hasil Dari Optimasi dan Iterasi

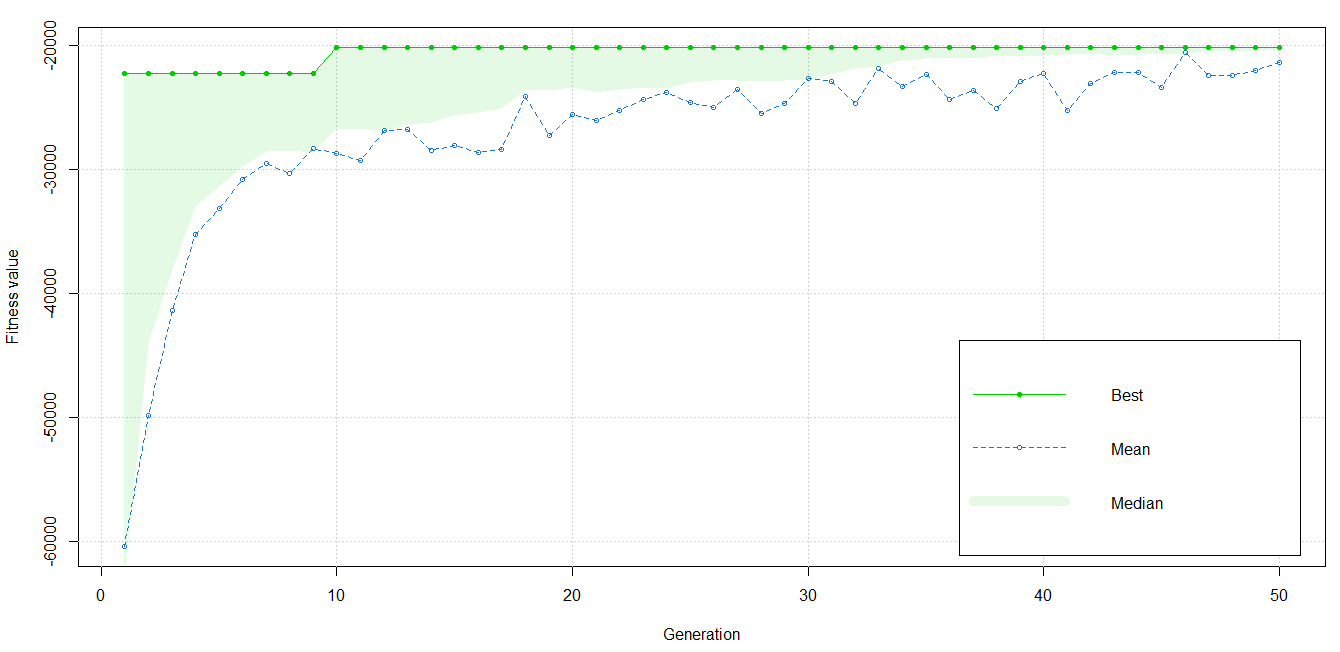
Dari proses iterasi sebanyak 50 kali, mendapatkan nilai solusi x1 = 2.097333 dan x2 = 1.003002. Untuk *elitsm* itu merupakan jumlah individu atau kromosom terbaik yang dapat bertahan pada setiap generasinya atau iterasi yang dilakukan. Penelitian ini kromosom yang dapat berahan dari 50 kali iterasi sebanyak 3 kromosom.

1. **Plot Hasil Optimasi dan Iterasi**

****

Gambar 5. 21 Function Plot

Setelah proses optimasi dan iterasi selesai, selanjutnya kita plot atau kita lihat dalam bentuk chart. Berikut merupakan hasil yang didapatkan.



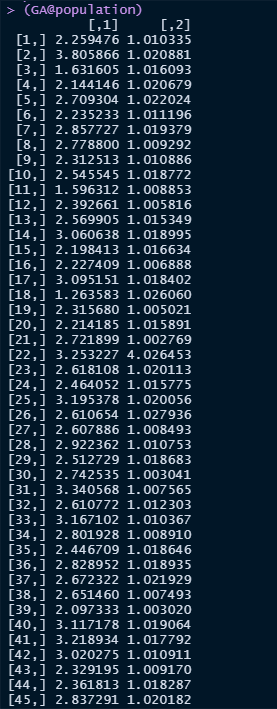
Gambar 5. 22 Hasil Plot

1. **Populasi Baru Terbentuk**

****

Gambar 5. 23 Function Untuk Menampilkan Keseluruhan Populasi

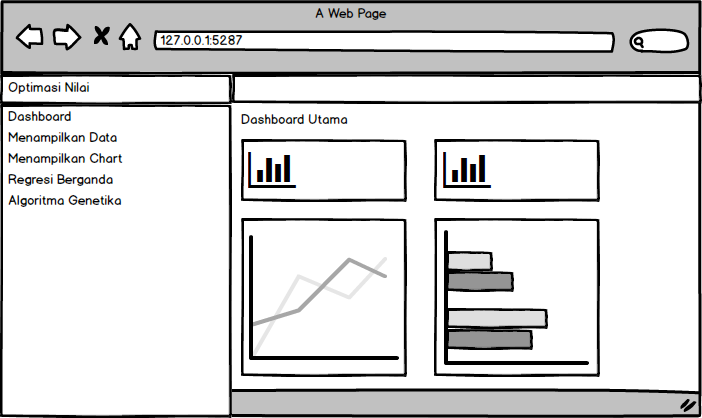
Setelah semua proses dilakukan, berikut merupakan hasil dari keseluruhan solusi yang dihasilkan pada proses optimasi dan iterasi.



