

SINGLE LAYER PERCEPTRON DENGAN BACKPROPAGATION DALAM MEMPREDIKSI KELULUSAN MAHASISWA PADA MATA KULIAH PTI DI UNIVERSITAS XYZ

Shedriko

*Program Studi Informatika, Universitas Indraprasta PGRI
Jl. Nangka No 58C, Tanjung Barat, Jagakarsa, Jakarta Selatan – 12530
shedriko@gmail.com*

ABSTRAK

Universitas XYZ merupakan suatu universitas swasta nasional yang telah cukup lama berdiri, dan telah memiliki lima fakultas yang salah satunya merupakan pasca sarjana. Dengan biaya kuliah yang relatif terjangkau, menjadi daya tarik tersendiri bagi para calon mahasiswa sehingga universitas ini memiliki mahasiswa yang relatif banyak. Hal tersebut membuat satu mata kuliah pada suatu program studi harus diampu oleh lebih dari satu orang dosen. Penelitian ini menggunakan metode analisis kuantitatif dengan metodologi penyelesaian menggunakan Algoritma *Single Layer Perceptron* dengan *BackPropagation* terhadap kelulusan mahasiswa pada mata kuliah PTI (Pengantar Teknologi Informasi). Hasil dari penelitian ini berupa rumusan atau pola nilai *training* terhadap tiga atribut yaitu nilai tugas, nilai ujian tengah semester dan nilai ujian akhir, yang dapat digunakan untuk menghitung klasifikasi atau *prediksi* kelulusan terhadap data baru atau data *testing*. Pola rumusan tersebut dapat digunakan untuk klasifikasi terhadap kelas lainnya untuk mata kuliah yang sama atau berbeda dalam menentukan kelulusan mahasiswa.

Kata Kunci: *single, layer, perceptron, backpropagation, kelulusan.*

ABSTRACT

The University of XYZ is a well established university which has five faculties with one of them is post graduate. Charging with a very low cost of tuition fee has attracted so many high school graduation and make it becoming the university with the bulk number of students. It has caused one subject is taught by more than one lecturer. This research is using quantitative analysis method with BackPropagation Single Layer Perceptron Algorithm methodology in passing decision on PTI (Pengantar Teknologi Informasi) subject. The result gives pattern of training data towards three attributes, i.e. tasks, mid-semester and final exam score, which can classify or predict graduation for new data test. The pattern of this equation can also be used for other class classification, on the same or different subjects.

Keyword: single, layer, perceptron, backpropagation, graduation.

PENDAHULUAN

Universitas XYZ merupakan suatu universitas yang sudah cukup lama berdiri yang berlokasi di salah satu bilangan di DKI Jakarta. Universitas ini memiliki 5 fakultas dengan 1 di antaranya merupakan fakultas pasca sarjana. Perguruan tinggi yang telah meningkatkan statusnya dari sekolah tinggi menjadi universitas ini memiliki visi yang cukup unik, di samping sebagai penyelenggara pendidikan yang berguna untuk mencerdaskan kehidupan bangsa sehingga kelak dapat meningkatkan taraf kehidupan para peserta didik, namun juga berperilaku sebagai perguruan tinggi yang membantu mengentaskan kemiskinan yaitu dengan memberikan biaya pendidikan yang relatif murah dan terjangkau kepada para

peserta didik. Biaya pendidikan yang murah tidak serta merta menjadikan universitas tersebut murahan dengan kualitas pendidikan yang buruk dan tidak menjanjikan.

Penelitian dilakukan pada Jurusan Informatika yang terdiri dari sekitar 45an kelas per angkataannya. Kondisi ini menjadikan suatu mata kuliah harus diampu oleh sekitar 11an dosen per semesternya. Koordinasi yang baik antar dosen sangat diperlukan untuk menjamin kualitas pendidikan yang relatif seragam dari penyampaian materi oleh para dosen yang berbeda tersebut. Sedangkan mata kuliah yang dipilih bagi keperluan penelitian adalah Pengantar Teknologi Informasi (PTI).

“Dalam gundukan data tersimpan pengetahuan yang dapat merubah hidup seorang pasien atau bahkan merubah dunia” (Dean, 2014). Dengan semakin berkembangnya teknologi komputer, *big data* serta data *mining* dan *machine learning* telah menjadi penunjang yang diandalkan dalam teknologi informasi sebagai bagian dari hidup manusia yang tidak dapat terpisahkan (Smola & Vishwanathan, 2010). Secara garis besar, fungsi-fungsi dalam data *mining* dapat dibagi ke dalam 2 kategori (Han, Kamber, & Pei, 2012), yaitu:

1. *Descriptive*, bertugas mencirikan atau mengklasifikasikan sifat-sifat data tertentu ke dalam suatu set data tujuan
2. *Predictive*, bertugas melakukan induksi atau rangsangan terhadap suatu set data untuk dapat melakukan proses *prediksi* atau ramalan atau perkiraan

Kategori *predictive* dalam data *mining* memiliki sifat yang tumpang tindih dengan fungsi dari *machine learning*. *Machine learning* biasanya dibagi ke dalam 2 fase (Hertzmann & Fleet, 2012):

1. *Training*: suatu model dipelajari dari suatu kumpulan data *training* (data untuk pembelajaran)
2. *Application*: model yang ada tersebut digunakan untuk menentukan hasil dari suatu set data *test* baru

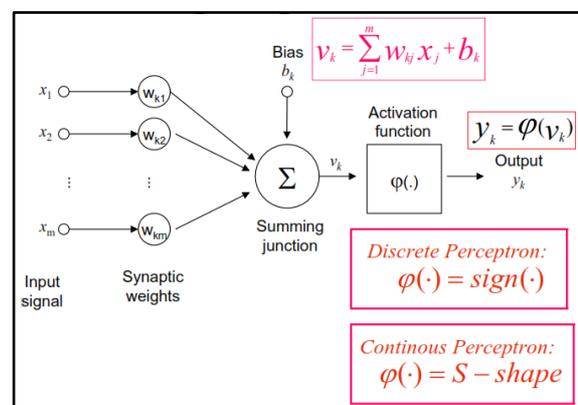
Berikut adalah beberapa tipe *machine learning* (Hertzmann & Fleet, 2012):

1. *Supervised Learning*, tipe pembelajaran dimana data *training* yang berisi jawaban yang benar diberikan terlebih dulu sebagai acuan
2. *Unsupervised Learning*, tipe pembelajaran dimana data yang diberikan harus dianalisa dan diklasifikasi terlebih dulu menurut pola-pola tertentu sebelum dinyatakan benar
3. *Reinforcement Learning*, tipe pembelajaran dimana suatu agen (berupa robot atau *controller*) mencari pola pembelajaran tertentu untuk memberikan aksi yang optimal berdasarkan pengalaman terdahulu
- 4.

Beberapa algoritma dasar yang merupakan *Supervised Learning* adalah *Naive Bayes*, *Nearest Neighbors Estimators*, *A Simple Classifier*, *Perceptron* dan *K-Means* (Simeone, 2018). Dalam penelitian ini algoritma yang

digunakan adalah *Single Layer Perceptron* dengan *BackPropagation*.

Perceptron adalah bentuk *neural network* paling sederhana yang terdiri dari *single neuron* dengan bobot dan bias yang *adjustable* (dapat diubah-ubah nilainya). Perceptron merupakan metode klasifikasi dengan hanya dua kelas, dimana pola vektornya diambil dari dua kelas terpisah dan selama training, algoritma ini akan melakukan konvergensi dan *positioning* posisi hasil pada bidang yang membagi dua kelas dengan memperbaiki nilai bobot terus menerus hingga posisi klasifikasi diperoleh (Bennamoun, 2003).



Gambar 1. *Single Layer Perceptron* (Bennamoun, 2003)

Dimana:

X_n : input ke n

W_{kn} : bobot input ke n

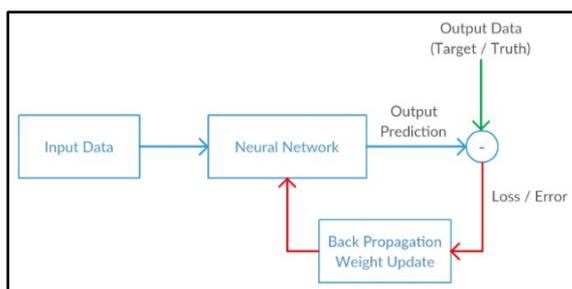
b_k : bias, simpangan

V_k : sigma, jumlah perkalian bobot dengan input ditambah bias

$\phi(.)$: sigmoid, fungsi aktivasi

y_k : output berupa prediksi, hasil dari aktivasi/sigmoid (bernilai 0 atau 1)

Propagation atau penyebaran merupakan aktivitas pengekseskuan prosedur tahap demi tahap. *Propagation* dibagi dua, yaitu *forward propagation* adalah proses yang dilakukan dengan arah maju dari mulai input hingga output, sedangkan *backward propagation* (*backpropagation*) adalah proses yang dilakukan dengan arah mundur untuk melakukan penghitungan nilai error dengan menghitung ulang nilai bobot dan biasanya (Lucas, 2017).



