



**PENERAPAN METODE JARINGAN SYARAF TIRUAN UNTUK
MERAMALKAN KAPASITAS LAYANAN PADA JASA
PERAWATAN SEPATU (STUDI KASUS SEPATU BERSIH)
SKRIPSI**

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Sandy Ikhsan Armita
NIM: 125150207111044



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019**



PENGESAHAN

PENERAPAN METODE JARINGAN SYARAF TIRUAN UNTUK MERAMALKAN
KAPASITAS LAYANAN PADA JASA PERAWATAN SEPATU

(STUDI KASUS SEPATU BERSIH)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Sandy Ikhsan Armita

NIM: 125150207111044

Skripsi ini telah diuji dan dinyatakan lulus pada

2 Agustus 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing 2

Nurul Hidayat, S.Pd, M.Sc

Ratih Kartika Dewi, S.T., M.Kom

NIP: 19680430 200212 1 001

NIK: 201503 890520 2 001

Mengetahui

Ketua Jurusan Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D

NIP: 19710518 200312 1 001



PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 1 Mei 2019

Sandy Ikhsan Armita

NIM: 125150207111044



PRAKATA

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa, dimana telah melimpahkan rahmat nya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini dengan judul “Penerapan Metode Jaringan Syaraf Tiruan Untuk Meramalkan Kapasitas Layanan Pada Jasa Perawatan Sepatu (Studi Kasus Sepatu Bersih)”. Dimana skripsi merupakan salah satu syarat untuk memenuhi persyaratan dalam menyelesaikan studi diprogram studi Sarjana Teknik Informatika Universitas Brawijaya.

Keadaan skripsi ini tidak lepas dari bimbingan serta arahan dari berbagai pihak. Oleh karena itu di kesempatan kali ini penulis ingin mengucapkan rasa terimakasih dan mengucapkan rasa hormat sebesar besarnya kepada :

1. Nurul Hidayat, S.Pd, M.Sc selaku dosen pembimbing satu dan Ratih Kartika Dewi, S.T., M.Kom selaku dosen pembimbing dua yang telah membimbing dan memberikan pengarahannya dengan sabar dan bijaksana, dan penyampaian ilmu, meberikan masukan serta saran dalam pengerjaan skripsi ini.
2. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
3. Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya beserta jajaran yang telah mempermudah proses birokrasi.
4. Seluruh Dosen dan katyawan/karyawati fakultas Ilmu Komputer Universitas Braijaya atas dukungan dan kerjasamanya selama ini.
5. Kepada kedua orang tua penulis, yang telah memberikan dukungan motivasi dan materiil. Serta semua do’a dalam kelancaran kuliah serta skripsi penulis.
6. Semua pihak yang telah membantu dan berbagi ilmu dalam penyelesaian skripsi, yang tidak dapat penulis sebutkan satu-persatu.

Semoga segala pertolongan dan kebaikan mendapat berkah dan balasan dari Allah SWT. Dengan kerendahan hati penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, oleh karena itu kritik dan saran yang sifatnya membangun guna memperbaiki penulisan selanjutnya. Akhir kata, penulis berharap bahwa skripsi ini dapat bermanfaat bagi segala pihak.

Malang, 1 Mei 2019

Penulis

sandyarmita@gmail.com



ABSTRAK

Sandy Ikhsan Armita, Penerapan Metode Jaringan Syaraf Tiruan Untuk Meramalkan Kapasitas Layanan Pada Jasa Perawatan Sepatu (Studi Kasus Sepatu Bersih)

Pembimbing: Nurul Hidayat S.Pd, M.Sc dan Ratih Kartika Dewi S.T., M.Kom

Jasa perawatan sepatu merupakan bentuk pengembangan dari usaha jenis laundry konvensional dengan teknis yang lebih spesifik. Bentuk usaha ini tergolong baru karena sebelumnya orang hanya merawat sepatu dengan alakadarnya seperti hanya sebatas dicuci. Sebenarnya potensi pasar untuk jenis usaha ini sangatlah menjanjikan mengingat sepatu merupakan hal umum yang digunakan sehari-hari untuk beraktifitas dan menunjang penampilan. Jenis usaha ini mulai muncul diawal tahun 2009 dan hanya melalui media online seperti kaskus, hingga pada awal tahun 2012 jenis usaha ini mulai menjamur seiring dengan pasar yang semakin paham jika merawat sepatu perlu peralatan dan bahan khusus. Lambat lahun karena semakin meningkatnya permintaan maka muncul persaingan antar penyedia jasa ini, banyak usaha yang akhirnya harus bangkrut dan tutup karena gagal melakukan strategi-strategi bisnis, salah satunya adalah meramalkan pemasukan jasa. Dari masalah tersebut, ada sebuah metode yang dapat diimplementasikan dalam penyelesaian permasalahan tersebut yaitu dengan menggunakan metode Jaringan Syaraf Tiruan. Dengan memperhitungkan proses dari backpropagation pada prosesnya dan menghasilkan nilai MSE terendah untuk menentukan keakuratan dari prediksi yang dihasilkan.

Kata kunci: Sistem, Sepatu, Backpropagation, Prediksi, MSE, akurasi



ABSTRACT

Sandy Ikhsan Armita, The Implementation of Artificial Neural Network to Predict the Service Capacity of Shoe Treatment Service (Case Study of Sepatu Bersih)

Supervisors: Nurul Hidayat S.Pd, M.Sc dan Ratih Kartika Dewi S.T., M.Kom

Shoe treatment business is a sub business from developing conventional laundry with more complex and specific technic. This kind of business is a new one for the market, because all they know how to treat their shoe is just by washing it regularly. Actually the market potential for this business is promising, because shoe is daily essential for everyone's activity and of course it's part of the appearance to make a better looks. This business was known by the market in the early 2009, it was started by some accounts offering their service on the forum such as Kaskus, till the early 2012 this business has a bigger market due to the education of the people. The technic, tools, and stuff for shoe treatment had to be advance. By the time because of gaining demand of the market, the competition in this business is getting real. There so many business had to close too soon because cant do the strategical business plan, one of it was how to predict the service capacity. Therefore from that problem there is a method that could be useful for predicting the service capacity, using the Artificial Neuron Network method. By counting the process from the backpropagation and looking for the smallest output of MSE to decide the most accurate number of the prediction.

Keyword: System, Shoe, Backpropagation, Prediction, MSE, accuracy

**DAFTAR ISI**

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
PRAKATA	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
BAB 1 PENDAHULUAN	33
1.1 Latar belakang.....	33
1.2 Rumusan masalah	34
1.3 Tujuan.....	34
1.4 Manfaat.....	35
1.5 Batasan masalah	35
1.6 Sistematika pembahasan	35
BAB 2 LANDASAN KEPUSTAKAAN	37
2.1 Tinjauan Pustaka	37
2.2 Sepatu	40
2.3 Laundry Sepatu	40
2.4 Kecerdasan Buatan.....	41
2.5 Jaringan syaraf tiruan.....	41
2.6 Arsitektur Jaringan	42
2.7 Backproagation	44
2.8 Pelatihan Standar Backpropagation.....	45
2.9 Pengujian MSE.....	47
BAB 3 METODOLOGI	48
3.1 Studi literatur	49
3.2 Analisis Kebutuhan.....	49
3.3 Pengumpulan Data.....	49
3.4 Perancangan Sistem.....	50
3.5 Implementasi.....	51
3.6 Pengujian Sistem.....	51



3.7 Analisis.....	51
3.8 Kesimpulan.....	51
BAB 4 PERANCANGAN	52
4.1 Jaringan Syaraf Tiruan.....	52
4.2 Backpropagation	52
4.3 Perhitungan Manual.....	67
4.3.1 Contoh perhitungan manual proses backpropagation untuk data latih	67
4.3.2 Contoh perhitungan manual proses backpropagation untuk data uji. 75	
4.4 Perancangan Antarmuka.....	77
4.4.1 Halaman Utama	77
4.4.2 Halaman Data Latih.....	78
4.4.3 Halaman Backpro-train	79
4.4.4 Data Uji.....	80
4.4.5 Halaman Backpro-Test	81
4.4.6 Halaman Denormalisasi data	82
BAB 5 IMPLEMENTASI.....	83
5.1 Implementasi Program.....	83
5.1.1 Inisialisasi awal	83
5.1.2 Normalisasi Data	87
5.1.3 Proses Data Latih.....	87
5.1.4 Proses Data Uji	88
BAB 6 PEGUJIAN DAN ANALISIS	93
6.1 Sistematika Pengujian	93
6.2 Analisis dan Pembahasan.....	93
6.3 Pengujian dan Analisis Jumlah Neuron.....	93
6.4 Pengujian dan Analisis Jumlah Hidden Layer.....	94
6.5 Pengujian dan Analisis Dalam Jumlah Iterasi.....	95
6.6 Analisis Hasil Pengujian.....	97
BAB 7 KESIMPULAN DAN SARAN.....	98
7.1 Kesimpulan.....	98
7.2 Saran.....	98
DAFTAR PUSTAKA	99



DAFTAR TABEL

Tabel 2.1 Tinjauan Pustaka	38
Tabel 4.1 Inisialisasi bobot awal jaringan.....	67
Tabel 4.2 bobot jaringan dari lapisan masukan ke lapisan tersembunyi.....	68
Tabel 4.3 bobot dari lapisan tersembunyi ke lapisan <i>output</i>	68
Tabel 4.4 Suku perubahan bobot ke unit tersembunyi	71
Tabel 4.5 Unit tersembunyi yang mengalami perubahan bobot.....	74
Tabel 4.6 Perubahan bobot di unit tersembunyi.....	74
Tabel 4.7 Inisialisasi data uji.....	75
Tabel 4.8 bobot jaringan dari lapisan <i>input</i> ke lapisan tersembunyi.....	75
Tabel 4.9 bobot dari lapisan tersembunyi ke lapisan <i>output</i>	75
Tabel 4.10 Dari hasil prediksi berdasarkan data uji propagasi balik.....	76
Tabel 4.11 Denormalisasi Data	77
Tabel 6.1 Hasil Uji penjumlahan tiap-tiap neuron	94
Tabel 6.2 Hasil Pengujian pada tiap-tiap Hidden Layer	95
Tabel 6.3 Hasil pengujian nilai jumlah pada Iterasi	96



DAFTAR GAMBAR

Gambar 2.1 Jaringan dalam Layar Jamak (Siang, 2009).....	43
Gambar 2.2 Fungsi Aktivasi dalam Biner Sigmoid.....	44
Gambar 2.3 Grafik Fungsi Bipolar Sigmoid Dalam Pengaktifasiannya	45
Gambar 3.1 Blok metode penelitian peramalan kapasitas layanan	48
Gambar 3.2 Diagram Blok Sistem Peramalan Kapasitas Layanan Sepatu	50
Gambar 3.3 Diagram Blok Pengujian Akurasi Peramalan Kapasitas layanan	51
Gambar 4.1 Fungsi Aktivasi dalam Sigmoid Biner.....	53
Gambar 4.2 Fungsi Aktivasi dalam Sigmoid Bipolar.....	53
Gambar 4.3 Diagram Alir penghitungan nilai pada z_{net_j}	55
Gambar 4.4 Diagram Alir penghitungan nilai pada z_j	56
Gambar 4.5 Diagram Alir penghitungan nilai pada y_{net_k}	57
Gambar 4.6 Diagram Alir penghitungan nilai pada y_{net_k}	58
Gambar 4.7 Diagram Alir Peghitungan Nilai pada θ_k	59
Gambar 4.8 Diagram Alir penghitungan nilai pada Δw_{kj}	60
Gambar 4.9 Diagram Alir penghitungan nilai pada θ_{net_j}	61
Gambar 4.10 Diagram Alir penghitungan nilai pada θ_k	62
Gambar 4.11 Diagram Alir penghitungan ilai pada Δv_{ji}	63
Gambar 4.12 Diagram Alir penghitungan nilai pada w_{kj} (<i>baru</i>).....	64
Gambar 4.13 Diagram Alir Penghitungan nilai pada v_{ji} (<i>baru</i>).....	65
Gambar 4.14 Diagram Alir Backpropagation	66
Gambar 4.15 Antarmuka Pada Halaman Utama.....	77
Gambar 4.16 Antarmuka Data latih	78
Gambar 4.17Antarmuka backpro-train.....	79
Gambar 4.18 Antramuka Data Uji.....	80
Gambar 4.19 Antramuka backpro-test	81
Gambar 4.20 Antarmuka pada denormalisasi data	82
Gambar 6.1 Hasil Uji Pada Jumlah tiap-tiap Neuron	94
Gambar 6.2 Hasil pengujian pada tiap-tiap Hidden Layer	95



BAB 1 PENDAHULUAN

1.1 Latar belakang

Sepatu dalam praktiknya di akhir dekade ini sudah sangat menjadi bagian dari apa yang disebut fashion. Sejak dulu hingga beberapa waktu terakhir produsen sepatu membuat dari berbagai jenis material mulai kulit hingga kayu, hingga akhirnya pada 2010 terdapat terobosan yaitu merubah bahan karet serta plastik untuk sepatu dan berbagai jenis bahan lainnya yang berasal dari petro (Wolley,2008). Berbagai jenis bahan sepatu yang beragam disamping fungsinya sebagai alat pelindung karena memiliki kualitas serta bahan yang baik, disamping kenyamanan menjadi aspek utama sepatu harus dirawat agar memiliki umur yang lama dan terbebas dari hal yang menyebabkan penyakit yang bersarang pada sepatu. Hal yang paling sulit dalam merawat sepatu adalah ketidak tahuan dari pemilik sepatu umumnya karena agar sepatu tidak mudah rusak saat pertama kali dirawat (Motawi, 2015).

Dari yang didapat dalam Asosiasi Persepatuan Indonesia nilai impor sepatu dalam kurun 2013-2014 mengalami peningkatan yang cukup signifikan yaitu US 45 juta dollar. Pelaku usaha atau produsen sepatu dalam negeri tak mau kalah dengan para produsen yang berasal dari luar negeri, perkembangan usaha sepatu di dalam negeri pun ikut berkembang dengan turut andil menyumbang sebesar 23%, kebutuhan sepatu dalam negeri juga mengikuti tren positif yaitu terus mengalami peningkatan produksi setiap tahunnya sebesar 1.2%. Hal tersebut ditengarai dipengaruhi oleh tingkat pertumbuhan ekonomi Indonesia yang dinilai cukup kondusif setiap tahun berkisar antara 4,5% hingga 5,3% dalam kurun waktu 2014 hingga 2016 (Azry , 2017).

Pada awalnya bidang pemasaran salah satu yang menjadi masalah utama dari usaha sepatu bersih. Karena awamnya pasar terhadap jenis usaha ini, pembuatan strategi pemasaran yang jelas segmen dan tata caranya adalah hal utama yang harus dijadikan fokus oleh para pelaku usaha laundry sepatu, agar dapat dikenal dan menaikkan omzet (Rofik, 2018).

Semakin menjamurnya jasa perawatan sepatu persaingan pun mau tak mau tidak terelakan. Agar dapat menciptakan daya saing dalam dunia perawatan sepatu, para pelaku usaha harus dapat memberikan *value* lebih, kunci utama dari jasa perawatan sepatu adalah waktu dan hasil. Karena semakin singkat waktu pengerjaan dengan kualitas pengerjaan yang baik akan berbanding lurus dengan perkembangan usaha perawatan sepatu itu sendiri. Maka dari itu prediksi jumlah *customer* atau layanan di hari berikutnya menjadi sangat berguna agar tidak terjadi hambatan lamanya waktu laundry sepatu yang dikarenakan jumlah pelanggan yang melebihi kapasitas pelayanan.

Terdapat satu dari banyak metode Jaringan syaraf tiruan yang dapat digunakan untuk meramalkan kapasitas layanan sepatu di hari berikutnya. Berbagai hubungan yang tidak sejajar dan faktor ekonomi yang sudah sangat



beragam serta variasi berbagai faktor dapat diukur dan diselesaikan oleh metode JST. JST dapat digunakan untuk mengukur hubungan berbagai faktor dalam peramalan (Setiawan, 2008). Untuk menentukan kapasitas layanan sepatu di kemudian hari diperlukan proses perencanaan serta pengambilan keputusan yang baik. Pada umumnya multi layer atau banyak lapisan adalah bentuk atau ciri dari propagasi balik dengan harapan dapat mengurangi kesalahan saat dilakukannya penghitungan yang dilakukan oleh jaringan. Terdapat langkah demi langkah dalam propagasi balik atau backpropagation itu sendiri diantaranya memasukkan data kedalam jaringan *input* (feedforward), nilai dalam eror dan nilai ketika masukan dihasilkan lalu melalui proses pembaharuan. Begitu nilai bobot dan juga bias terhitung, jaringan sudah siap menerima berbagai nilai masukan. (Triyono, 2016) .

Pada kasus sebelumnya pola seri waktu telah dijalankan Badrul Anwar (2011) jauh hari sebelum ini, dari situ beliau menghasilkan nilai akurasi cukup tinggi yaitu 94% dalam penerapannya untuk prediksi suku bunga bank. Finky Dona dan Vincent Suhartono adalah 2 orang berikutnya yang menggunakan *Backpropagation* untuk memprediksi saham(2013), dari hasil percobaan yang dilakukan beberapa kali oleh mereka, *output* dengan akurasi tinggi didapatkan yaitu 95,93%.

Dari seluruh penyampaian yang berisi berbagai penelitian, maka penulis melihat celah dan masalah yang bisa dikembangkan yaitu “penerapan metode jaringan syaraf tiruan untuk meramalkan kapasitas layanan pada jasa perawatan sepatu (studi kasus sepatu bersih)”. Dari hasil keseluruhan sistem peramalan jasa perawatan sepatu ini diharap membantu usaha laundry sepatu untuk memprediksi jumlah layanan untuk hari berikutnya. Sistem menyediakan masukan berupa *history* jumlah layanan setiap harinya untuk diproses, sehingga perhitungan jumlah layanan sepatu dapat diprediksi pada hari yang akan datang.

1.2 Rumusan masalah

Berdasarkan masalah-masalah yang sudah disampaikan mengenai jasa perawatan sepatu, maka sebelum meramalkan masalah kapasitas jasa perawatan sepatu, ditarik rumusan masalah:

1. Bagaimana dalam sistem prediksi jumlah layanan laundry sepatu dapat digunakan dengan metode jaringan syaraf tiruan.
2. Bagaimana hasil akhir dalam pengujian sistem prediksi jumlah layanan laundry sepatu dengan metode jaringan syaraf tiruan.

1.3 Tujuan

Hal akhir yang diinginkan untuk tercapai pada penelitian mengenai peramalan sepatu:

1. sistem dapat memprediksi jumlah layanan laundry sepatu dari data riwayat layanan pada outlet sepatu bersih.



2. Menguji tingkat akurasi dalam sistem peramalan jumlah kapasitas layanan sepatu dengan menggunakan metode jaringan syaraf tiruan (JST).

1.4 Manfaat

Hal yang nantinya diharap dari sistem prediksi jumlah kapasitas layanan sepatu ini adalah dapat mempersiapkan sepatu yang akan masuk dikemudian hari sehingga usaha laundry dapat mengantisipasi dan memperkirakan jumlah layanan dikemudian hari.

1.5 Batasan masalah

Agar permasalahan dalam peramalan jumlah kapasitas layanan sepatu tidak melebar dan memiliki batasan-batasan yang dapat diterapkan dalam penelitian, maka ditentukan:

1. Sistem peramalan layanan jasa perawatan sepatu ini menggunakan jumlah layanan laundry sepatu pada sepatu bersih.
2. Hasil daripada sistem peramalan kapasitas layanan ini berupa jumlah sepatu pada hari berikutnya.
3. Bahasa Program yang digunakan dalam peramalan kapasitas layanan sepatu yang dipakai adalah JAVA
4. Bentuk pengujian sistem peramalan sepatu menggunakan pengujian akurasi.
5. Metode yang dipakai pada penelitian peramalan jumlah kapasitas layanan sepatu ini merupakan *backpropagation*.

1.6 Sistematika pembahasan

Bentuk urutan yang dipergunakan pada peramalan jumlah kapasitas layanan sepatu yang dipergunakan adalah:

BAB I PENDAHULUAN

Dalam permulaan atau awal dari peramalan kapasitas layanan ini terdapat alasan yang dimuat pada latar belakang, dan perumusan masalah, lalu tujuan dari apa yang ingin dicapai pada penelitian, manfaat yang didapat dari penelitian, beberapa batasan-batasan masalah, urutan bagaimana skripsi ini ditulis yang mendasari sistem peramalan jumlah layanan sepatu pada Sepatu Bersih.