Gambar 5. 24 Keseluruhan Nilai Populasi Yang Terbentuk

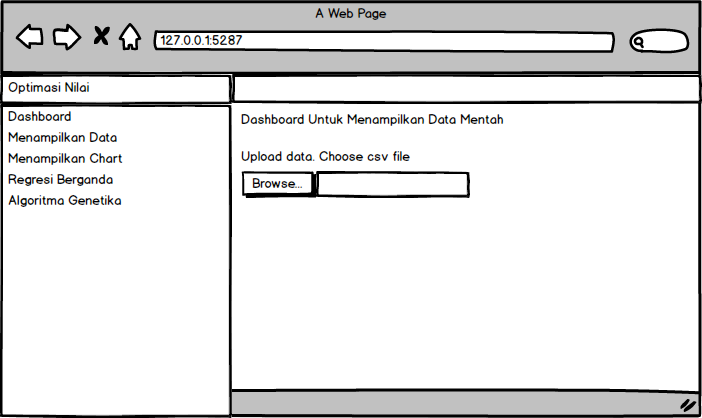
* 1. ***User Interface***

1. *Dashboard* Utama



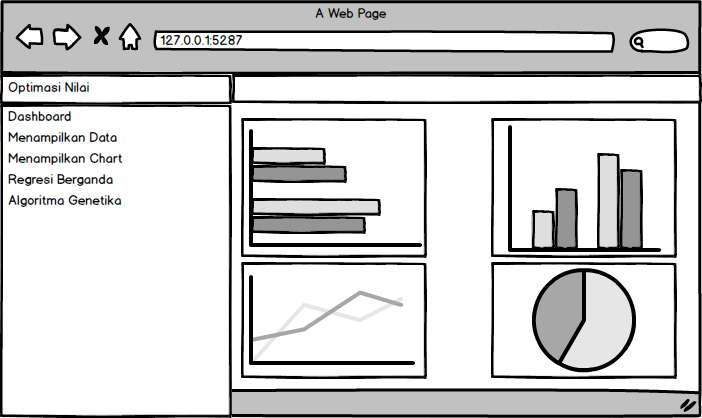
Gambar 5. 25 Dashboard Utama

1. Menampilkan Data



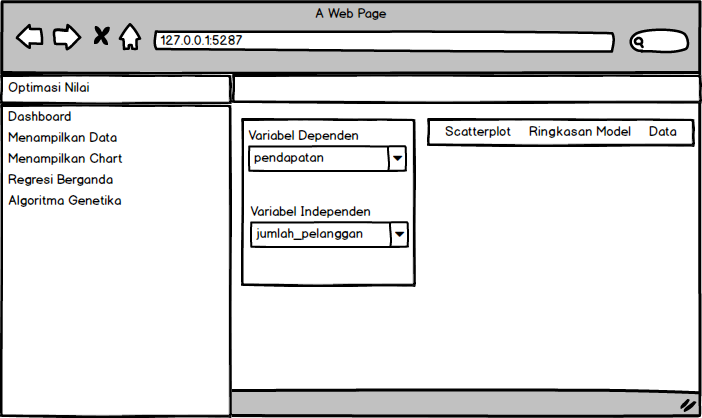
Gambar 5. 26 Halaman Menampilkan Data

1. Menampilkan Chart



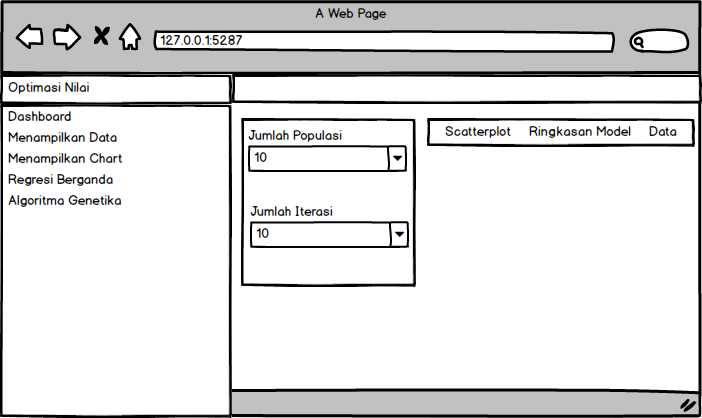
Gambar 5. 27 Halaman Menampilkan Chart

1. Regresi Berganda



Gambar 5. 28 Halaman Regresi Berganda

1. Algoritma Genetika



Gambar 5. 29 Halaman Algoritma Genetika