BAB II TINJAUAN PUSTAKA

Sekumpulan teori yang mendasari sistem peramalan kapasitas layanan sepatu yang berhubungan langsung dengan penelitian ini juga dasar teori mengenai permasalahan layanan perawatan dan laundry sepatu, Jaringan syaraf tiruan, juga berbagai dasar hal lain yang bisa dijadikan acuan penelitian ini.

BAB III METODOLOGI PENELITIAN



Seluruh hal mengenai proses juga metodologi mengenai sistem peramalan kapasitas layanan, perancangan dalam sistem juga analisa akan dijelaskan pada bagian ini. Bentuk-bentuk metodologi yang dipakai pada sistem peramalan ini adalah studi literatur dari berbagai sumber, pengumpulan data langsung di kantor tempat bisnis ini beroperasi, analisis berbagai bentuk kebutuhan, perancangan dari bagaimana urutan pembentukan sistem dimulai, penerapan dari rancangan yang dibuat, pengujian dan analisis pada sistem yang sudah diuji untuk diambil kesimpulan.

BAB IV PERANCANGAN SISTEM

Kebutuhan-kebutuhan untuk merancang model dari sistem peramalan kapasitas layanan akan dibahas dengan merujuk pada metode yang dipakai.

BAB V IMPLEMENTASI

Penerapan sistem peramalan kapasitas layanan sepatu dari bentuk rancangan yang selesai dibuat sebelumnya.

BAB VI PENGUJIAN DAN ANALISIS

Hasil yang akan diambil dari sistem peramalan jumlah kapasitas layanan melalui uji dari sistem itu sendiri serta dilanjutkan analisa keakuratan dari keluaran sistem.

BAB VII PENUTUP

Penarikan dari hasil sistem peramalan kapasitas layanan berupa kesimpulan dan kekurangan yang terdapat demi pengembangan mengenai penelitian ini kedepannya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Tinjauan Pustaka

Dalam peramalan kapasitas layanan sepatu dibutuhkan kajian-kajian yang berkaitan dengan permasalahan untuk dijadikan dasar dalam penelitian. Pada sistem peramalan layanan ini akan digunakan berbagai rujukan untuk menjadikan dasar atau acuan dari beberapa penelitian tersebut ditunjukkan di tabel 2.1 yang tertera.

Dasar yang paling awal untuk dijadikan acuan dalam peramalan kapasitas layanan sepatu ini adalah identifikasi bentuk-bentuk penyimpangan pada anak-anak yang sudah diteliti 2018 kemarin oleh Fadhilla Puji. Dikarenakan jenis penyimpangan yang beragam pada anak-anak maka perlu dilakukan pengenalan sejak dini. 38 masukan neuron digunakan dalam penelitian ini, 5 neuron tersembunyi, dan 3 keluaran neuron. Penyimpangan dalam tumbuh kembang anak dapat dilihat dalam penelitian ini memiliki akurasi cukup tinggi yaitu 91.11% dengan 81 data latih, 9 data uji, batas kesalahan 0,0009, dan learning rate 0,1.

Prediksi permintaan jumlah semen merupakan penelitian selanjutnya yang dilakukan oleh Mahendro Agni Giri Pawoko dengan menggunakan *Backpropagation*. Penelitian ini memiliki tujuan untuk melakukan penghitungan prediksi permintaan semen dalam beberapa kurun waktu kedepan, supaya dapat mengeluarkan aturan atau regulasi lebih cepat sehingga mempersingkat waktu untuk penyesuaian kapasitas produksi tanpa harus menghentikan produksi yang sudah berjalan. Banyak cara dapat digunakan dalam prediksi beberapa jenis kasus, *backpropagation* merupakan salah satu yang dapat digunakan karena mempunyai *output* prediksi yang baik, seperti halnya prediksi yang telah diceritakan di atas yaitu jumlah semen yang diminta dalam kurun waktu tertentu dan produksi gula. 6 masukan neuron dan 4 neuron tersembunyi serta 1 keluran neuron. Learning rate bernilai 0,8 merupakan parameter terbaik, 1-4 merupakan interval dari bobot awal serta nilai iteraksi maksimum sebesar 200. 0,049064 merupakan nilai dari nilai MSE terbaik sebagai bentuk prediksi yang bagus.

Pengelompokan harga jual beli dari minyak sawit merupakan salah satu topik yang dinilai juga bisa dijadikan rujukan, topik ini dibahas oleh Dwi FR. Dalam tulisannya pertimbangan pembelian minyak kelapa sawit oleh pasar dipengaruhi oleh naik atau turunnya harga minyak yang ada. Dalam penelitian ini mengupas permasalahan klasifikasi harga minyak kelapa sawit dengan metode yang sama yakni *Backpropagation*. Harga minyak aka dimodelkan dalam jangka 5 bulan sebelumnya untuk mencari tahu hasil di bulan ke 6. Hasil yang dikeluarkan memiliki tingkat akurasi cukup baik yaitu 69,57% dengan neuron tersembunyi sebanyak 50, nilai dari proses *learning rate* pun sebesar 0,1 serta jumlah maksimum iterasi diketahui sebanyak 70.000.

Penelitian untuk dijadikan landasan peramalan kapasitas layanan yang selanjutnya adalah mengenai peramalan profit yang didapat dari ternak ayam



Alabio, telah dilakukan oleh M. Ihsan Diputra di Kalimantan Selatan. Dengan berlandaskan jaringan syaraf tiruan sistem ini dapat mengkalkulasi peramalan dari profit telur ayam alabio. Algoritma tersebut sering digunakan dikarenakan penerapannya sudah sering diaplikasikan pada jenis sistem peramalan. Berbagai subsistem atau biasa disebut fitur inputan yang digunakan pada peramalan profit telur ayam alabio ini antara lain jumlah babon atau biasa disebut induk, bibit yang menyusut, harga total pakan, penyusutan kandang, biaya buruh, dan vaksin serta obat yang digunakan pada ayam alabio itu sendiri. dari sistem tersebut *output* yang diharapkan oleh peneliti adalah berupa keuntungan bersih atau profit yang didapat dari telur ayam alabio. Pengujian wajib dilakukan dalam melakukan penelitian karena memiliki tujuan utama yaitu nilai parameter yang paling baik atau optimal. Nilai parameter tersebut berupa penilaian *learning rate* sebesar 0,8, lalu jumlah *neuron* yang didapat pada hidden layer sebesar 17, limitasi daripada *MAPE* latih sebesar 10%, dan sebanyak 90% data latih. Nilai *MAPE* peramalan teroptimal yang didapatkan sebesar 25,7852%.

Penelitian tentang jeruk siam menjadi rujukan yang terakhir dalam rujukan mengenai sistem peramalan kapasitas layanan sepatu. Bahasan dalam penelitian ini adalah penggunaan dosis pupuk dengan metode backpropagation berdasar lingkungan tanaman juga karakteristiknya. Peramalan kondisii pupuk menjadi bahasan utama dengan mengukur kesamaan dan ekosistemnya. Penelitian lingkungan tanaman yang dilihat dari dosis beserta karakteristik pupuk ini menggunakan Jaringan Saraf Tiruan (JST) propagasi balik. Arsitektur yang diterapkan dalam JST terdiri dari tiga '*lapisan masukan saraf*' yang mewakili lebar kanopi, bentukan tanah serta teksturnya dan intensitas dari curah hujan, 1 lapisan yang tersembunyi atau biasa dinamakan dengan *hidden layer*, dan 3 syaraf atau juga *neuron* merepresentasikan komposisi pupuk yakni berupa nitrogen, fosfor, dan kalium. dari peramalan yang dilakukan pada buah jeruk adalah 3 syaraf masukan dengan 5 lapisan tersembunyi dan 3 syaraf keluaran. sehingga menghasilkan rancangan arsitektur peramalan terbaik.

Tabel 2.1 Tinjauan Pustaka

No	Judul	Tujuan	Metode	Hasil
1.	Identifikasi Penyimpangan Tumbuh Kembang Anak Dengan Algoritme <i>Backpropagation</i>	Pengelompokan gejala yang terjadi dan nampak pada anak untuk mengidentifikasi jenis penyimpangan	Metode Backpropagation	Dalam data latih berjumlah 81, didapat rerata akurasi 91.11%, 9 data uji didapat, 01 <i>learning rate</i> , dan batasan nilai <i>error</i> 0,0009.
2.	Prediksi Jumlah Permintaan Semen Menggunakan	untuk melakukan penghitungan	Metode Backpropagation	Dari perhitungan nilai MSE terbaik yang dapat



	Jaringan Syaraf Tiruan <i>Backpropagation</i>	prediksi permintaan semen yang terjadi dalam waktu kedepan, supaya dapat mengeluarkan aturan atau regulasi lebih cepat sehingga mempersingkat waktu untuk penyesuaian kapasitas produksi		digunakan adalah sebesar 0,049064.
3.	Implementasi Metode <i>Backpropagation</i> Untuk Klasifikasi Kenaikan Harga Minyak Kelapa Sawit	Penulisan ini membahas perihal klasifikasi harga mengenai minyak kelapa sawit	Metode propagasi balik	Hasil penghitungan klasifikasi yang didapatkan memiliki tingkat akurasi cukup baik 69,57%
4.	Prediksi Tingkat Keuntungan Usaha Peternakan Itik Alabio Petelur menggunakan Jaringan Syaraf Tiruan <i>Backpropagation</i> (Kasus di Kabupaten Hulu Sungai Utara Kalimantan Selatan)	penelitian ini digunakan jaringan syaraf tiruan <i>backpropagation</i> untuk melakukan peramalan profit yang didapat dari ternak ayam petelur alabio.	Metode <i>Backpropagation</i>	Nilai <i>MAPE</i> peramalan terbaik yang didapat sebesar 25,7852%.
5.	Peramalan Dosis Pupuk Berdasarkan Karakteristik dan Lingkungan Tanaman Jeruk Siam Menggunakan Metode	Mengitung jumlah dosis pupuk yang digunakan sesuai ekosistem dan karakteristkik jeruk	Metode <i>Backpropagation</i>	Nilai dari evaluasi presentasi rerata dari data komposisi dosis pupuk adalah 9.178% yang nitrogen, fosfor, dan kalium



	<i>Backpropagation</i>			didapat dari kesalahan perhitungan rerata dosis pupuk.
--	------------------------	--	--	--

2.2 Sepatu

Sepatu saat ini telah jauh berkembang menjadi bagian dari *fashion* dan tidak hanya menjadi tren, namun sudah lebih ke arah alat penunjang aktifitas yang juga dapat menunjukkan status sosial maupun budaya mengenai sepatu. Pada perkembangan sepatu, sepatu kerap dibuat dari berbagai bahan seperti kulit hingga kanvas, tetapi hampir satu dekade yang lalu para produsen mulai bereksperimen dalam membuat sepatu dari berbagai bahan seperti sepatu berbahan karet, plastik, dan bahan-bahan lainnya (Wolley,2008). Banyak sekali bahan dasar atau material sepatu seperti mesh dan knit, sepatu pada harfiahnya sebagai alat untuk melindungi bagian tubuh bawah tepatnya kaki, juga membutuhkan perhatian dan perawatan yang baik dan sesuai, karena dapat memperpanjang umur juga menjaga agar kondisi sepatu tetap bersih. Karena tepatnya orang dengan pengidap penyakit gula sangat rentan dengan sepatu yang tidak higienis karena bersifat sensitif terhadap kuman. Jenis penanganan dan perawatan sepatu tidak mudah karena perlu peralatan khusus agar tidak mudah rusak (Motawi, 2015).

2.3 Laundry Sepatu

Ada banyak sekali peluang bisnis yang bisa dijalankan pada era belakangan ini, khususnya bagi para kawula muda. Berbagai kisah dari muda mudi yang membuat pencapaian dari dunia bisnis sangatlah memotivasi. Di Indonesia sendiri peluang bisnis sudah mengalami capaian yang cukup baik serta signifikan dengan predikatnya sebagai negara berkembang, bukanlah hal yang naif jika negeri ini memiliki peluang yang sangat luas untuk mengurangi atau bahkan menyudahi masalah perekonomian dan bisnis yang buruk. Bukanlah hal yang dianggap sebelah mata jika menumbuhkan potensi dan bibit-bibit kawula muda untuk terus melakukan inovasi serta turut menciptakan gebrakan, terutama perihal bisnis.

Belakangan ini, pelaku jasa laundry konvensional dan anak-anak muda menaruh perhatian pada bisnis jasa perawatan sepatu. Terlebih saat sepatu sudah menjadi barang umum bagi semua kalangan. Terlebih ketika munculnya komunitas-komunitas yang menggunakan sepatu sebagai aspek utama seperti pemain bola, pemain basket, dan mahasiswa. Karena tuntutan zaman yang semakin cepat seringkali hal-hal yang mungkin bersifat sepele tidak jarang justru



malah menjadi hal yang menjanjikan, terlebih memiliki pengalaman dan keahlian khusus di bidangnya atau biasa disebut profesional (Indarto, 2015).

Jasa perawatan sepatu merupakan bentuk dari jasa atau layanan yang menyediakan berbagai macam layanan sesuai dengan peruntukannya. Tak jarang masyarakat banyak yang tidak mengetahui dengan benar bagaimana merawat sepatu mereka dengan benar karena tidak mengetahui juga tidak memiliki peralatan yang sesuai dengan sepatu yang dimiliki.

Namun sekarang jasa perawatan sepatu dapat dijadikan solusi bagi mereka yang memang memiliki sepatu yang beragam dan membutuhkan perawatan sesuai dengan peruntukannya. Peralatan dan bahan untuk merawat sepatu pun tidak boleh sembarang karena harus memiliki kriteria khusus seperti pH sabun rendah dan sikat bulu kuda untuk material tertentu. Untuk alat yang digunakan pun juga mesin-mesin yang memang diperuntukkan untuk proses pengerjaan sepatu.

Hadirnya jasa perawatan sepatu, terdapat nilai lebih yang bisa didapat oleh masyarakat sekitar sebagai tempat untuk merawat atribut yang bersifat menunjang penampilan dan aktifitas sehari-hari juga memiliki lebih banyak waktu untuk mengerjakan hal yang diinginkan karena dapat menghemat waktu dan juga tenaga. Selain hal-hal tersebut para pelanggan juga akan mendapatkan ekstra *dustbag* untuk penyimpanan sepatu yang baik dari Sepatu Bersih.

2.4 Kecerdasan Buatan

Entitas buatan yang dapat melakukan dan menunjukkan kecerdasan bisa disebut dengan *Artificial Intelligence* atau AI. Komputer adalah benda atau bahkan hal yang paling sering disebut sebagai benda yang memiliki AI. Komputer diberi *input* dengan berbagai bentuk kecerdasan dengan harapan bisa serta mampu melakukan berbagai macam hal seperti manusia.

Pembubuhan kecerdasan pada benda-benda seperti handphone dan komputer dinilai mampu untuk membantu manusia dalam menangani urusan-urusannya. Sebuah perhitungan cerdas yang dapat diakui atau dianggap sebuah premis dari sebuah bidang studi adalah AI (Motawi, 2015).

2.5 Jaringan syaraf tiruan

Jaringan yang terbentuk pada syaraf biologis ternyata memiliki kesamaan atau kemiripan dalam fungsinya dengan karakteristik yang dimiliki oleh jaringan syaraf tiruan (Siang, 2009). Bentuk umum dari model jaringan biologi yang dipadukan dengan matematika adalah Jaringan Syaraf Tiruan itu sendiri, menggunakan perumpamaan:

1. Terdapat berbagai elemen sederhana yang bisa menghasilkan informasi pada tiap neuron.



2. Pengiriman sinyal dikirimkan oleh berbagai penghubung ke setiap neuron
3. Bobot antar neuron turut berpengaruh dalam kuat lemahnya sinyal.
4. Dalam penentuan nilai keluaran, fungsi yang kerap digunakan adalah aktivasi untuk pengenaanya pada masukan yang telah diterima. Batas ambang digunakan dalam perbandinganya dengan besarnya nilai keluaran.

Penentuan terhadap Jaringan Syaraf Tiruan dapat ditentukan oleh :

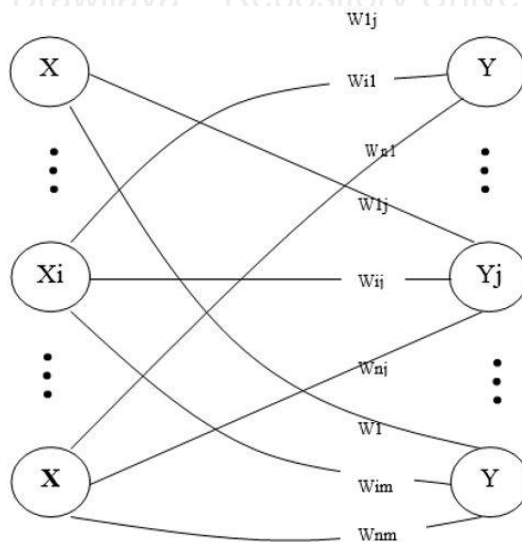
1. Pattern yang terbentuk antar sesama neuron.
2. Penentuan metode yang dipakai dalam penentuan bobot penghubungnya.
3. Fungsi aktivasi

2.6 Arsitektur Jaringan

Dari apa yang sudah disebut oleh Siang pada penelitiannya, terdapat bentuk jaringan yang sering digunakan, diantaranya:

1. Jaringan dalam Lapisan Tunggal (*Single Layer Network*)

Penyambungan neuron secara langsung dengan sejumlah *output*. Hanya didapati satu neuron ourput dalam perceptron.



Gambar 2.1

Jaringan dalam Layer Tunggal (Siang, 2009)

gambar pola jaringan diatas merupakan bentuk dari unit input n (X) dan unit *output* m (Y)

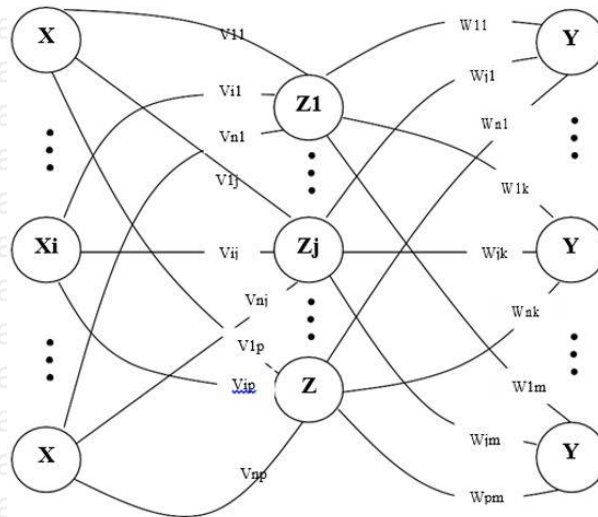


Penghubungan seluruh unit baik unit masukan atau keluaran dihubungkan secara rata pada Jaringan dalam lapisan tunggal, sekalipun bobot yang dimiliki sangat jauh berbeda. Penghubungan jaringan dalam lapisan tunggal tidak memasangkan antar sesama, baik sesama masukan maupun sesama nilai keluaran.

Pernyataan bilangan atau nilai bobot hubungan antar masukan dan keluaran diwakilkan dalam bentuk huruf w . Didalam pelatihan jaringan dalam lapisan tunggal terdapat bobot inilah yang independen satu sama lain. Modifikasi bobot perlu dilakukan pada jaringan dalam lapisan tunggal guna hasil akurasi yang baik dikarenakan sifat dari jaringan dalam lapisan tunggal yang sederhana, maka jenis jaringan ini cocok untuk digunakan pada pengenalan pola.

2. Jaringan dalam Layar Jamak

Jaringan dalam layar tunggal memiliki bentuk perluasan yaitu jaringan dalam lapisan jamak. Pada jaringan dalam layar jamak tidak hanya ada unit masukan dan keluaran saja, terdapat *layer* atau lapisan yang tersembunyi. Bahkan *layer* atau lapisan yang tersembunyi dapat lebih dari satu. Sama pada jaringan dalam layar tunggal bahwa penghubungan jaringan dalam layar jamak tidak memasangkan antar sesama, baik sesama nilai keluaran dan sesama nilai masukan.



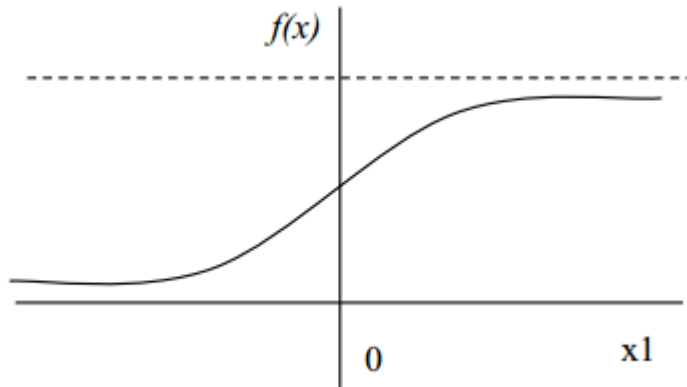
Gambar 2.1 Jaringan dalam Layar Jamak (Siang, 2009)

Pola jaringan dalam layar jamak diatas merupakan rangkaian dari unit *input* n (X) lalu lapisan tersembunyi (Z) dan unit keluaran yang diwakilkan dengan (Y). Kelebihan jaringan dalam layar jamak dapat mengurai masalah ataupun perhitungan yang lebih rumit dibanding dengan jaringan dalam layar tunggal, Kelemahan pada jaringan dalam layar jamak adalah prosesnya yang tidak secepat jaringan dalam layar tunggal.



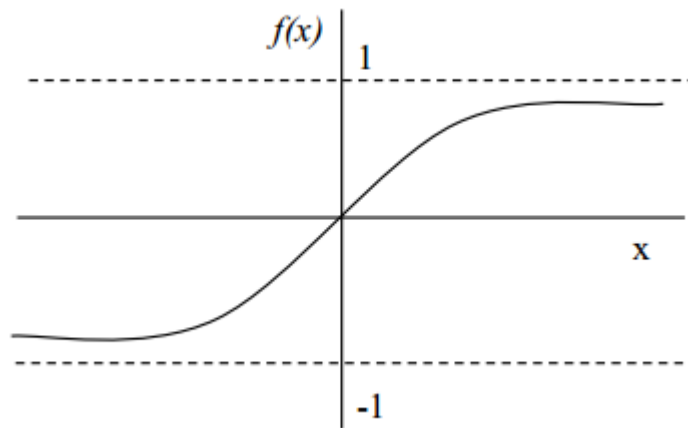
2.7 Backpropagation

Syarat-syarat yang harus terpenuhi pada fungsi aktivasi backpropagation adalah: bersambung, ditinjau dari fungsi tidak memiliki kecenderungan penurunan, serta mudah untuk terdiferensial. Ketika 3 syarat tersebut terpenuhi maka akan digunakan range 0,1 dalam fungsi sigmoid dalam biner. Pada $f(x) = \frac{1}{1 + e^{-x}}$ dengan turunan $f'(x) = f(x)(1 - f(x))$. Grafiknya dapat dilihat pada gambar 2.3 (fungsi aktivasi dalam sigmoid biner).



Gambar 2.2 Fungsi Aktivasi dalam Biner Sigmoid

Karena keidentikan antara fungsi dari bipolar sigmoid dan biner sigmoid membuat keduanya sering dipakai, dengan digunakannya range (-1,1). Dengan bentuk turunan berupa $f'(x) = \frac{f(x)(1-f(x))}{2}$ maka $f(x) = \frac{2}{1 + e^{-x}} - 1$. Grafik fungsinya terlihat pada gambar 2.4 (Grafik Aktivasi dalam fungsi Bipolar Sigmoid).



Gambar 2.3 Grafik Fungsi Bipolar Sigmoid Dalam Pengaktivasiannya

'1' adalah nilai maksimum yang dapat dimiliki oleh fungsi sigmoid. Perubahan pola dalam nilai masukan dan keluaran harus dirubah dengan syarat atau catatan targetnya melebihi 1. Untuk tercapainya range yang sama seperti fungsi yang digunakan dalam sigmoid. Selain lapisan keluaran atau *output* dapat digunakan aktivasi dalam sigmoid. Fungsi identitas $f(x) = x$ merupakan bentuk fungsi dari aktivasi yang dijalankan pada *output* atau bisa disebut keluaran.

2.8 Pelatihan Standar Backpropagation

Tahapan atau urutan dalam proses propagasi balik ada tahap bergerak maju, propagasi bergerak mundur, serta propagasi berbalik mundur, dan bobot yang dapat berubah-ubah. 1 lapisan tersembunyi yang menerapkan aktivasi dalam biner sigmoid memiliki algoritma pelatihan seperti berikut:

Langkah 0: Bilangan-bilangan anak kecil menginisialisasi seluruh nilai dalam bobot

Langkah 1: Langkah 2-8 harus diulang dalam kondisi penghentian masih tidak terpenuhi

Langkah 2: Data pelatihan wajib menjalankan tahap 3-8 dalam tiap-tiap pasangannya.

Tahap I : Propagasi Maju

Langkah 3: Sinyal masukan diterima oleh setiap unit lapisan tersembunyi yang berada di bawahnya.

Langkah 4: layer tersembunyi dihitung tiap-tiap nilai keluarannya

$$z_j (j = 1, 2, \dots, p) \\ z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (2.1)$$



$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (2.2)$$

Langkah 5: Setiap network pada unit $y_k (k = 1, 2, \dots, m)$ wajib dihitung

$$y_{net_k} = v_{k0} + \sum_{j=1}^p z_j w_{kj} \quad (2.3)$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} \quad (2.4)$$

Tahap II : Propagasi mundur

Langkah 6: Kesalahan pada masing-masing *output* $y_k (k = 1, 2, \dots, m)$ dihitung nilai keluaranya di faktor yang bernilai ϑ

$$\vartheta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (2.5)$$

ϑ_k terdapat nilai bobot yang berada di atas nilai eror atau salah adalah bentuk unit eror yang dipergunakan pada pergantian dalam bobot lapisan yang ada di bawahnya (langkah 7)

Laju dengan percepatan pada α digunakan oleh w_{kj} dalam menghitung perubahan suku di bobotnya.

$$\Delta w_{kj} = \alpha \vartheta_k z_j; \quad k=1,2,\dots,m; j=0,1,2,\dots,p \quad (2.6)$$

Langkah 7: dalam faktor ϑ terdapat perhitungan berbagai unit yang tersembunyi $y_j (j = 1, 2, \dots, p)$.

$$\vartheta_{net_j} = \sum_{k=1}^m \vartheta_j w_{kj} \quad (2.7)$$

Faktor ϑ unit tersembunyi dari :

$$\vartheta_k = \vartheta_{net_j} f'(z_{net_j}) = \vartheta_{net_j} z_j (1 - z_j) \quad (2.8)$$

Bobot yang terdapat dalam v_{ji} dihitung untuk merubah bobot selanjutnya.

$$\Delta v_{ji} = \alpha \vartheta_j x_i; \quad j=1,2,\dots,p; i=0,1,2,\dots,n \quad (2.9)$$

Tahap III : Perubahan bobot

Langkah 8 : Semua perubahan pada tiap-tiap bobot dihitung.

Unit *output* dalam pergantian nilai bobot yang dituju:

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} ; k=1,2,\dots,m ; j=0,1,2,\dots,p \quad (2.10)$$

Unit tersembunyi yang dituju oleh perubahan nilai bobot:

$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji} ; j=1,2,\dots,p ; i=0,1,2,\dots,n \quad (2.11)$$

Jaringan dinyatakan siap untuk dipakai pada pengenalan pola ketika pelatihan sudah berakhir. *Output* dari nilai pada jaringan hanya memakai tahap sistem propagasi bergerak maju.

Pendahuluan langkah 4-5 harus diselesaikan disamping fungsi sigmoid dalam biner. Tidak beda dengan langkah 6-7 juga didahulukan.

Langkah 9: akhir iterasi, pengujian saat berhenti.



2.9 Pengujian MSE

MSE merupakan metode alternatif dalam suatu metode peramalan. Pendekatan ini penting karena teknik ini menghasilkan kesalahan yang moderat lebih disukai oleh suatu peramalan yang menghasilkan kesalahan yang sangat besar. MSE dihitung dengan menjumlahkan kuadrat semua kesalahan peramalan pada setiap periode dan membaginya dengan jumlah periode peramalan.

Rumus menghitung *Mean Square Error* (MSE) adalah sebagai berikut :

$$MSE = \frac{\sum Et^2}{n}$$

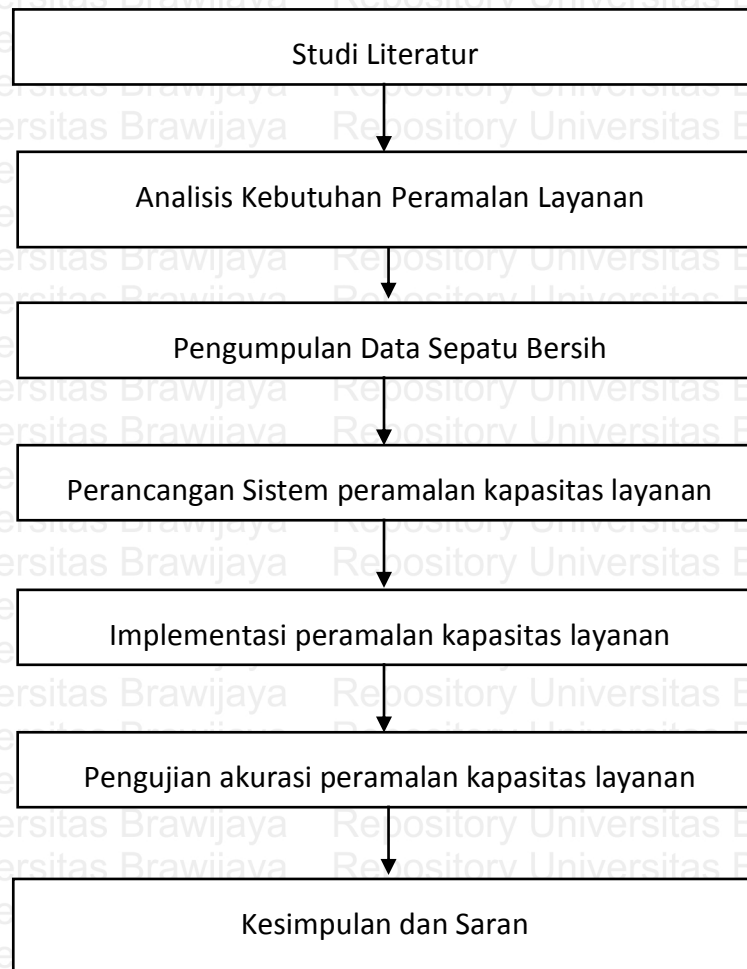
Et^2 = nilai galat kuadrat

n = banyak data



BAB 3 METODOLOGI

Metode dalam jaringan syaraf tiruan digunakan pada bagian ini untuk menjabarkan urutan pada penelitian “Sistem peramalan kapasitas layanan sepatu”. Tahapan atau berbagai urutan dalam penelitian ini adalah studi literatur, analisis kebutuhan peramalan layanan, pengumpulan data sepatu bersih, perancangan sistem peramalan kapasitas layanan, penerapan atau implementasi peramalan kapasitas layanan, pengujian akurasi peramalan kapasitas layanan, kesimpulan dan saran. Blok metode-metode atau urutan dalam pengerjaan peramalan kapasitas layanan pada sepatu tergambar pada Gambar 3.1 (Blok metode penelitian peramalan kapasitas layanan).



Gambar 3.1 Blok metode penelitian peramalan kapasitas layanan



3.1 Studi literatur

Pembelajaran yang dilakukan untuk menunjang penelitian mengenai “Sistem peramalan Jumlah Layanan pada jasa perawatan sepatu (studi kasus sepatu bersih)”. Diantaranya:

1. Sepatu
2. Metode Jaringan syaraf tiruan
3. Layanan laundry sepatu
4. Backpropagation

3.2 Analisis Kebutuhan

Analisis kebutuhan dalam membangun peramalan kapasitas layanan sepatu adalah kebutuhan yang diperlukan pada pengembangan “Sistem peramalan Jumlah Layanan pada jasa perawatan sepatu (studi kasus sepatu bersih)”. Penyesuaian penggunaan data dan variabel yang dipakai dalam kebutuhan sistem layanan peramalan sepatu perlu dilakukan analisis.

Dari keseluruhan yang dibutuhkan untuk mengembangkan Sistem peramalan kapasitas layanan dalam penelitian sistem peramalan kapasitas layanan ini meliputi:

1. Data dari outlet sepatu bersih yang diperlukan, adalah :
Jumlah sepatu pada riwayat layanan di sepatu bersih
2. Untuk melakukan atau menjalankan prediksi pada sistem peramalan kapasitas layanan ini, variabel yang dipakai meliputi:
Jumlah layanan perawatan sepatu setiap hari di sepatu bersih .

3.3 Pengumpulan Data

Dari kumpulan riwayat sepatu pada sepatu bersih yaitu jumlah layanan pada setiap harinya adalah hal yang dibutuhkan dalam peramalan kapasitas layanan sepatu. Manajer operasional sepatu bersih memberikan sumber data yang diperlukan, dari wawancara dengan manajer operasional sepatu bersih tersebut didapatkan jumlah layanan dan riwayat perawatan sepatu pada sepatu bersih.

Baik berupa data utama atau biasa disebut dengan primer juga bentuk data sekunder, keduanya dipergunakan untuk mengumpulkan berbagai data untuk kegiatan layaknya seperti penelitian data yang diperoleh dari pihak lain seperti literatur disebut atau masuk dalam data sekunder. Sedangkan melakukan wawancara seperti dengan manajer operasional pada sepatu bersih adalah data primer karena didapat dari yang bersangkutan dengan penelitian peramalan kapasitas layanan sepatu di sepatu bersih ini. Dari apa yang sudah disebutkan pengumpulan data memiliki sifat kuantitatif pada sepatu bersih, akan dilakukan



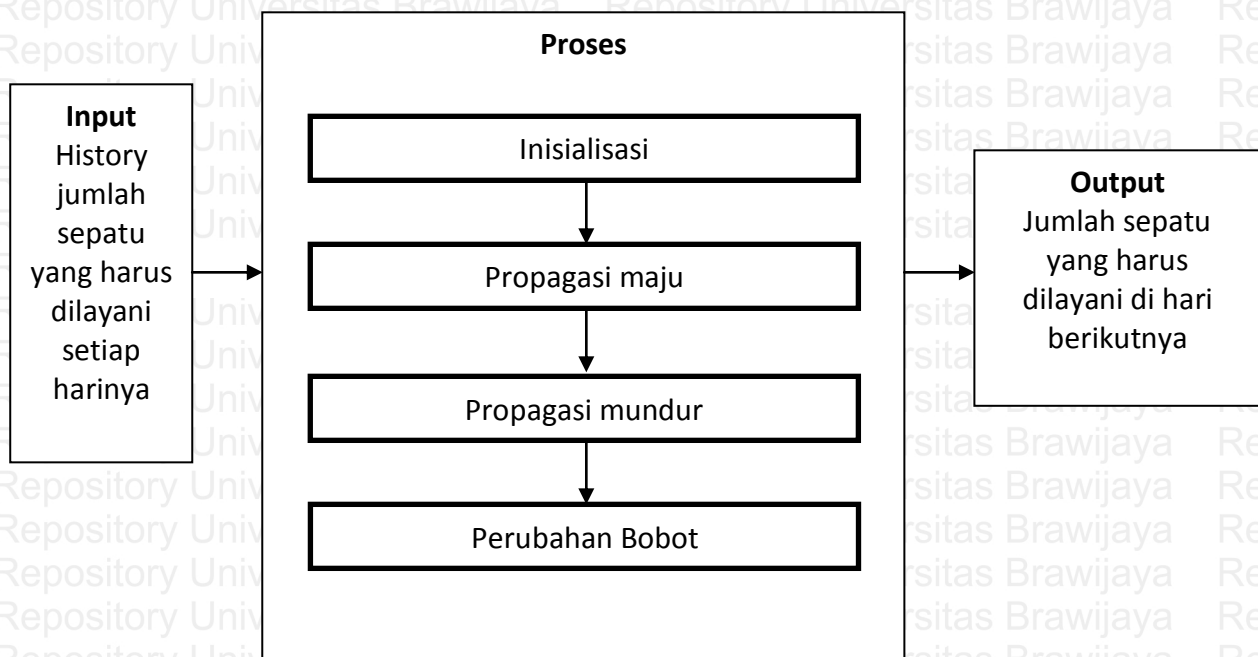
dengan wawancara langsung kepada manajer operasional sepatu bersih dan kuisisioner.

3.4 Perancangan Sistem

Pembuatan sistem peramalan sepatu diambil dari lapangan yaitu outlet sepatu bersih dan analisis dalam peramalan kapasitas layanan sepatu dibuat atau dibangun dari hasil pengumpulan data sepatu bersih juga telah dilakukan proses pembelajaran pada sepatu bersih. Berbagai hal dalam pembuatan sistem peramalan kapasitas layanan sepatu layaknya bagaimana arsitekturnya dirancang, juga kebutuhan berbagai perangkat lunak akan dijelaskan pada bagian ini.

Pemodelan “Sistem Peramalan kapasitas layanan sepatu” pada sepatu bersih ini, digunakan mesin berupa inferensi dengan metodenya yaitu propagasi balik. Jumlah layanan sepatu pada hari sebelumnya dijadikan untuk memperoleh peramalan jumlah sepatu yang harus dilayani di esok harinya. Dengan memakai propagasi balik sistem peramalan kapasitas layanan sepatu akan menghasilkan jumlah layanan sepatu yang diramalkan.

Sistem peramalan kapasitas layanan sepatu digambarkan dengan menggunakan blok diagram guna memvisualisasikan serta mengurai fungsi serta cara kerjanya.



Gambar 3.2 Diagram Blok Sistem Peramalan Kapasitas Layanan Sepatu



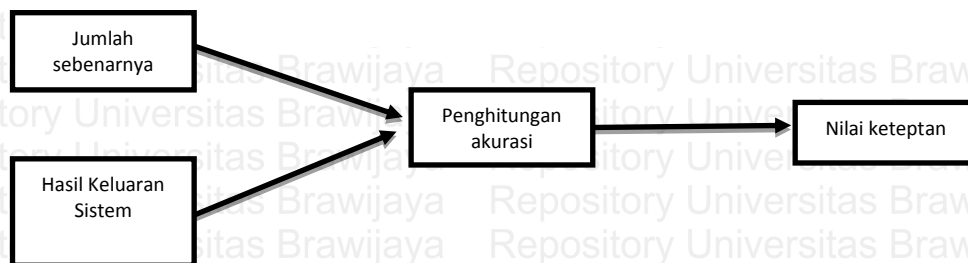
3.5 Implementasi

Penentuan perancangan sistem berdasarkan penerapan sistem yang telah dibuat. Berbagai bahasa pemrograman memungkinkan digunakan dalam pembentukan sistem ini, namun java dipilih guna merancang kebutuhan sistem peramalan kapasitas layanan sepatu, juga peralatan pendukung lainnya. Penerapannya pun meliputi:

- a) Pembuatan *User Interface*.
- b) Penggunaan java dalam pembuatan sistem peramalan kapasitas layanan sepatu digunakan dengan jaringan syaraf tiruan.

3.6 Pengujian Sistem

Dari apa yang sudah dirancang dan dibuat diperlukan pengujian demi menemukan dan menentukan tingkat keberhasilan dari sistem peramalan kapasitas layanan sepatu. Perbandingan data fakta dengan hasil dari keluaran prediksi digunakan untuk menguji nilai akurasi. Pada gambar berikut digambarkan alur pengujian sistem peramalan kapasitas layanan sepatu:



Gambar 3.3 Diagram Blok Pengujian Akurasi Peramalan Kapasitas layanan

3.7 Analisis

Dalam hal proses analisis, hasil akhir terlihat dari apa yang terjadi dan berjalan di lapangan apakah sistem peramalan kapasitas layanan sepatu sudah memiliki akurasi yang baik dalam hal meramalkan kapasitas sepatu.

3.8 Kesimpulan

Setelah melewati berbagai proses atau langkah-langkah dari seluruh tahap seperti studi mengenai literatur sistem peramalan sampai masuk ke tahap atau proses pengujian serta hasil yang dikeluarkan dari proses analisis, menarik sebuah kesimpulan menjadi hal akhir yang dilakukan. Penarikan untuk menghasilkan kesimpulan dilakukan dengan pencocokan antara peramalan sistem dengan data yang ada pada sepatu bersih. Disamping itu melakukan evaluasi guna mencari bentuk kesalahan dan kekurangan yang dimiliki dalam peramalan kapasitas layanan sepatu untuk penelitian-penelitian kedepannya sekalipun dengan metode yang berbeda.



BAB 4 PERANCANGAN

Penjelasan metode Backpropagation dalam kaitannya dengan kebutuhan yang diperlukan dalam perancangan sistem prediksi jumlah layanan laundry sepatu .

4.1 Jaringan Syaraf Tiruan

Menurut dari pendapat Siang (2009) terdapat berbagai bentuk pola jaringan yang umum dipakai, diantaranya :

1. Jaringan dalam Layar Tunggal (*Single Layer Network*)

Penghubungan langsung neuron-neuron antara jaringan *input* dan *output*. Dalam perceptron, ada sebuah unit neuron *output*. Di dalam jaringan tersebut, semua unit *input* atau masukan dihubungkan dengan semua unit *output*, meskipun dengan bobot yang berbeda-beda. Tidak ada unit *input* yang dihubungkan dengan unit *input* lainnya. Demikian pula dengan unit *output*.

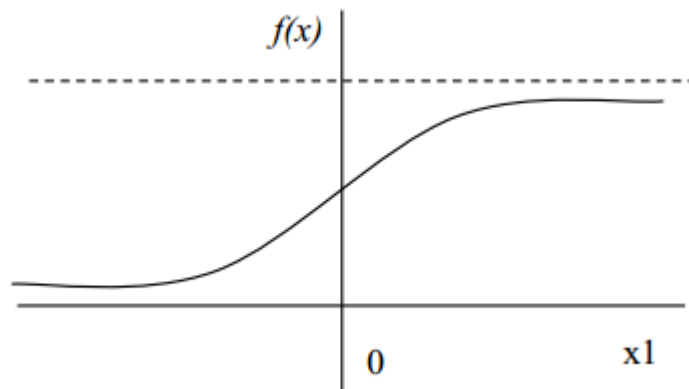
W dinyatakan sebagai besaran bobot diantara unit masukan atau *input* dan unit keluaran atau *output*. Bobot-bobot ini saling independen dalam tahapan pelatihannya. Dalam taapan pelatihan, bobot-bobot tersebut mengalami modifikasi guna meningkatkan akurasi dari hasil. Model semacam ini tepat digunakan untuk pengenalan pola karena kesederhanaannya.

2. Jaringan Layar Jamak

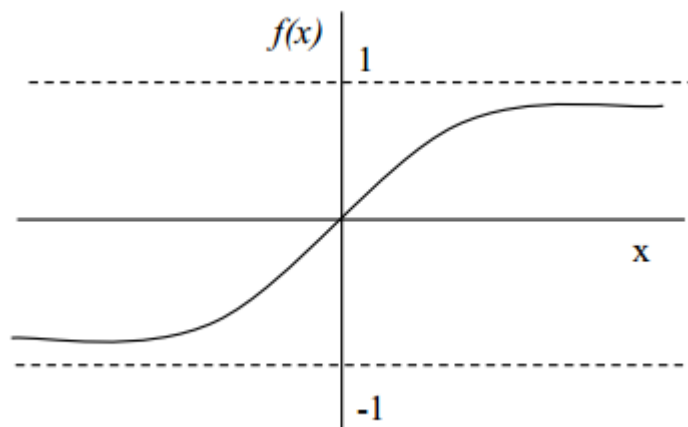
Bentuk pola ini adalah perluasan dari layar tunggal atau single layer . Dalam jaringan ini, selain unit *input* maupun *output*, ada unit-unit lainnya (sering disebut layer tersembunyi atau hidden layer). Juga dimungkinkan terdapat beberapa lapisan yang tersembunyi. Sama seperti pada unit *input* maupun *output*, unit-unit dalam satu lapisan tidak saling berhubungan satu sama lainnya.

4.2 Backpropagation

Dalam proses propagasi balik, bebrapa syarat harus dipenuhi sebelum menggunakan fungsi aktivasi, diantaranya kontinyu terdiferensial dengan mudah serta fungsi tidak memiliki kecenderungan menurun. Ketika ada fungsi yang dapat memenuhi ketiganya, maka kerap fungsi yang dipakai adalah fungsi biner sigmoid yang mempunyai range (0,1). Turunan yang digunakan $f'(x) = f(x)(1 - f(x))$ dengan $f(x) = \frac{1}{1 + e^{-x}}$. Grafik dari penerjemahan fungsinya dapat dilihat di gambar 4.1.



Gambar 4.1 Fungsi Aktivasi dalam Sigmoid Biner



Gambar 4.2 Fungsi Aktivasi dalam Sigmoid Bipolar

Fungsi yang mirip dan sering digunakan pada fungsi lain adalah fungsi dalam bentuk sigmoid biner, menggunakan syarat range $(-1,1)$. Menggunakan turunan $f'(x) = \frac{f(x)(1-f(x))}{2}$ dengan $f(x) = \frac{2}{1+e^{-x}} - 1$. Grafik fungsinya dapat dilihat di gambar 2.4.

Fungsi dalam bentuk sigmoid memiliki nilai batas atas sama dengan 1. Maka untuk bentuk pola yang targetnya lebih dari 1, pola masukan dan keluaran harus terlebih dahulu ditransformasi sehingga semua polanya dapat mempunyai range yang sepadan seperti fungsi dalam sigmoid yang dipergunakan. Bentuk lain adalah dengan menggunakan fungsi aktivasi dalam sigmoid hanya pada lapisan yang bukan merupakan lapisan keluaran atau *output*. Dalam lapisan keluaran, fungsi identitas $f(x) = x$ adalah fungsi yang dipakai untuk aktivasi.



Pelatihan Standar Backpropagation

Terdapat 3 tahapan dalam pelatihan propagasi mundur, yaitu maju, propagasi berbalik mundur, serta perubahan nilai dalam bobot. Jaringan dengan satu lapisan tersembunyi (dengan fungsi aktivasi sigmoid biner) mempunyai algoritma pelatihan sebagai berikut :

Langkah 0: Inisialisasi semua dari nilai bobot dengan bilangan acak kecil

Langkah 1: Jika kondisi penghentian belum dicapai, lakukan tahapan atau langkah 2 sampai 8

Langkah 2: Untuk tiap-tiap pasang pada data pelatihan, dilakukan langkah atau tahap 3 sampai 8

Tahap I : Propagasi Maju

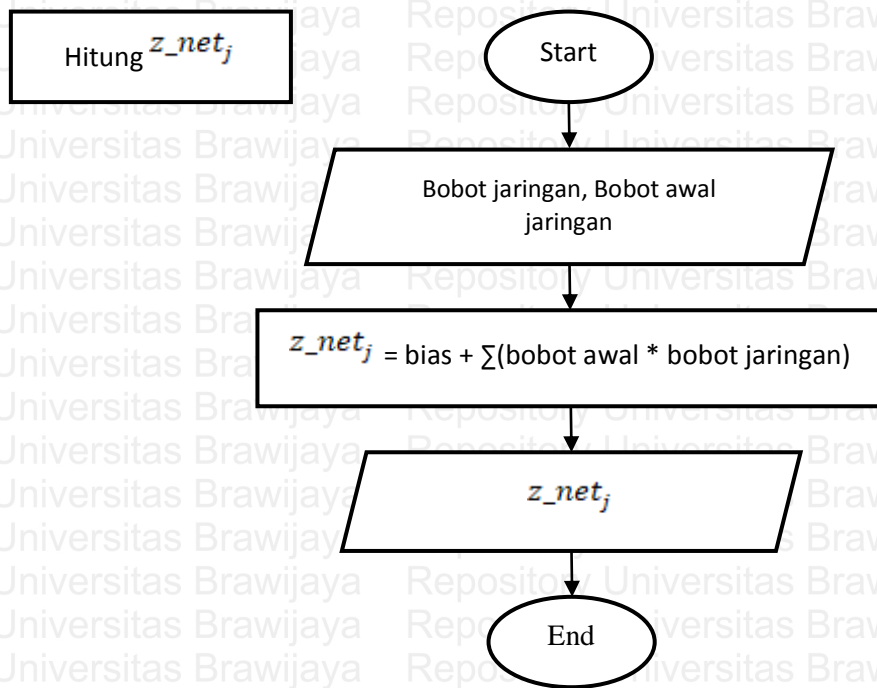
Langkah 3: Unit yang tersembunyi menerima sinyal dari unit masukan yang berada di bawahnya.

Langkah 4: Penghitungan seluruh keluaran di dalam unit yang tersembunyi

$z_j (j = 1, 2, \dots, p)$

$$z_{net_j} = v_{j0} + \sum_{i=1}^n x_i v_{ji} \quad (2.1)$$

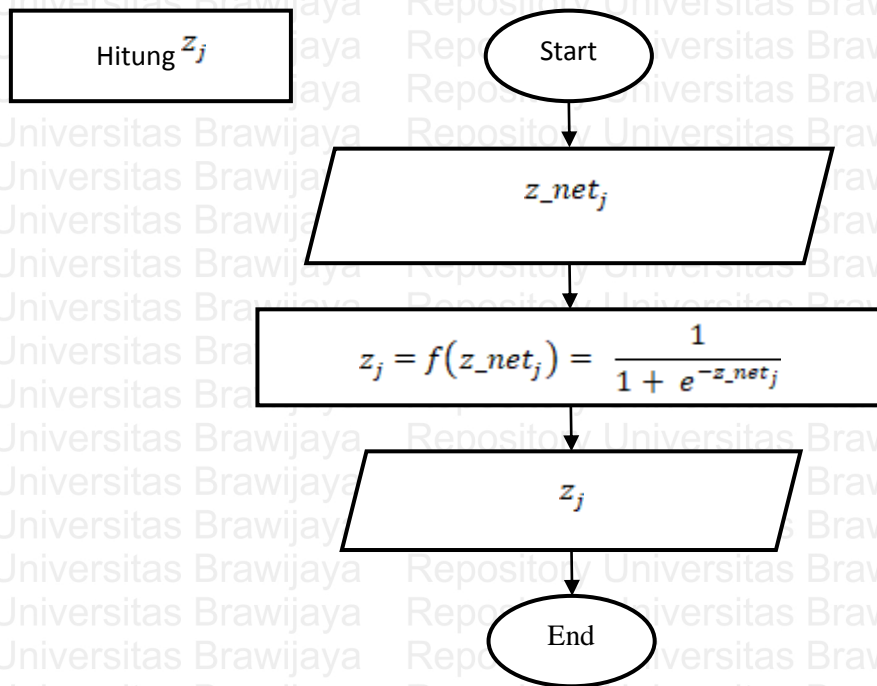
Diagram penghitungan nilai z_{net_j} dapat dilihat di gambar 4.3 (penghitungan nilai pada z_{net_j}).



Gambar 4.3 Diagram Alir penghitungan nilai pada z_{net_j}

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}} \quad (2.2)$$

Penghitungan z_j ditunjukkan oleh diagram alir gambar 4.4 (Diagram Alir penghitungan nilai pada z_j).



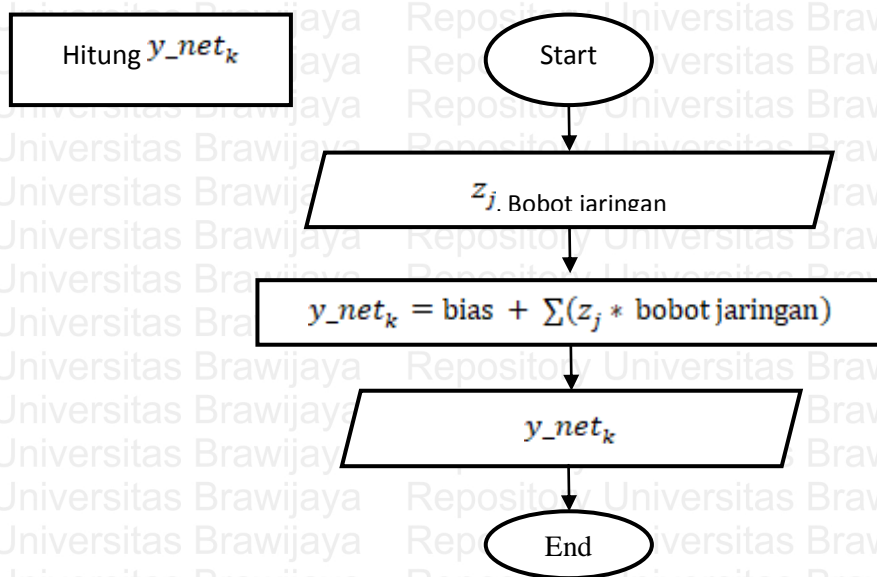
Gambar 4.4 Diagram Alir penghitungan nilai pada z_j

Langkah 5: Melakukan penghitungan pada seluruh keluaran jaringan di unit

$$y_k (k = 1, 2, \dots, m)$$

$$y_{net_k} = v_{k0} + \sum_{j=1}^p z_j w_{kj} \quad (2.3)$$

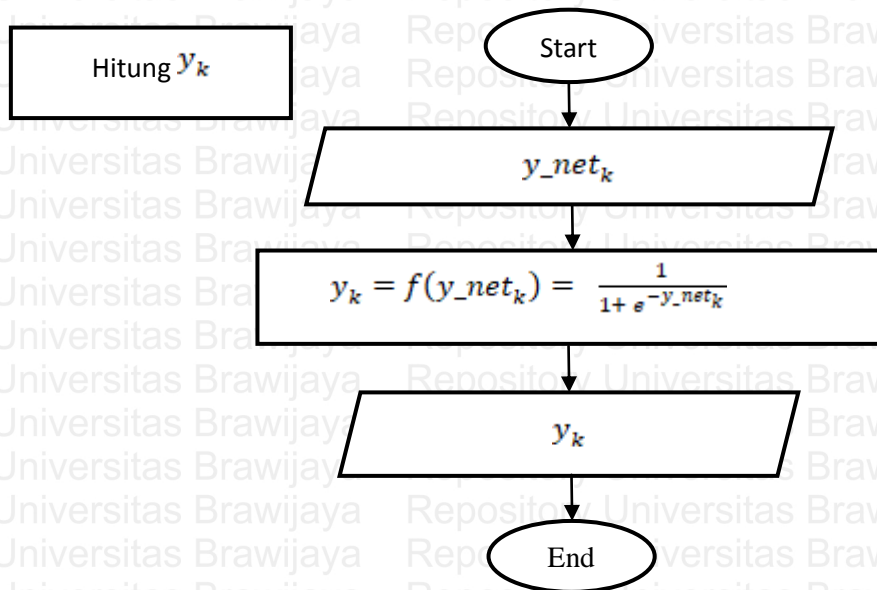
Diagram untuk menghitung y_{net_k} ditunjukkan oleh gambar diagram alir 4.5 (diagram alir penghitungan nilai pada y_{net_k})



Gambar 4.5 Diagram Alir penghitungan nilai pada y_{net_k}

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}} \quad (2.4)$$

Diagram untuk perhitungan y_k ditunjukkan oleh gambar diagram alir 4.6 (Diagram perhitungan y_{net_k}).



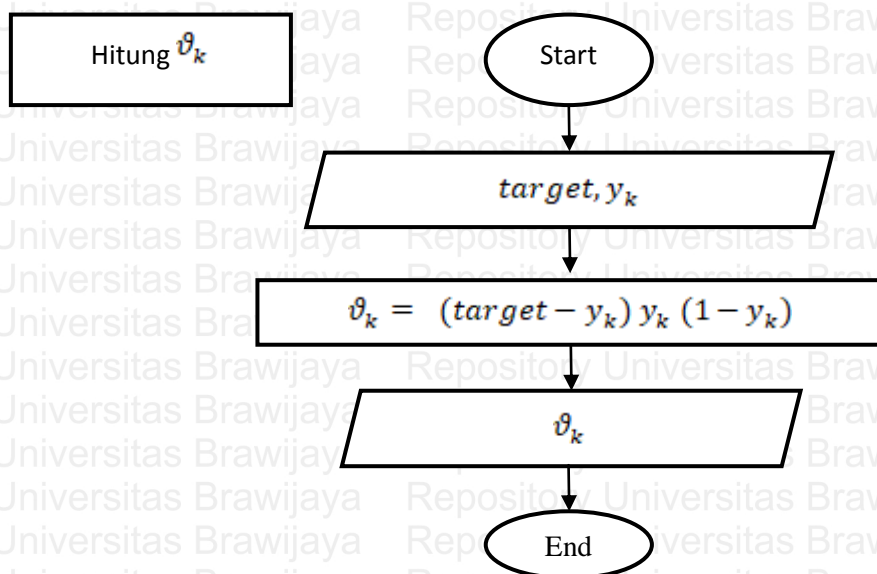
Gambar 4.6 Diagram Alir penghitungan nilai pada y_{net_k}

Tahap II : Propagasi mundur

Langkah 6: Perhitungan pada faktor ϑ unit keluaran berdasar pada nilai kesalahan di tiap-tiap unit keluaran $y_k (k = 1, 2, \dots, m)$

$$\vartheta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k) \quad (2.5)$$

Diagram untuk menghitung ϑ_k ditunjukkan oleh diagram alir Gambar 4.7 (diagram alir penghitungan nilai pada ϑ_k).



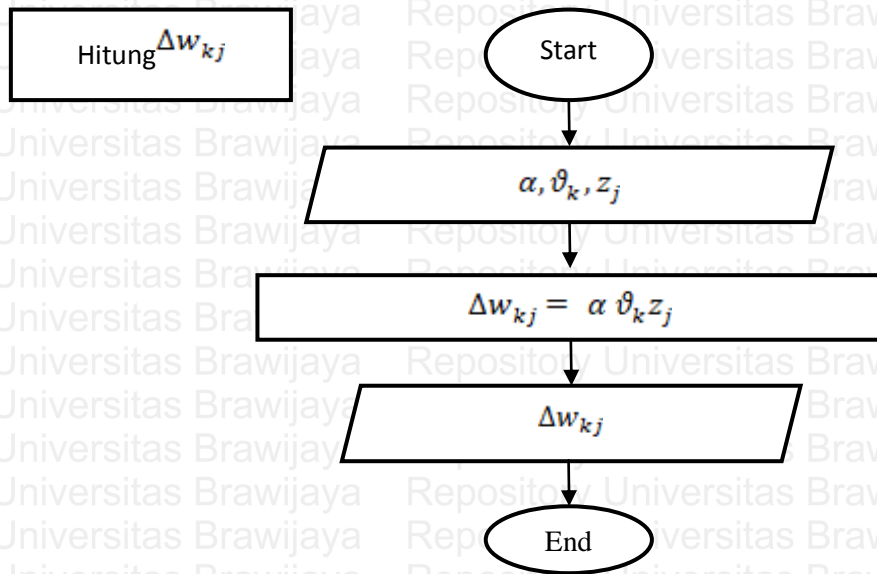
Gambar 4.7 Diagram Alir Peghitungan Nilai pada ϑ_k

ϑ_k Adalah bagian nilai kesalahan yang digunakan dalam merubah nilai bobot pada lapisan bobot di bawahnya (langkah 7)

Hitung suku dalam perubahan bobot w_{kj} (yang akan digunakan nantinya untuk perubahan bobot) menggunakan laju percepatan α .

$$\Delta w_{kj} = \alpha \vartheta_k z_j; \quad k = 1, 2, \dots, m; \quad j = 0, 1, 2, \dots, p \quad (2.6)$$

Diagram yang ditunjukkan dalam menghitung Δw_{kj} digambarkan oleh diagram alir gambar 4.8 (diagram alir penghitungan nilai pada Δw_{kj}).

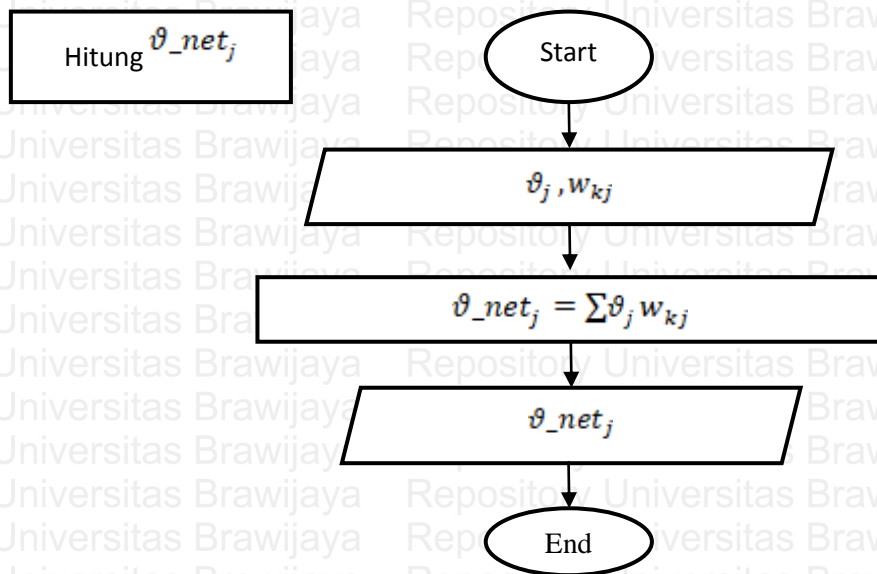


Gambar 4.8 Diagram Alir penghitungan nilai pada Δw_{kj}

Langkah 7: Penghitungan ϑ menurut nilai kesalahan tiap-tiap unit tersembunyi $y_j (j = 1, 2, \dots, p)$.

$$\vartheta_{net_j} = \sum_{k=1}^m \vartheta_j w_{kj} \quad (2.7)$$

Diagram untuk menghitung ϑ_{net_j} ditunjukkan oleh diagram alir gambar 4.9 (diagram alir penghitungan nilai pada ϑ_{net_j}).

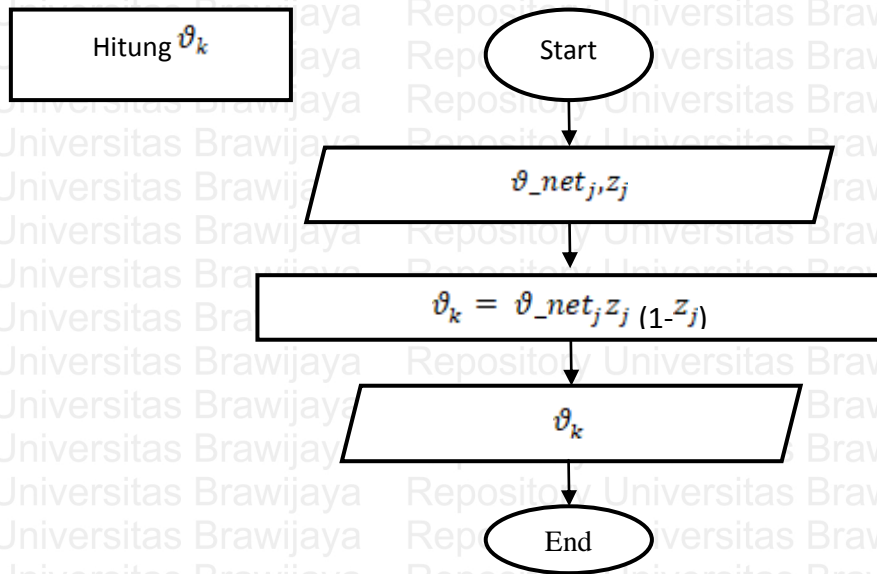


Gambar 4.9 Diagram Alir penghitungan nilai pada ϑ_{net_j}

Faktor ϑ unit tersembunyi dari :

$$\vartheta_k = \vartheta_{net_j} f'(z_{net_j}) = \vartheta_{net_j} z_j (1 - z_j) \quad (2.8)$$

Diagram untuk menghitung ϑ_k digambarkan dengan diagram alir di gambar 4.10 (diagram alir penghitungan nilai pada ϑ_k).

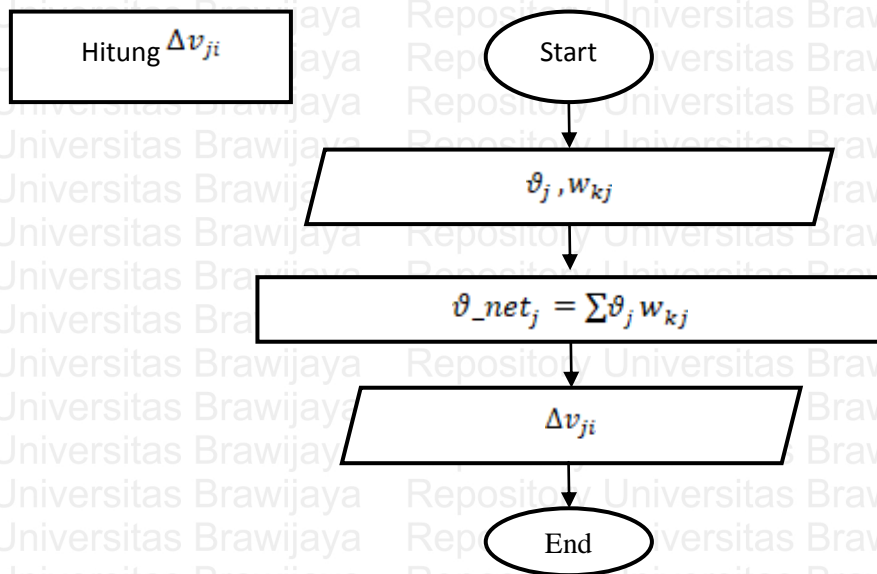


Gambar 4.10 Diagram Alir penghitungan nilai pada θ_k

Hitung suku perubahan bobot v_{ji} (yang dipakai untuk merubah bobot v_{ji}).

$$\Delta v_{ji} = \alpha \theta_j x_i ; \quad j = 1, 2, \dots, p; i = 0, 1, 2, \dots, n \quad (2.9)$$

Diagram untuk menghitung θ_{net_j} digambarkan oleh diagram alir gambar 4.11 (diagram alir perhitungan θ_{net_j}).



Gambar 4.11 Diagram Alir penghitungan ilai pada Δv_{ji}

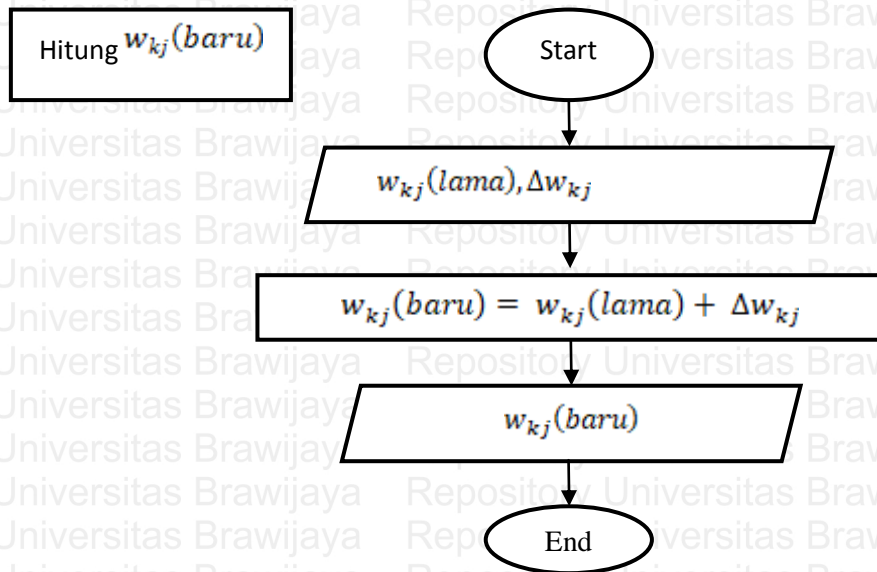
Tahap III : Perubahan bobot

Langkah 8: Penghitungan tiap-tiap pergantian bobot

Pergantian bobot garis yang menuju ke unit keluaran atau output :

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj} \quad ; k = 1, 2, \dots, m \quad ; j = 0, 1, 2, \dots, p \quad (2.10)$$

Diagram untuk menghitung $w_{kj}(\text{baru})$ ditunjukkan oleh diagram alir di gambar 4.12 (diagram alir penghitungan nilai pada $w_{kj}(\text{baru})$).

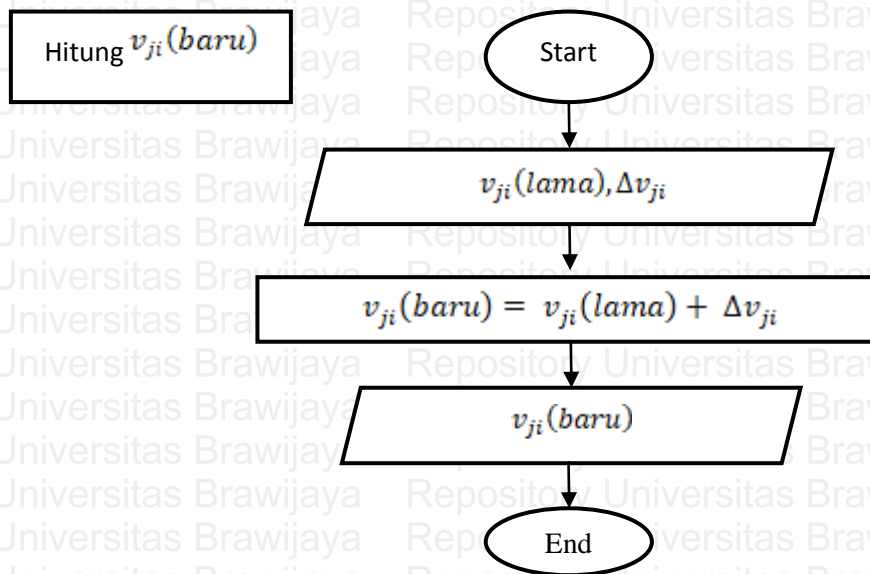


Gambar 4.12 Diagram Alir penghitungan nilai pada $w_{kj}(baru)$

Perubahan bobot linear yang menuju unit yang tersembunyi :

$$v_{ji}(baru) = v_{ji}(lama) + \Delta v_{ji} \quad ; j = 1,2,\dots,p \quad ; i = 0,1,2,\dots,n \quad (2.11)$$

Diagram untuk menghitung $v_{ji}(baru)$ ditunjukkan oleh diagram alir yang digambarkan dalam gambar 4.13($v_{ji}(baru)$).



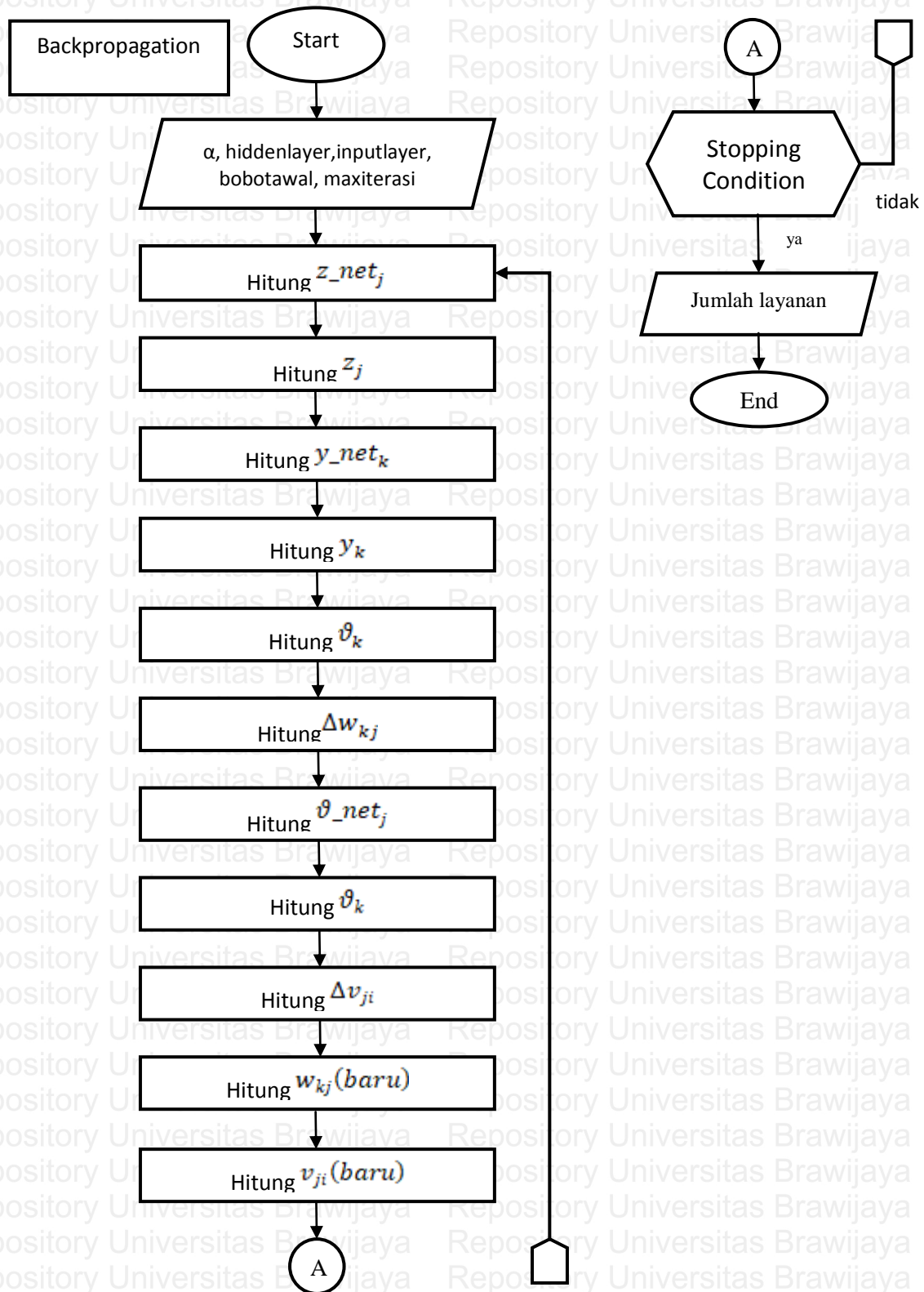
Gambar 4.13 Diagram Alir Penghitungan nilai pada $v_{ji}(baru)$

Pengenalan pola dapat dilaksanakan tepat setelah proses pelatihan berakhir. Dalam hal ini, hanyalah propagasi maju saja yang dapat digunakan untuk menghasilkan keluaran dalam jaringan.

Jika aktivasi sigmoid bukan menggunakan biner, urutan 4 sampai dengan 5 harus sudah selesai. Sama halnya untuk turunannya pada tahap 6 sampai 7.

Langkah 9: Uji kondisi berhenti (akhir iterasi).

Diagram alir proses perhitungan backpropagation mulai dari inisialisasi bobot hingga memenuhi kondisi untuk menghentikan iterasi ditunjukkan oleh gambar 4.14.



Gambar 4.14 Diagram Alir Backpropagation



4.3 Perhitungan Manual

Tahapan di bawah merupakan langkah-langkah dari sistem peramalan layanan sepatu:

Langkah 0: bilangan acak yang bernilai kecil menginisialisasikan nilai dari bobot

Langkah 1: Langkah atau tahapan 2 sampai dengan 8 dijalankan jika dalam penghentian kondisi belum terpenuhi

Langkah 2: Dilakukan langkah 3 sampai dengan 8 bagi tiap-tiap data pelatihan

Langkah 3: Unit yang tersembunyi menerima sinyal dari unit masukan dibawahnya

Langkah 4: Pada unit yang tersembunyi dihitung semua unit keluaranya $z_j (j = 1, 2, \dots, p)$

Langkah 5: Pada jaringan dalam unit dihitung seluruh keluaranya. Hitung $y_k (k = 1, 2, \dots, m)$

Langkah 6: Dalam tiap-tiap unit keluaran dihitung nilai faktor ϑ mengacu pada erornya $y_k (k = 1, 2, \dots, m)$

Langkah 7: Dalam tiap-tiap unit tersembunyi dihitung nilai faktor ϑ mengacu pada nilai salah atau eror yang berada di unit yang tersembunyi $y_j (j = 1, 2, \dots, p)$.

Langkah 8: Pergantian semua bobot dihitung

Langkah 9: Akhir iterasi, merupakan pengujian dalam kondisi berhenti

4.3.1 Contoh perhitungan manual proses backpropagation untuk data latih

Inisialisasi awal

Perhitungan secara manual memiliki contoh sebagai berikut, nilai α diberikan sebesar 0,5, 2 untuk layer yang tersembunyi, bernilai 6 untuk layer masukan, dan memiliki satu kali nilai iterasi maksimum, Tabel 4.1 menunjukkan bobot pertama jaringan yang diinisialisasi, Tabel 4.2 menggambarkan bobot yang diterima oleh lapisan tersembunyi oleh lapisan *input*, sedangkan tabel 4.3 menunjukkan bobot yang diterima oleh lapisan keluaran yang berasal dari lapisan *output*.

Tabel 4.1 Inisialisasi bobot awal jaringan

	Z1	Z2
X1	-1	-0,5



X2	-0,5	0,9
X3	-1	0,7
X4	0,1	-0,8
X5	0,5	0,9
X6	0,4	-0,2
1	-0,7	-0,1

Tabel 4.2 bobot jaringan dari lapisan masukan ke lapisan tersembunyi

LATIH KE-	X1	X2	X3	X4	X5	X6	TARGET
1	0,40769 2	0,16153 8	0,40769 2	0,1 9	0,53076 9	0,22307 7	0,53076 9
2	0,16153 8	0,40769 2	0,1 9	0,53076 9	0,22307 7	0,53076 9	0,53076 9
3	0,40769 2	0,1 9	0,53076 9	0,22307 7	0,53076 9	0,53076 9	0,34615 4
4	0,1 9	0,53076 9	0,22307 7	0,53076 9	0,53076 9	0,34615 4	0,28461 5
5	0,53076 9	0,22307 7	0,53076 9	0,53076 9	0,34615 4	0,28461 5	0,1

Tabel 4.3 bobot dari lapisan tersembunyi ke lapisan output

	Y
Z1	1
Z2	-0,6
1	0,1

Hitung semua keluaran dalam unit tersembunyi $z_j (j = 1, 2, \dots, p)$ data latih satu

$$z_{net_1} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$= -0,7 + ((0,407692308 * -1) + (0,161538462 * -0,5) + (0,407692308 * -1) + (0,1 * 0,1) + (0,530769231 * 0,5) + (0,223076924 * 0,4))$$

$$= -1,231538462$$

$$z_{net_2} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$= -0,7 + ((0,161538462 * -0,5) + (0,407692308 * -1) + (0,1 * 0,1) + (0,530769231 * 0,5) + (0,223076924 * 0,4) + (0,530769231 * -0,7))$$

$$= 0,480000001$$



$$z_1 = f(z_{net_1}) = \frac{1}{1 + e^{-z_{net_1}}}$$

$$= \frac{1}{1 + e^{-(-1,231538462)}}$$

$$= 0,225912272$$

$$z_2 = f(z_{net_2}) = \frac{1}{1 + e^{-z_{net_2}}}$$

$$= \frac{1}{1 + e^{-0,480000001}}$$

$$= 0,617747875$$

Hitung semua keluaran jaringan dalam unit $y_k (k = 1, 2, \dots, m)$

$$y_{net_k} = v_{k0} + \sum_{j=1}^p z_j w_{kj}$$

$$= 0,1 + ((0,225912272352397 * 1) + (0,617747874887317 * -0,6))$$

$$= -0,044736453$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

$$= \frac{1}{1 + e^{-(-0,044736453)}}$$

$$= 0,488817752$$

Hitung faktor ϑ unit keluaran berdasarkan eror di setiap unit keluaran

$$y_k (k = 1, 2, \dots, m)$$

$$\vartheta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k)$$

$$= (0,530769231 - 0,488817751759017) * (0,488817751759017) * (1 - 0,488817751759017)$$

$$= 0,010482624$$

$$\Delta w_{10} = \alpha \vartheta_1 z_0$$

$$= 0,5 * 0,0105 * 1$$

$$= 0,00525$$

$$\Delta w_{11} = \alpha \vartheta_1 z_1$$

$$= 0,5 * 0,0105 * 0,2259$$

$$= 0,00119$$



$$\begin{aligned}\Delta w_{12} &= \alpha \vartheta_1 z_2 \\ &= 0,5 * 0,0105 * 0,6177 \\ &= 0,00324\end{aligned}$$

Hitung faktor ϑ unit tersembunyi berdasarkan eror di setiap unit tersembunyi $y_j (j = 1, 2, \dots, p)$.

$$\begin{aligned}\vartheta_{net_1} &= \sum_{k=1}^m \vartheta_1 w_{k1} \\ &= 0,01048 * 1 \\ &= 0,01048\end{aligned}$$

$$\begin{aligned}\vartheta_{net_2} &= \sum_{k=1}^m \vartheta_2 w_{k2} \\ &= 0,01048 * -0,6 \\ &= -0,00629\end{aligned}$$

$$\begin{aligned}\vartheta_1 &= \vartheta_{net_1} z_1 (1-z_1) \\ &= 0,010482624085031 * 0,225912272352397 * (1-0,225912272352397) \\ &= 0,001833159\end{aligned}$$

$$\begin{aligned}\vartheta_2 &= \vartheta_{net_2} z_2 (1-z_2) \\ &= -0,00628957445101857 * 0,617747874887317 * (1-0,617747874887317) \\ &= -0,001485191\end{aligned}$$

$$\begin{aligned}\Delta v_{11} &= \alpha \vartheta_1 x_1 \\ &= 0,5 * 0,00183315850523269 * 0,407692308 \\ &= 0,00037368\end{aligned}$$

$$\begin{aligned}\Delta v_{12} &= \alpha \vartheta_1 x_2 \\ &= 0,5 * 0,00183315850523269 * 0,161538462 \\ &= 0,000148063\end{aligned}$$

$$\begin{aligned}\Delta v_{13} &= \alpha \vartheta_1 x_3 \\ &= 0,5 * 0,00183315850523269 * 0,407692308 \\ &= 0,000373682\end{aligned}$$

$$\begin{aligned}\Delta v_{14} &= \alpha \vartheta_1 x_4 \\ &= 0,5 * 0,00183315850523269 * 0,1 \\ &= 0,0000917\end{aligned}$$

$$\Delta v_{15} = \alpha \vartheta_1 x_5$$



$$= 0,5 * 0,00183315850523269 * 0,530769231$$

$$= 0,00048649$$

$$\Delta v_{16} = \alpha \vartheta_1 x_6$$

$$= 0,5 * 0,00183315850523269 * 0,223076924$$

$$= 0,00020447$$

$$\Delta v_{1t} = \alpha \vartheta_1 x_t$$

$$= 0,5 * 0,00183315850523269 * 0,530769231$$

$$= 0,00091658$$

$$\Delta v_{21} = \alpha \vartheta_2 x_1$$

$$= 0,5 * -0,00148519141757028 * 0,407692308$$

$$= -0,0003028$$

$$\Delta v_{22} = \alpha \vartheta_2 x_2$$

$$= 0,5 * -0,00148519141757028 * 0,161538462$$

$$= -0,00012$$

$$\Delta v_{23} = \alpha \vartheta_2 x_3$$

$$= 0,5 * -0,00148519141757028 * 0,407692308$$

$$= -0,0003028$$

$$\Delta v_{24} = \alpha \vartheta_2 x_4$$

$$= 0,5 * -0,00148519141757028 * 0,1$$

$$= -0,0000743$$

$$\Delta v_{25} = \alpha \vartheta_2 x_5$$

$$= 0,5 * -0,00148519141757028 * 0,530769231$$

$$= -0,0003941$$

$$\Delta v_{26} = \alpha \vartheta_2 x_6$$

$$= 0,5 * -0,00148519141757028 * 0,223076924$$

$$= -0,0001657$$

$$\Delta v_{2t} = \alpha \vartheta_2 x_t$$

$$= 0,5 * -0,00148519141757028 * 0,530769231$$

$$= -0,0007426$$

Hasil dari perubahan bobot dapat dilihat pada tabel 4.4.

Tabel 4.4 Suku perubahan bobot ke unit tersembunyi

X1	X2	X3	X4	X5	X6	1	
0,00037368	0,00014806	0,00037368	0,0000917	0,00048649	0,00020447	0,00091658	Z1



-0,0003028	-0,00012	-0,0003028	-0,000074	-0,0003941	-0,0001657	-0,0007426	22
------------	----------	------------	-----------	------------	------------	------------	----

Hitung semua perubahan bobot

$$\begin{aligned}w_{11}(\text{baru}) &= w_{11}(\text{lama}) + \Delta w_{11} \\ &= 1 + 0,00118407671363265 \\ &= 1,00118408\end{aligned}$$

$$\begin{aligned}w_{12}(\text{baru}) &= w_{12}(\text{lama}) + \Delta w_{12} \\ &= -0,6 + 0,00323780937588524 \\ &= -0,5967622\end{aligned}$$

$$\begin{aligned}w_{1t}(\text{baru}) &= w_{1t}(\text{lama}) + \Delta w_{1t} \\ &= 0,1 + 0,00524131204251548 \\ &= 0,10524131\end{aligned}$$

$$\begin{aligned}w_{11}(\text{baru}) &= w_{11}(\text{lama}) + \Delta w_{11} \\ &= 1 + 0,00118407671363265 \\ &= 1,00118408\end{aligned}$$

$$\begin{aligned}w_{12}(\text{baru}) &= w_{12}(\text{lama}) + \Delta w_{12} \\ &= -0,6 + 0,00323780937588524 \\ &= -0,5967622\end{aligned}$$

$$\begin{aligned}w_{1t}(\text{baru}) &= w_{1t}(\text{lama}) + \Delta w_{1t} \\ &= 0,1 + 0,00524131204251548 \\ &= 0,10524131\end{aligned}$$

$$\begin{aligned}v_{11}(\text{baru}) &= v_{11}(\text{lama}) + \Delta v_{11} \\ &= 0,000373682310964072 + -1 \\ &= -0,9996263\end{aligned}$$

$$v_{12}(\text{baru}) = v_{12}(\text{lama}) + \Delta v_{12}$$



$$= 0,000148062802768753 + -0,5$$

$$= -0,4998519$$

$$v_{13}(\text{baru}) = v_{13}(\text{lama}) + \Delta v_{13}$$

$$= 0,000373682310964072 + -1$$

$$= -0,9996263$$

$$v_{14}(\text{baru}) = v_{14}(\text{lama}) + \Delta v_{14}$$

$$= 0,0000917 + 0,1$$

$$= 0,10009166$$

$$v_{15}(\text{baru}) = v_{15}(\text{lama}) + \Delta v_{15}$$

$$= 0,000486492065061731 + 0,5$$

$$= -0,50048649$$

$$v_{16}(\text{baru}) = v_{16}(\text{lama}) + \Delta v_{16}$$

$$= 0,000204467680275873 + 0,4$$

$$= 0,40020447$$

$$v_{1r}(\text{baru}) = v_{1r}(\text{lama}) + \Delta v_{1r}$$

$$= 0,000916579252616343 + -0,7$$

$$= -0,6990834$$

$$v_{21}(\text{baru}) = v_{22}(\text{lama}) + \Delta v_{21}$$

$$= -0,00030275055842551 + -0,5$$

$$= -0,5003$$

$$v_{22}(\text{baru}) = v_{22}(\text{lama}) + \Delta v_{22}$$

$$= -0,000119957768684952 + 0,9$$

$$= 0,89988$$

$$v_{23}(\text{baru}) = v_{23}(\text{lama}) + \Delta v_{23}$$

$$= -0,00030275055842551 + 0,7$$

$$= 0,699697$$

$$v_{24}(\text{baru}) = v_{24}(\text{lama}) + \Delta v_{24}$$

$$= -7,42595708785142E-05 + -0,8$$

$$= -0,80007$$



$$\begin{aligned}v_{25}(\text{baru}) &= v_{25}(\text{lama}) + \Delta v_{25} \\ &= -0,00039414695329579 + 0,9 \\ &= 0,899606\end{aligned}$$

$$\begin{aligned}v_{26}(\text{baru}) &= v_{26}(\text{lama}) + \Delta v_{26} \\ &= -0,000165655966491389 + -0,2 \\ &= -0,20017\end{aligned}$$

$$\begin{aligned}v_{2t}(\text{baru}) &= v_{2t}(\text{lama}) + \Delta v_{2t} \\ &= -0,000742595708785142 + -0,1 \\ &= -0,10074\end{aligned}$$

Unit tersembunyi yang mengalami perubahan bobot ditunjukkan oleh tabel 4.5.

Tabel 4.5 Unit tersembunyi yang mengalami perubahan bobot

	Z1	Z2
X1	-0,9996263	-0,5003
X2	-0,4998519	0,89988
X3	-0,9996263	0,699697
X4	0,10009166	-0,80007
X5	0,50048649	0,899606
X6	0,40020447	-0,20017
1	-0,6990834	-0,10074

Kembali ke langkah pertama tapi menggunakan data ke 2 dan bobot terbaru, begitu seterusnya hingga proses pada data kelima hingga didapatkan hasil akhir seperti yang telah diperlihatkan oleh Tabel 4.6.

Tabel 4.6 Perubahan bobot di unit tersembunyi

	Z1	Z2
X1	-1,0051666	-0,49488
X2	-0,50476504	0,903897
X3	-1,0062874	0,705937
X4	0,09268199	-0,79345
X5	0,49329482	0,9057
X6	0,39447186	-0,19524
1	-0,71500727	-0,08668



4.3.2 Contoh perhitungan manual proses backpropagation untuk data uji

Contoh yang diberikan dalam sistem penghitungan manual, tabel 4.7 menunjukkan nilai-nilai dari inisialisasi data uji, sedangkan oleh tabel 4.8 menunjukkan bobot daripada jaringan tersembunyi yang diterima dari lapisan *input*, dan Tabel 4.9 menunjukkan bobot yang diterima oleh lapisan keluaran dari lapisan tersembunyi.

Tabel 4.7 Inisialisasi data uji

UJI KE-	X1	X2	X3	X4	X5	X6	TARGET
1	0,653846154	0,161538462	0,1	0,715385	0,407692	0,469231	0,284615385
2	0,161538462	0,1	0,715385	0,407692	0,469231	0,284615	0,161538462
3	0,1	0,715384616	0,407692	0,469231	0,284615	0,161538	0,161538462
4	0,715384616	0,407692308	0,469231	0,284615	0,161538	0,161538	0,161538462

Tabel 4.8 bobot jaringan dari lapisan *input* ke lapisan tersembunyi

	Z1	Z2
X1	-1,005166597	-0,494881399
X2	-0,504765036	0,903897497
X3	-1,006287397	0,705936703
X4	0,09268199	-0,793450489
X5	0,493294824	0,905700342
X6	0,394471863	-0,195238741
1	-0,71500727	-0,086678137

Tabel 4.9 bobot dari lapisan tersembunyi ke lapisan *output*

	Y
Z1	0,980163
Z2	-0,64894
1	0,011615

Hitung semua keluaran dalam unit tersembunyi z_j ($j = 1, 2, \dots, p$) data uji satu

$$z_{net_1} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

$$= -0,715007270383435 + ((0,653846154 * -1,00516659705275) + (0,161538462 * -0,504765035944172) + (0,1 * -1,00628739697541) + (0,715384616 * 0,0926819901707208) + (0,407692308 * 0,493294823648014) + (0,46923077 * 0,394471862740928))$$

$$= -1,10188518$$

$$z_{net_2} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$



$$= -0,0866781374101646 + ((0,653846154 * -0,494881399076452) + (0,161538462 * 0,903897496579359) + (0,1 * 0,705936703294884) + (0,715384616 * -0,793450488556303) + (0,407692308 * 0,905700342274859) + (0,46923077 * -0,195238741154724))$$

$$= -0,48363379$$

$$z_1 = f(z_{net_1}) = \frac{1}{1 + e^{-z_{net_1}}}$$

$$= \frac{1}{1 + e^{-(-1,10188518)}}$$

$$= 0,249386835$$

$$z_2 = f(z_{net_2}) = \frac{1}{1 + e^{-z_{net_2}}}$$

$$= \frac{1}{1 + e^{-(-0,48363379)}}$$

$$= 0,381394427$$

Hitung semua keluaran jaringan di unit $y_k (k = 1, 2, \dots, m)$

$$y_{net_k} = v_{k0} + \sum_{j=1}^p z_j w_{kj}$$

$$= 0,0116146456849328 + ((0,249386835081723 * 0,980162801516388) + (0,381394426526083 * -0,648937509404328))$$

$$= 0,008553195$$

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

$$= \frac{1}{1 + e^{-0,008553195}}$$

$$= 0,502138286$$

Kembali lagi ke proses hitung semua keluaran dalam unit yang tersembunyi $z_j (j = 1, 2, \dots, p)$ tetapi menggunakan data uji dua, tiga dan seterusnya hingga menghasilkan hasil akhir di tabel 4.10 di bawah.

Tabel 4.10 Dari hasil prediksi berdasarkan data uji propagasi balik

HASIL PREDIKSI	TARGET	SELISIH	KUADRAT SELISIH
0,502138286	0,284615385	0,217523	0,047316212
0,456879664	0,161538462	0,295341	0,087226426
0,447307815	0,161538462	0,285769	0,081664123



0,446725914	0,161538462	0,285187	0,081331883
-------------	-------------	----------	-------------

Pada tabel 4.11 menunjukkan perbandingan nilai target dan prediksi yang didapatkan dari proses denormalisasi.

Persamaan yang digunakan : $x = \frac{(x-0,1)(x_{max}-x_{min})}{0,8} + x_{min}$

Tabel 4.11 Denormalisasi Data

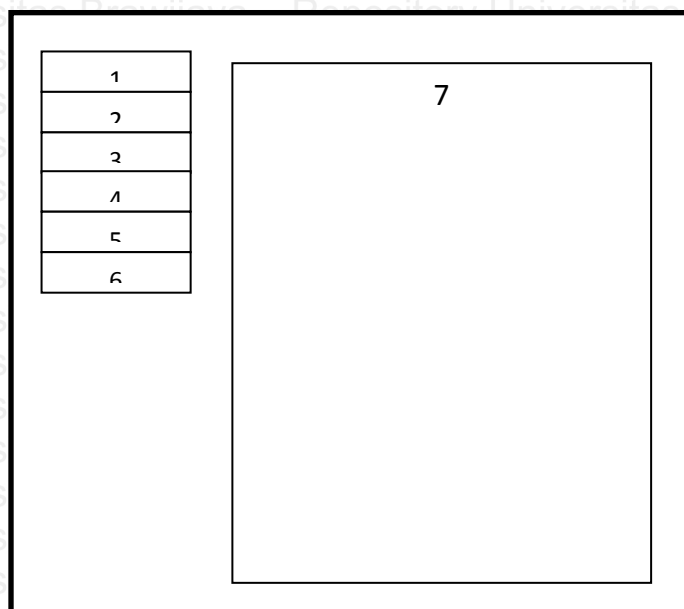
HASIL PREDIKSI	TARGET
7	4
6	2
6	2
6	2

4.4 Perancangan Antarmuka

Penghubung diantara pengguna dan juga sistem yang dapat melakukan komunikasi antara keduanya disebut antarmuka ataupun *interface*. Sistem antarmuka akan menerima dan mengolah *input* atau masukan yang diberikan oleh pengguna sehingga dihasilkan bentuk-bentuk informasi yang dapat dimengerti. Dalam sistem peramalan kapasitas pelayanan sepatu kaitanya dengan antarmuka dan spesifikasi akan dijelaskan dalam bagian ini.

4.4.1 Halaman Utama

di saat awal pengguna menjalankan sistem akan terlihat seperti ini, Gambar 4.15 akan menunjukkan data apa saja yang digunakan dalam sistem peramalan layanan sepatu.



Gambar 4.15 Antarmuka Pada Halaman Utama

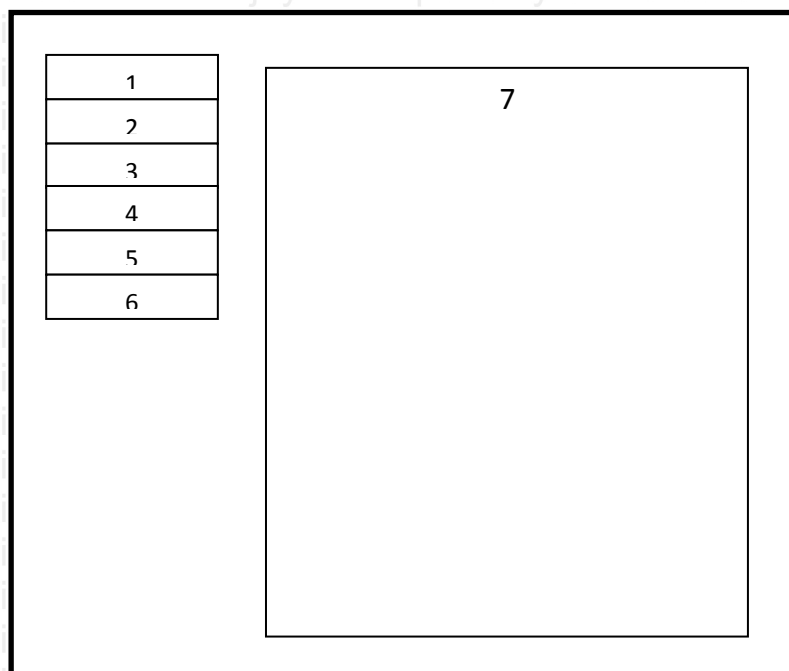


Keterangan :

1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test
6. Denormalisasi data
7. Data yang digunakan

4.4.2 Halaman Data Latih

Ketika tombol dari data latih pada halaman ini digunakan maka akan memunculkan halaman ini, hasil dari nilai normalisasi akan muncul ketika pengguna memilih menu ambil data latih dengan syarat sebelumnya sudah memasukan data yang diminta, seperti tanggal dimulainya data latih, lalu jumlah pola data dan juga data latih. Gambar 1.6 menunjukkan gambaran dari halaman data latih.



Gambar 4.16 Antarmuka Data latih

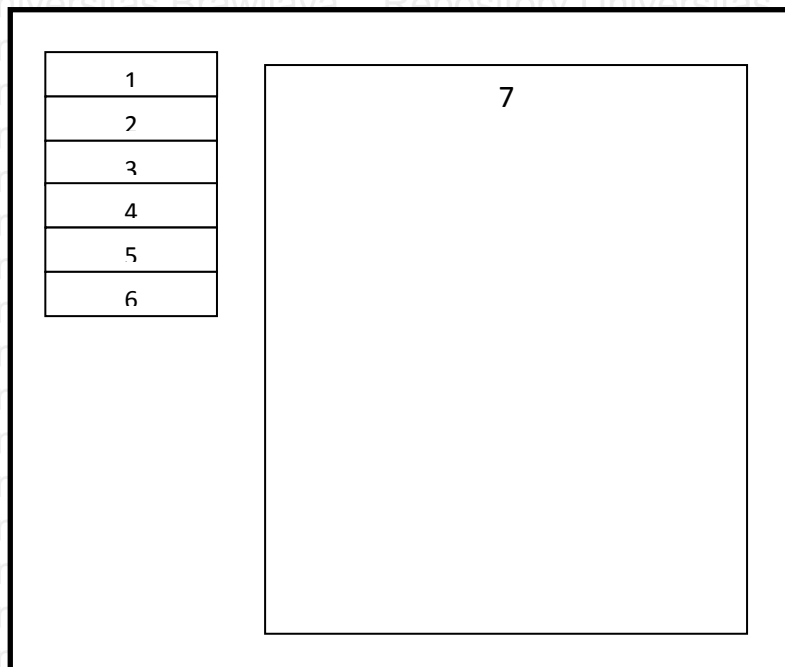


Keterangan :

1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test
6. Denormalisasi data
7. Data latih yang dipilih dan hasil normalisasi

4.4.3 Halaman Backpro-train

Pada saat tombol backpro-train maka akan muncul halaman seperti ini, nilai dari tiap-tiap parameter yang akan dipakai, akan diminta untuk dimasukkan. Hasil dari perhitungan dalam proses data latih akan muncul setelah dipilihnya menu mulai pelatihan. Gambar 4.17 menunjukkan gambar halaman antar muka backpro-train.



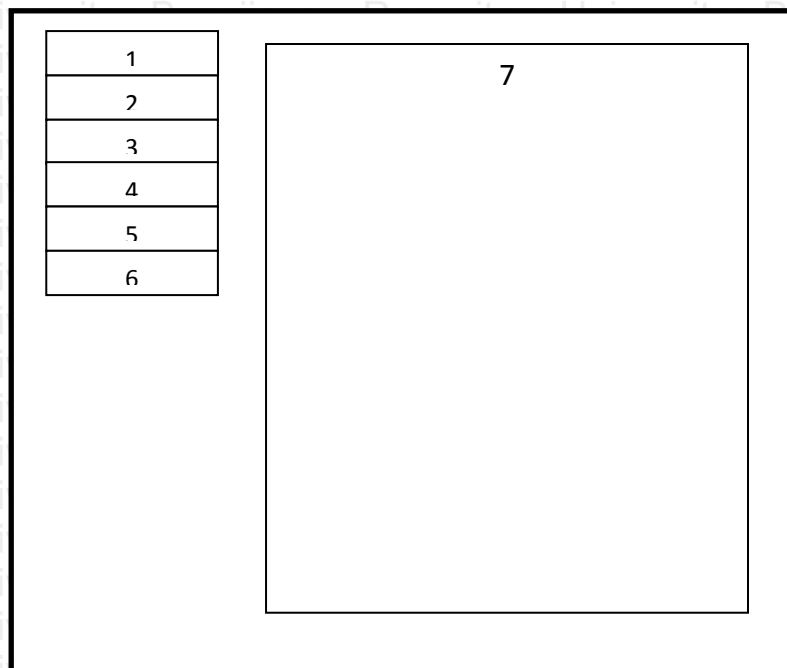
Gambar 4.17Antarmuka backpro-train



Keterangan :

1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test
6. Denormalisasi data
7. Nilai parameter dan hasil proses data latih

4.4.4 Data Uji



Gambar 4.18 Antramuka Data Uji

ketika panel atau tombol dari data uji diklik maka tampilan ini akan muncul, hasil daripada normalisasi dalam sistem ini akan ditampilkan jika data-data yang diminta seperti data tanggal mulainya data uji, lalu tiap-tiap pola data dan data uji telah dimasukkan. Hasil perhitungan peramalan akan dihitung dan ditampilkan dengan perhitungan propagasi mundur, jika dilanjutkan dengan memilih menu mulai uji. Gambar 4.18 ditunjukkan halaman data uji.

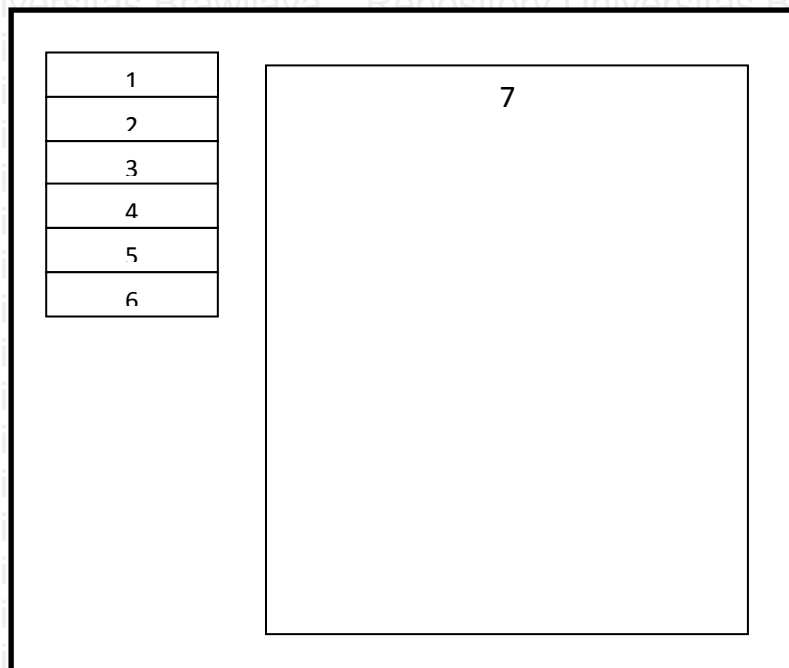


Keterangan :

1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test
6. Denormalisasi data
7. Data uji yang dipilih dan hasil normalisasi

4.4.5 Halaman Backpro-Test

Ketika tombol backpro-test dipilih maka akan muncul tampilan halaman seperti ini, terdapat tampilan dari hasil proses perhitungan data uji. Pada gambar 4.9 akan ditunjukkan tampilan antarmuka dari halaman.



Gambar 4.19 Antarmuka backpro-test

Keterangan :

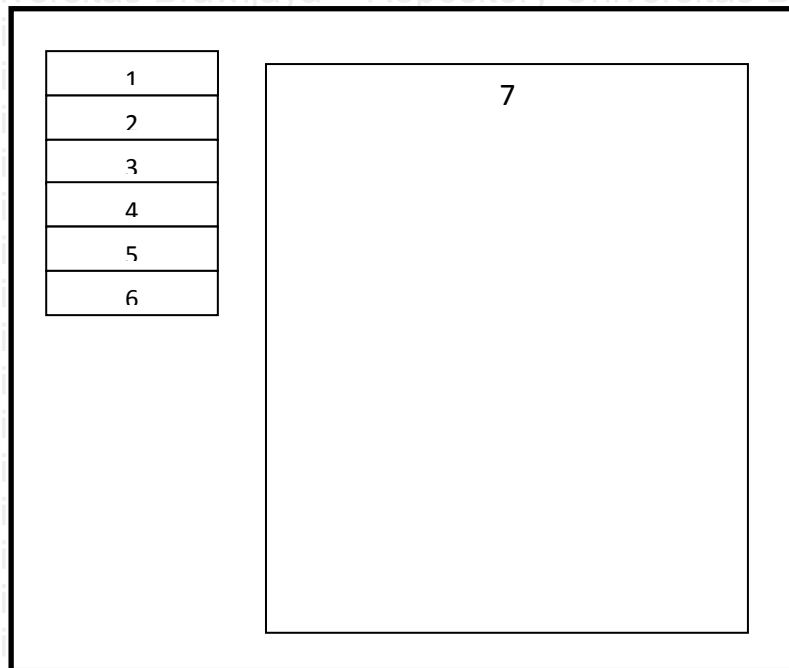
1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test



6. Denormalisasi data
7. Hasil dari proses data uji

4.4.6 Halaman Denormalisasi data

Ketika pengguna memilih tombol denormalisasi data, maka tampilan halaman akan seperti ini, prediksi jumlah layanan pada sepatu dan nilai MSE akan ditampilkan sesuai dengan jumlah pada data uji yang ada. Pada gambar 4.20 dapat dilihat tampilan dari halaman denormalisasi data.



Gambar 4.20 Antarmuka pada denormalisasi data

Keterangan :

1. Data digunakan
2. Data latih
3. Backpro-train
4. Data uji
5. Backpro test
6. Denormalisasi data
7. Hasil MSE dan prediksi jumlah layanan



BAB 5 IMPLEMENTASI

Sistem yang sudah selesai dibuat akan dijelaskan pada bab ini. Penjelasan pada bab ini adalah penerapan bagaimana sistem peramalan kapasitas layanan dan bagaimana antarmuka diterapkan.

5.1 Implementasi Program

Dalam bahasan sebelumnya sudah diberikan penjelasan tentang bagaimana sistem dirancang, dan pada bagian Implementasi sistem peramalan layanan kapasitas ini akan menjelaskan mengenai penerapan dari perancangan sistem menggunakan java sebagai bahasa pilihannya.

5.1.1 Inisialisasi awal

Tahapan pertama dari perhitungan backpropagation adalah inisialisasi, disini ada 2 inisialisasi yaitu inisialisasi dalam Source code 5.1 telah ditunjukkan bagaimana proses data latih dan juga data uji.

```

1 public class Olah_Data {
2     //1. attribute untuk menampung nilai maksimum dan minimum data->
3     untuk normalisasi data
4     double bahan_maksimum;
5     double bahan_minimum;
6     //2. Data seperti uji beserta latih yang telah dinormalisasi
7     diletakan pada atribut
8     double datamulai_normalisasi[][];
9     //3. attribute untuk denormalisasi data latih dan uji
10    double data_denormalisasi[];
11
12    //4. attribute untuk menampung data latih
13    double data_latihsp[][];
14    String label_data_latih[];
15    double target_latih[];
16
17    //5. attribute untuk menampung data uji
18    double data_uji[][];
19    String label_data_uji[];
20
21    //1. Fungsi-fungsi untuk mendapatkan data maksimum dan minimum
22    public void Cari_Maks_Minmulai_Data(double data[]){
23        this.bahan_maksimum = data[0];
24        this.bahan_minimum = data[0];

```



```

25
26     for(int spt=0; spt < data.length; spt++){
27         if (data[spt] > this.bahan_maksimum){
28             this.bahan_maksimum = data[spt];
29         } else if (data[spt] < this.bahan_minimum){
30             this.bahan_minimum = data[spt];
31         }
32     }
33 }
34
35 public double getBahan_maksimum(){
36     return bahan_maksimum;
37 }
38
39 public double getBahan_minimum() {
40     return bahan_minimum;
41 }
42 //Pencarian nilai batas atas dan bawah oleh fungsi-fungsi
43
44 //2. Pelaksanaan data latih menggunakan berbagai fungsi
45 public void Normalisasi_Data(double bahan_maksimum, double
46 bahan_minimum, double data[][]){
47     this.data_normalisasi_spt = new
48     double[data.length][data[0].length];
49     for(int pts=0; pts < data.length; pts++){
50         for(int cra=0; cra < data[0].length; cra++){
51             this.data_normalisasi[pts][cra] = ((0.8*(data[pts][cra]-
52             bahan_minimum))/(bahan_maksimum-bahan_minimum))+0.1;
53         }
54     }
55 }
56 public double[][] getDatamulai_normalisasi() {
57     return datamulai_normalisasi;
58 }
59 //Akhir fungsi-fungsi untuk normalisasi data latih
60
61 //3. Fungsi-fungsi untuk melakukan denormalisasi data
62 public void Denormalisasi_Dataspt(double bahan_maksimum, double
63 bahan_minimum, double datamulai_normalisasi[]){
64     this.data_denormalisasi = new
65     double[datamulai_normalisasi.length];
66     int pembulat = 0;
67     for(int i=0; i < datamulai_normalisasi.length; i++){

```



```

66         this.data_denormalisasi[i] = 0.0;
67         BigDecimal bd = new BigDecimal((((datamulai_normalisasi[i]-
68         0.1)*(bahan_maksimum-bahan_minimum))/0.8)+bahan_minimum));
69         bd = bd.setScale(pembulat, BigDecimal.ROUND_DOWN);
70         this.data_denormalisasi[i] = bd.doubleValue();
71     }
72 }
73
74     public double[] getData_denormalisasi() {
75         return data_denormalisasi;
76     }
77     //Akhir fungsi-fungsi untuk denormalisasi
78
79     //4. Fungsi untuk mendapatkan data latih
80     public void Set_Data_Latih(String tanggal_data, int jumlah_data, int
81     jumlah_fitur, double data[], String label[]){
82         this.data_latih = new double[jumlah_data][jumlah_fitur+1];
83         this.labelan_data_latih = new String[jumlah_data];
84         int titik_data = Cari_posisi_label(tanggal_data,label);
85         int idx_data = 0;
86         for(int r=titik_data; r < (titik_data+jumlah_data); r++){
87             int x = r;
88
89             for(int b=0; b < jumlah_fitur+1; b++){
90                 this.data_latih[idx_data][b] = data[x];
91                 x++;
92             }
93
94             this.labelan_data_latih[idx_data] = label[i];
95             idx_data++;
96         }
97     }
98 }
99
100     public double[][] getData_latih() {
101         return data_latih;
102     }
103
104     public String[] getLabel_data_latih() {
105         return label_data_latih;
106     }
107 }
108     //akhir fungsi untuk mendapatkan data latih

```




```

109
110 //5. Fungsi untuk mendapatkan data uji
111 public void Set_Data_Uji(String tanggal_data, int jumlah_dataspt,
112 int jumlah_fitur, double data[], String label[]){
113     this.data_ujispt = new double[jumlah_dataspt+1][jumlah_fitur+1];
114     this.label_data_uji = new String[data_ujispt.length];
115     int idx_data = 0;
116     int titik_data = Cari_posisi_label(tanggal_data,label);
117     for (int i=titik_dataspt; i <= (titik_dataspt+jumlah_dataspt);
118 i++){
119         int x = i;
120
121         if (i != (titik_data+jumlah_dataspt)){
122             for (int g=0; g < jumlah_fitur+1; g++){
123                 this.data_ujispt[idx_data][g] = data[x];
124                 x++;
125             }
126         } else {
127             for (int g=0; g < jumlah_fitur; g++){
128                 this.data_ujispt[idx_data][g] = data[x];
129                 x++;
130             }
131             this.data_ujispt[idx_data][jumlah_fitur] = 0.0;
132         }
133
134         this.label_data_ujispt[idx_data] = label[i];
135         idx_data++;
136     }
137 }
138 }
139
140 public double[][] getData_uji(){
141     return data_ujispt;
142 }
143
144 public String[] getLabel_data_uji() {
145     return label_data_uji;
146 }
147
148 //akhir fungsi untuk mendapatkan data uji
149
150 private int Cari_posisi_label(String tanggal_data, String[] label) {
151     int pos = 0;

```



```

152
153     for(int a=0; a < label.length; a++){
154         if (label[a].equalsIgnoreCase(tanggal_data)){
155             pos = a;
156         }
157     }
158
159     return pos;
160 }
161
162 }

```

Source Code 5.1 Inisialisasi awal

5.1.2 Normalisasi Data

Berbagai masukan yang didapat harus diolah dulu dengan cara dilakukan normalisasi untuk memperkecil range antar data yang ditulis dan ditampilkan pada source code 5.2.

```

1 public void Normalisasi_Dataspt(double bahan_maksimum, double
2 bahan_minimum, double data[][]){
3     this.data_normalisasispt = new
4     double[data.length][data[0].length];
5
6     for(int d=0; d < data.length; d++){
7         for(int m=0; m < data[0].length; m++){
8             this.data_normalisasispt[d][m] = ((0.8*(data[d][m]-
9             bahan_minimum))/(bahan_maksimum-bahan_minimum))+0.1;
10        }
11    }
12
13    public double[][] getData_normalisasispt() {
14        return data_normalisasispt;
15    }
16
17    //Akhir fungsi-fungsi untuk normalisasi data latihan

```

Source Code 5.2 Normalisasi data

5.1.3 Proses Data Latih

Propagasi balik digunakan pada tahapan ini untuk mengolah data latih

```

1 public void Init_Bobot_Jaringan_Dari_In_Ke_Hidd(double data[], int
2 jumlah_hidden_layer){
3     this.bobot_in_to_hidd = new
4     double[data[0].length][jumlah_hidden_layer];
5
6     double minimum = -1.0;

```



```

4      double maksimum = 1.0;
5      int pembulat = 1;
6
7      for(int f=0; f < this.bobot_in_to_hidd.length; f++){
8          for(int e=0; e < this.bobot_in_to_hidd[0].length; e++){
9
10             double acak = minimum + (maksimum - minimum) *
11             random.nextDouble();
12
13             BigDecimal bd = new BigDecimal(acak);
14             bd = bd.setScale(pembulat, BigDecim
15             al.ROUND_UP);
16
17             this.bobot_in_to_hidd[f][e] = bd.doubleValue();
18         }
19     }
20
21     public double[][] getBobot_in_to_hidd() {
22         return bobot_in_to_hidd;
23     }

```

Source Code 5.3 Proses penghitungan pada data latih

5.1.4 Proses Data Uji

Source code 5.4 menunjukkan bagaimana proses pengolahan data diuji dengan propagasi balik.

```

1     public void Init_Bobot_Jaringan_Dari_Hidd_Ke_Out(int
2     jumlah_hidden_layer){
3         this.bobot_hid_to_out = new double[jumlah_hidden_layer+1];
4         double minimumspt = -1.0;
5         double maksimumspt = 1.0;
6         int pembulat = 1;
7
8         for(int i=0; i < this.bobot_hid_to_out.length; i++){
9             double mix = minimumspt + (maksimumspt - minimumspt) *
10            random.nextDouble();
11            BigDecimal bd = new BigDecimal(mix);
12            bd = bd.setScale(pembulat, BigDecimal.ROUND_UP);
13            this.bobot_hid_to_out[i] = bd.doubleValue();
14        }
15    }
16
17    public double[] getBobot_hid_to_out() {

```



```

16         return bobot_hid_to_out;
17     }
18
19     /* 2 - 5 => Proses Backpro -> Propagasi Maju */
20     //2. fungsi hitung nilai Z_net
21     public void Hitung_ZNet(double bobot_in_to_hidd[ ][ ], double
22     data[ ][ ], int idx){
23         this.spt_znetspt = new double[bobot_in_to_hidd[0].length];
24
25         for(int i=0; i < spt_znet.length; i++){
26             this.spt_znet[i] = 0.0;
27             for(int j=0; j < data[0].length-1; j++){
28                 this.spt_znet[i] +=
29                 (data[idx][j]*bobot_in_to_hidd[j][i]);
30             }
31             this.spt_znet[i] +=
32             bobot_in_to_hidd[bobot_in_to_hidd.length-1][i];
33         }
34
35         public double[] getSpt_znet() {
36             return spt_znet;
37         }
38
39         //3. fungsi penghitungan Z_j
40         public void Hitung_Zj(double z_net[ ]){
41             this.z_j = new double[z_net.length];
42
43             for(int i=0; i < z_j.length; i++){
44                 this.z_j[i] = 1.0/(1.0+Math.pow(Math.exp(1), (z_net[i]*-
45                 1)));
46             }
47
48             public double[] getZ_j() {
49                 return z_j;
50             }
51
52             //4. fungsi hitung y_net
53             public void Hitung_Ynet(double z_j[ ], double
54             bobot_hid_to_out[ ]){
55                 this.y_net = 0.0;
56
57                 for (int k=0; k < z_j.length; k++){
58                     this.y_net += z_j[k]*bobot_hid_to_out[k];

```



```

58     }
59
60     this.y_net += bobot_hid_to_out[bobot_hid_to_out.length-1];
61 }
62
63     public double getY_net() {
64         return y_net;
65     }
66
67     //5. fungsi hitung y_k
68     public void Hitung_Yk(int idx, double target_latih[][],double
y_net){
69         this.y_k = 1.0/(1.0+Math.pow(Math.exp(1), (y_net*-1)));
70
71         this.error =
Math.pow((target_latih[idx][target_latih[0].length-1]-y_k), 2);
72     }
73
74     public double getY_k() {
75         return y_k;
76     }
77
78     public double getError() {
79         return error;
80     }
81
82     /* Akhir Proses Backpro -> Propagasi Maju */
83
84     /*6 - .. Proses Backpro -> Propagasi Mundur */
85
86     //6. Fungsi hitung faktor koreksi
87     public void Hitung_Faktor_Koreksi(int idx,double
Target_data[][], double y_k){
88         this.faktor_koreksi =
(((Target_data[idx][Target_data[0].length-1]-y_k)*(y_k))*(1-y_k));
89     }
90
91     public double getFaktor_koreksi() {
92         return faktor_koreksi;
93     }
94
95     //7. Fungsi hitung perubahan bobot
96     public void Hitung_Perubahan_Bobot(double z_j[], double
faktor_koreksi, double Alpha){
97         this.suku_perubahan_bobot = new double[z_j.length+1];
98

```



```

99     for(int i=0; i < suku_perubahan_bobot.length; i++){
100         if (i < suku_perubahan_bobot.length-1){
101             this.suku_perubahan_bobot[i] =
102             (Alpha*faktor_koreksi)*(z_j[i]);
103         } else {
104             this.suku_perubahan_bobot[i] = Alpha*faktor_koreksi;
105         }
106     }
107
108     public double[] getSuku_perubahan_bobot() {
109         return suku_perubahan_bobot;
110     }
111
112     //8. Fungsi hitung faktor kesalahan di tiap unit tersembunyi
113     public void Hitung_Kesalahan_Tiap_Unit(double faktor_koreksi,
114     double bobot_hid_to_out[]){
115         this.faktor_koreksi_hid_unit = new
116         double[bobot_hid_to_out.length-1];
117         for(int i=0; i < this.faktor_koreksi_hid_unit.length; i++){
118             this.faktor_koreksi_hid_unit[i] =
119             faktor_koreksi*bobot_hid_to_out[i];
120         }
121     }
122     public double[] getFaktor_koreksi_hid_unit() {
123         return faktor_koreksi_hid_unit;
124     }
125     //9. Fungsi hitung faktor kesalahan di unit tersembunyi
126     public void Hitung_Kesalahan_Unit_Tersembunyi(double
127     faktor_koreksi_hid_unit[], double z_j[]){
128         this.faktor_salah_hid_unit = new double[z_j.length];
129         for(int w=0; w < this.faktor_salah_hid_unit.length; w++){
130             this.faktor_salah_hid_unit[w] =
131             (faktor_koreksi_hid_unit[w]*z_j[w])*(1.0-z_j[w]);
132         }
133     }
134     public double[] getFaktor_salah_hid_unit() {
135         return faktor_salah_hid_unit;
136     }
137
138     //10. Fungsi perubahan bobot in ke hid
139     public void Hitung_Perubahan_Bobot_Ke_Unit_Hidden(int idx,
double data_latih_normalisasi[][], double faktor_salah_hid_unit[], av

```



```

140     double Alpha, double spbobot_jaringan[][]){
141
142         this.suku_perubahanilai_bobot_ke_unit_tersembunyi = new
143         double[bobot_jaringan.length][spbobot_jaringan[0].length];
144
145         for(int k=0; k < bobot_jaringan[0].length; k++){
146             for(int j=0; j < bobot_jaringan.length; j++){
147                 if(j < bobot_jaringan.length-1){
148                     this.suku_perubahanilai_bobot_ke_unit_tersembunyi[j][k] =
149                     (faktor_salah_hid_unit[k]*data_latih_normalisasi[idx][j])*Alpha;
150                 } else {
151                     this.suku_perubahanilai_bobot_ke_unit_tersembunyi[j][k] =
152                     faktor_salah_hid_unit[k]*Alpha*1;
153                 }
154             }
155         } //salah variable
156         public double[][]
157         getSuku_perubahanilai_bobot_ke_unit_tersembunyi() {
158             return suku_perubahanilai_bobot_ke_unit_tersembunyi;
159         }
160
161         //11. Fungsi perubahan bobot hid ke out
162         public void Hitung_Perubahan_Bobot_Ke_Unit_Out(double
163         suku_perubahan_bobot[]){
164             for(int i=0; i < suku_perubahan_bobot.length; i++){
165                 this.bobot_hid_to_out[i] += suku_perubahan_bobot[i];
166             }
167         }
168
169         public void Hitung_Perubahan_Bobot_Jaringan(double
170         suku_perubahan_bobot_jaringan[], double bobot_jaringan[][]){
171             for(int i=0; i < bobot_jaringan.length; i++){
172                 for(int j=0; j < bobot_jaringan[0].length; j++){
173                     this.bobot_in_to_hidd[i][j] =
174                     bobot_jaringan[i][j]+suku_perubahan_bobot_jaringan[i][j];
175                 }
176             }
177         }

```

Source Code 5.4 Proses Data Uji



BAB 6 PENGUJIAN DAN ANALISIS

Penjumlahan tiap-tiap neuron, penjumlahan lapisan tersembunyi juga penjumlahan iterasi merupakan bagian dari parameter propagasi balik yang akan diuji dalam bab ini. Dalam bab ini akan dibahas mengenai pengujian juga analisis dari penerapan sistem peramalan kapasitas layanan sepatu. Tujuan utama dari pengujian ini adalah mencari parameter bernilai paling baik menurut nilai dari angka MSE yang dihasilkan dari tiap-tiap hasil pengujian pada sistem.

6.1 Sistematika Pengujian

Masukan-masukan yang digunakan dalam sistem peramalan kapasitas layanan sepatu adalah untuk mencari parameter propagasi balik yang paling optimal dan memiliki nilai MSE terbaik perlu dilakukan pengujian, demi dapat menyelesaikan masalah peramalan kapasitas layanan. Masukan penelitian:

a. Tanggal

Tanggal Mulai latihan = 1 agustus 2018

Tanggal mulai uji = 9 agustus 2018

b. Data-data yang digunakan dalam sistem peramalan kapasitas layanan sepatu

Jumlah pada data uji = 9

Jumlah pada data latihan = 32

6.2 Analisis dan Pembahasan

Jumlah iterasi, lapisan tersembunyi dan jumlah syaraf merupakan aspek-aspek yang diuji

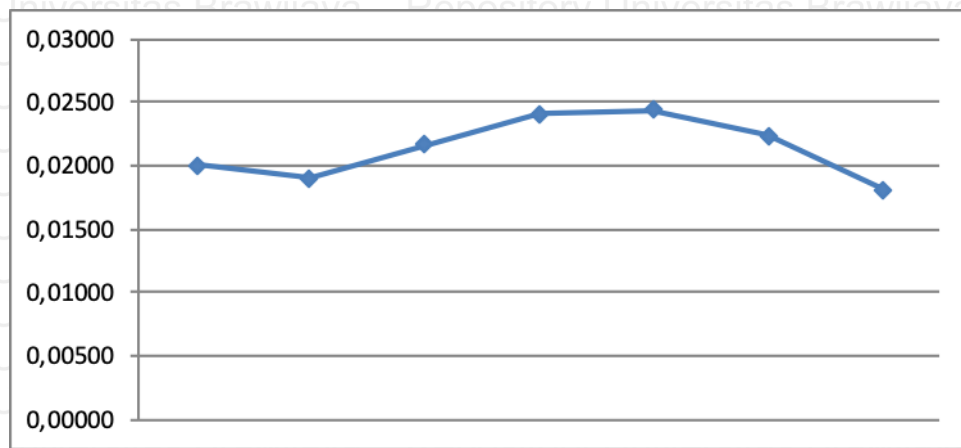
6.3 Pengujian dan Analisis Jumlah Neuron

Parameter jumlah Syaraf menjadi urutan pertama untuk diuji. Pengujian jumlah syaraf dilakukan untuk mencari tahu pengaruhnya pada nilai *MSE* yang dihasilkan. Penggunaan 4 syaraf atau neuron sampai 10 syaraf digunakan dalam pengujian. Pengambilan nilai rerata fitness dilakukan demi hasil yang valid dan dilakukan perulangan 10 kali. Dalam percobaan sistem peramalan ini dipergunakan nilai dari $\alpha = 0,5$, hidden layer = 2, dan iterasi = 150. Tabel 6.1 menunjukkan hasil uji coba tiap-tiap neuron.

**Tabel 6.1 Hasil Uji penjumlahan tiap-tiap neuron**

Jumlah Neuron	Nilai MSE Pengujian Ke										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
4	0,02112	0,02060	0,02062	0,02059	0,02062	0,02048	0,01652	0,01957	0,02098	0,01958	0,02007
5	0,01930	0,02039	0,02059	0,01918	0,01666	0,01846	0,01888	0,01927	0,01912	0,01844	0,01903
6	0,02014	0,01922	0,02054	0,02250	0,01810	0,02021	0,02474	0,02258	0,02360	0,02452	0,02161
7	0,02477	0,02111	0,02519	0,02493	0,02385	0,02357	0,02098	0,02509	0,02575	0,02532	0,02406
8	0,02309	0,02562	0,02401	0,02526	0,02592	0,02422	0,02329	0,02540	0,02503	0,02160	0,02434
9	0,02295	0,02167	0,02198	0,02045	0,02271	0,02346	0,02203	0,02239	0,02307	0,02283	0,02235
10	0,01923	0,01903	0,01929	0,01648	0,01797	0,01919	0,01921	0,01716	0,01773	0,01652	0,01818

Dari sekumpulan nilai-nilai yang digambarkan tabel 6.1 di atas dapat ditransformasikan menjadi sebuah bentuk gambaran grafik untuk memperjelas visualisasi dari nilai neuron dan MSE.

**Gambar 6.1 Hasil Uji Pada Jumlah tiap-tiap Neuron**

Data yang dihasilkan oleh grafik uji pada jumlah tiap-tiap neuron di atas dapat terlihat bahwa pengujian daripada nilai MSE cenderung naik dan turun, namun terdapat kecenderungan semakin besar neuron akan semakin baik pula nilai MSE nya. Hasil yang baik didapat pada neuron ke 7 yaitu MSE 0,01818

6.4 Pengujian dan Analisis Jumlah Hidden Layer

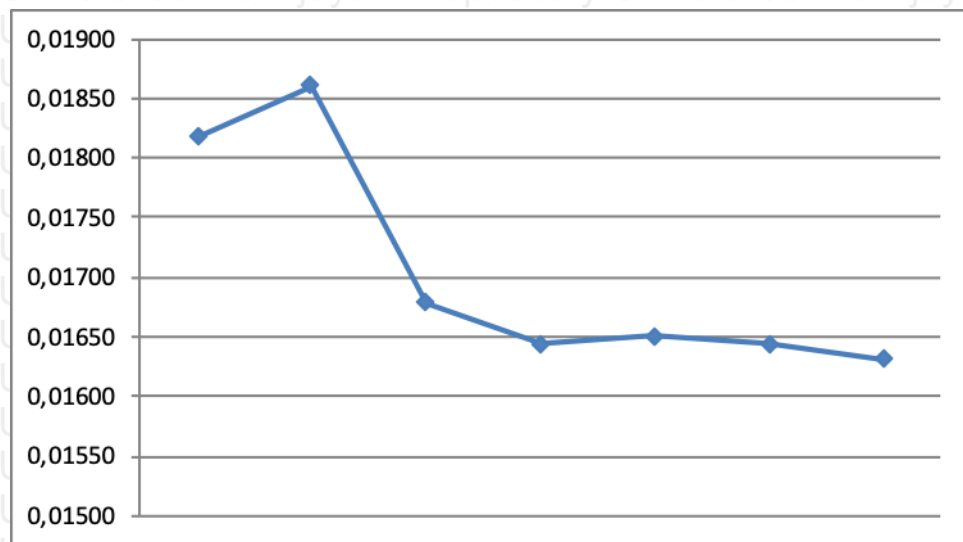
Parameter pada lapisan tersembunyi akan diuji pada tahap yang kedua. Tujuannya untuk mengetahui pengaruh nilai MES apakah berpengaruh terhadap besarnya jumlah lapisan tersembunyi. Untuk pengujian lapisan tersembunyi digunakan iterasi dengan kelipatan 1, dimulai 2-8. Hasil yang lebih valid bisa didapatkan dengan perulangan pada sistem peramalan kapasitas layanan sepatu sejumlah 10 tujuannya guna pengambilan nilai rerata MSE. Penggunaan $\alpha=0,5$, total 10 neuron didapat dengan menghasilkan MSE terkecil, dan iterasi sebanyak 150. Tabel 6.2 menggambarkan nilai dari sistem hasil uji coba iterasi.



Tabel 6.2 Hasil Pengujian pada tiap-tiap Hidden Layer

Hidden Layer	Nilai MSE Pengujian Ke										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
2	0,01923	0,01903	0,01929	0,01648	0,01797	0,01919	0,01921	0,01716	0,01773	0,01652	0,01818
3	0,01863	0,01868	0,01954	0,01681	0,01894	0,01860	0,01930	0,01728	0,01960	0,01863	0,01860
4	0,01603	0,01704	0,01824	0,01568	0,01515	0,01628	0,01820	0,01608	0,01756	0,01765	0,01679
5	0,01591	0,01460	0,01517	0,01931	0,01474	0,01505	0,01844	0,01851	0,01781	0,01485	0,01644
6	0,01820	0,01419	0,01693	0,01404	0,01632	0,01796	0,01697	0,01774	0,01652	0,01621	0,01651
7	0,01677	0,01532	0,01408	0,01763	0,01763	0,01708	0,01574	0,01802	0,01542	0,0166	0,01643
8	0,01509	0,01653	0,01729	0,01707	0,01550	0,01602	0,01598	0,01584	0,01831	0,01545	0,01631

Dari sekumpulan nilai yang tertera di tabel 6.1 di atas bisa ditransformasikan menjadi bentuk grafik untuk memperjelas visualisasi dari ujicoba lapisan tersembunyi pada MSE yang dalam gambar 6.2 terdapat hasil visualnya.



Gambar 6.2 Hasil pengujian pada tiap-tiap Hidden Layer

Dalam gambar data grafik yang tertera dapat terlihat nilai MSE cenderung naik dan turun, namun terdapat kecenderungan terlihat mulai percobaan hidden layer ke 2-8 semakin besar hidden layer akan semakin baik pula nilai MSE nya. Hasil yang baik didapat pada hidden layer ke 8 dengan MSE 0,01631.

6.5 Pengujian dan Analisis Dalam Jumlah Iterasi

Yang selanjutnya diuji dalam pengujian yang kedua adalah parameter dalam penjumlahan dari tiap-tiap iterasi. Tujuannya tidak lain mencari tau pengaruh tiap-tiap iterasi yang dijumlahkan terhadap nilai MSE. Jumlah kelipatan 10, mulai bilangan angka 120 hingga 180 digunakan untuk pengujian. Sistem peramalan kapasitas layanan ini dijalankan dengan 10 kali pengujian demi menghasilkan nilai yang valid dengan pengambilan rerata nilai MES. Digunakan alpha 0,5, jumlah neuron dari hasil pengujian sebelumnya yang menghasilkan MSE

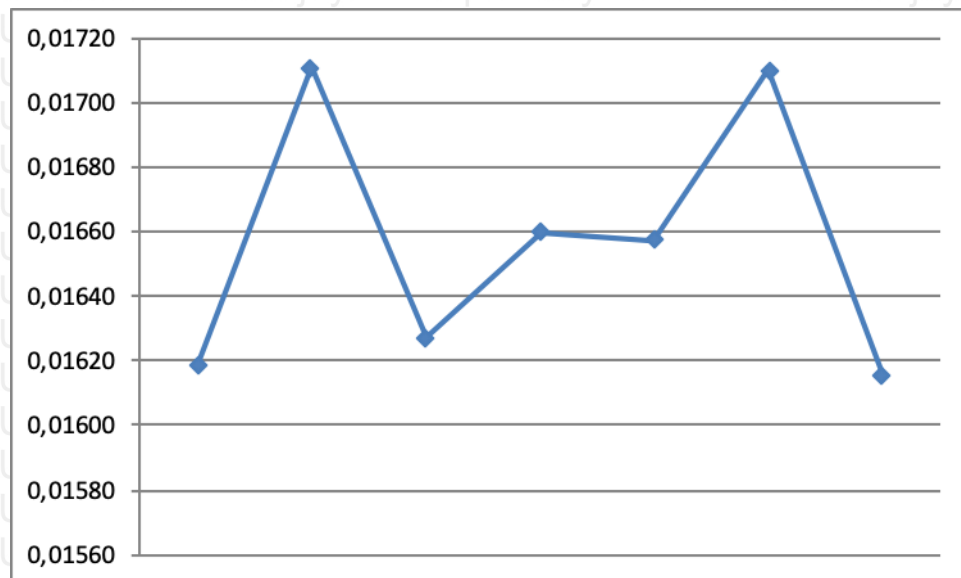


terkecil yaitu 10, jumlah lapisan tersembunyi sebelumnya adalah 8. Tabel 6.3 merupakan keluaran pengujian.

Tabel 6.3 Hasil pengujian nilai jumlah pada Iterasi

Iterasi	Nilai MSE Pengujian Ke										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
120	0,01544	0,01805	0,01457	0,01497	0,01652	0,01709	0,01693	0,01431	0,01575	0,01822	0,01618
130	0,01675	0,01810	0,01598	0,01773	0,01602	0,01598	0,01711	0,01823	0,01770	0,01746	0,01711
140	0,01558	0,01634	0,01620	0,01464	0,01777	0,01660	0,01589	0,01770	0,01505	0,01693	0,01627
150	0,01796	0,01651	0,01624	0,01673	0,01567	0,01658	0,01737	0,01483	0,01709	0,01700	0,01660
160	0,01522	0,01832	0,01538	0,01761	0,01819	0,01568	0,01484	0,01618	0,01774	0,01656	0,01657
170	0,01918	0,01786	0,01764	0,01699	0,01673	0,01651	0,01797	0,01600	0,01585	0,01625	0,01710
180	0,01712	0,01746	0,01729	0,01566	0,01463	0,01551	0,01625	0,01625	0,01541	0,01598	0,01616

Dari Tabel 6.3 dapat dibuat sebuah grafik untuk merepresentasikan uji coba jumlah iterasi dengan nilai *MSE*. Dan grafik hasil uji coba jumlah iterasi dapat dilihat pada gambar 6.3.



Gambar 6.3 Hasil pengujian nilai jumlah pada iterasi

Tidak stabilnya nilai *MSE* mulai percobaan dalam pengujian nilai jumlah pada iterasi terlihat di atas berupa naik turunnya garis. Dan tidak ada kecenderungan data hasil *MSE* selalu naik atau selalu turun. Hasil paling baik di angka 180 pada jumlah iterasi dengan nilai fitness 0,01616.

Dari data grafik di atas dapat terlihat bahwa nilai *MSE* cenderung naik dan turun, juga tidak ada kecenderungan untuk naik atau turun secara konstan. Hasil yang baik didapat pada iterasi ke 180 dengan fitness 0,01616.



6.6 Analisis Hasil Pengujian

Nilai MSE terbaik didapatkan dari penelitian ini dengan nilai 0,1616. Nilai ini dihasilkan dari nilai parameter dalam $\alpha = 0,5$ jumlah neuron = 10, jumlah hidden layer = 2 dan juga jumlah pada iterasi = 180.

Untuk mengukur ketepatan prediksi pada permasalahan prediksi jumlah layanan cuci sepatu dengan melihat nilai MSE terendah. MSE terendah menunjukkan tingkat kesalahan jumlah dalam prediksi kecil.



BAB 7 KESIMPULAN DAN SARAN

7.1 Kesimpulan

keseluruhan dalam proses dan implementasi algoritma untuk menyelesaikan masalah peramalan kapasitas layanan cuci sepatu, dapat ditarik kesimpulan berikut:

1. Algoritma backpropagation dapat diterapkan dalam masalah prediksi jumlah layanan cuci sepatu dengan menggunakan masukan tanggal, jumlah hidden layer, jumlah neuron, nilai alpha, maksimum iterasi, jumlah data yang di latih dan jumlah data yang melewati pengujian.
2. Untuk mengukur ketepatan prediksi pada permasalahan prediksi jumlah layanan cuci sepatu dengan melihat nilai *MSE* terendah. *MSE* terendah menunjukkan bahwa tingkat kesalahan jumlah dalam prediksi kecil. Nilai *MSE* terkecil yang didapatkan dari pengujian yaitu 0,01616.

7.2 Saran

Hal yang berguna untuk ditambah juga dikembangkan kedepan dari sistem peramalan kapasitas layanan sepatu ini adalah:

1. Di dalam program yang dijalankan data disimpan dalam bentuk file untuk melakukan edit ataupun update data harus merubah melalui file txt terlebih dahulu, akan lebih baik jikalau file disimpan di sebuah database.
2. Sistem dapat melakukan prediksi tetapi banyak parameter-parameter yang menghasilkan grafik tidak stabil, sehingga akan lebih baik jika ditambahkan dengan algoritma optimasi sehingga hasilnya lebih stabil dan angka error yang dihasilkan pada sistem yang dijalankan lebih kecil.



DAFTAR PUSTAKA

- Motawi, W. (2015). How Shoes are Made: A behind the scenes look at a real shoe factory. CreateSpace Independent Publishing Platform
- Pratt, L., & Wolley, L. (2008). Shoes. V&A: London.
- Anggriawan, Azry, Yulian Kasih, Ratna Juwita. 2017. Perencanaan Pendirian Usaha Jasa Pencucian, Perawatan dan Perbaikan Sepatu
- Rofik, Aunur. 2018. STRATEGI PEMASARAN JASA USAHA LAUNDRY SEPATU FARCLEAN SURABAYA. Prodi Manajemen Fakultas Ekonomi dan Bisnis. Universitas Narotama Surabaya
- Triyono, Andri, A. J. Santoso, Pranowo. 2016. Penerapan Metode Jaringan Syaraf Tiruan Backpropagation Untuk Meramalkan Harga Saham (IHSG). Magister Teknik Informatika. Universitas Atma Jaya Yogyakarta. 5528
- Cahyani, Fadilla P. 2018. Identifikasi Penyimpangan Tumbuh Kembang Anak Dengan Algoritme *Backpropagation*. *Fakultas Ilmu Komputer. Universitas Brawijaya Malang*
- Prawoko, Mahendro A. G. 2019. Prediksi Jumlah Permintaan Semen Menggunakan Jaringan Syaraf Tiruan *Backpropagation*. *Fakultas Ilmu Komputer. Universitas Brawijaya Malang*
- Rahayu, Dwi. 2018. Implementasi Metode *Backpropagation* Untuk Klasifikasi Kenaikan Harga Minyak Kelapa Sawit. *Fakultas Ilmu Komputer. Universitas Brawijaya Malang*
- Diputra, M. Ihsan. 2018. Prediksi Tingkat Keuntungan Usaha Peternakan Itik Alabio Petelur menggunakan Jaringan Syaraf Tiruan *Backpropagation* (Kasus di Kabupaten Hulu Sungai Utara Kalimantan Selatan) . *Fakultas Ilmu Komputer. Universitas Brawijaya Malang*
- Ridhani, M Najmi. 2017. Peramalan Dosis Pupuk Berdasarkan Karakteristik dan Lingkungan Tanaman Jeruk Siam Menggunakan Metode *Backpropagation*. *Fakultas Ilmu Komputer. Universitas Brawijaya Malang*
- Adam, S. P., Karras, D. A., Magoulas, G. D., & Vrahatis, M. N. (2014). Solving The Linear Interval olerance Problem for Weight Initialization of Neural Network
- Kusrini & Luthfi, E. T., 2009. *Algoritma Data Mining*. Yogyakarta: ANDI.
- Zulfikar & Budiantara, I. N., 2014. *Manajemen Dengan Pendekatan Komputasi Statistika*. Yogyakarta: Deepublish.