Gambar 2. Proses *BackPropagation* (Sena, 2017)

METODE PENELITIAN

Penelitian ini menggunakan metode analisis kuantitatif (Kothari, 2004) berdasarkan data-data yang diperoleh untuk kemudian memprosesnya dengan Metodologi Algoritma Perceptron. Penelitian yang berdasarkan analisis dilakukan dengan menggunakan fakta-fakta atau informasi yang sudah ada, kemudian menganalisisnya untuk membuat evaluasi kritis berkaitan dengan masalah tertentu (Kothari, 2004). Sedangkan penelitian kuantitatif merupakan penelitian yang didasarkan kepada angka-angka atau penghitungan kuantitas terhadap semua fenomena yang berkaitan dengan numerik (Kothari, 2004). Sehingga dapat disimpulkan bahwa penelitian ini dilakukan dengan melakukan analisis terhadap angka-angka berdasarkan rumusan tertentu terhadap suatu objek tertentu, dalam hal ini adalah *prediksi* kelulusan terhadap mahasiswa pada mata kuliah Pengantar Teknologi Informasi.

HASIL DAN PEMBAHASAN

Beberapa kriteria penilaian yang umum dilakukan pada kelulusan mahasiswa dalam suatu mata kuliah mencakup keseluruhan nilai dari kehadiran, tugas, ujian tengah semester, ujian akhir semester serta nilai-nilai penunjang lainnya seperti keaktifan di kelas dan kuis, dengan rincian sebagai berikut:

1. Nilai akhir harus mencapai nilai lebih besar atau sama dengan 56
2. Absensi harus memenuhi setidaknya 80 persen
3. Nilai tugas minimal 60
4. Menghadiri kuis yang diadakan
5. Aktif di kelas dengan bertanya atau menjawab pertanyaan-pertanyaan yang terlontar di kelas baik yang dikemukakan oleh dosen maupun mahasiswa pada sesi diskusi

Namun demikian, beberapa penilaian subyektif dapat saja terjadi dengan anggapan bahwa seorang mahasiswa mungkin lengah pada saat ujian, padahal dalam kegiatan belajar mengajar sehari-hari di kelas dosen melihat bahwa mahasiswa tersebut memiliki potensi yang cukup besar di perkuliahan tersebut. Berikut beberapa contoh yang mungkin terjadi dalam hal pemberian penilaian di kelas, yaitu:

1. Memenuhi nilai *total* minimum namun absensi kurang dari 80 persen, sehingga keputusan kelulusan dilihat pada nilai tugas yang tinggi, seperti lebih besar atau sama dengan 90
2. Memenuhi nilai *total* minimum namun absensi kurang dari 80 persen, sehingga keputusan kelulusan dilihat pada nilai keaktifan yang tinggi, seperti lebih dari 3 kali dalam mengajukan pertanyaan maupun menjawab, ditambah dengan selalu menghadiri kuis yang diadakan di kelas
3. Dan seterusnya

Nilai tugas biasanya mencakup beberapa nilai berikut ini:

1. Nilai absensi
Memberikan nilai dengan porsi 50 persen dari nilai tugas bila mahasiswa menghadiri seluruh perkuliahan yang biasanya berkisar sekitar 14 kali pertemuan
2. Nilai keaktifan di kelas
Biasanya dosen akan memberikan porsi hingga 30 persen dari nilai tugas bila mahasiswa aktif hingga bertanya atau menjawab sebanyak lebih dari atau sama dengan 3 kali
3. Nilai kuis
Nilai dengan porsi hingga 20 persen dari nilai tugas akan diberikan bila mahasiswa mendapatkan nilai yang sangat memuaskan dari soal yang diberikan

Adapun atribut yang digunakan sebagai *evidence* dalam Algoritma Perceptron pada penelitian ini terdiri dari 3 *item*, yaitu.

1. Nilai tugas
2. Nilai ujian tengah semester
3. Nilai akhir semester

Beberapa penampakan data *training* dimana mahasiswa dinyatakan tidak lulus dengan nilai target = 0 dan lulus dengan nilai target = 1 dapat dilihat pada tabel di bawah berikut ini.

Tabel 1. Beberapa data *training* yang menyatakan mahasiswa tidak lulus

1	Nilai Tugas	Nilai UTS	Nilai UAS	Kelulusan	Target
2	9,0	5,0	2,8	TIDAK LULUS	0
3	9,0	4,5	1,7	TIDAK LULUS	0
4	0,0	3,0	0,0	TIDAK LULUS	0
5	10,0	3,0	1,0	TIDAK LULUS	0
6	0,0	0,0	0,0	TIDAK LULUS	0
7	10,0	5,0	1,8	TIDAK LULUS	0
8	8,0	5,0	1,8	TIDAK LULUS	0
9	9,0	3,0	4,2	TIDAK LULUS	0
10	0,0	0,0	0,0	TIDAK LULUS	0
11	0,0	6,8	0,0	TIDAK LULUS	0
12	9,0	6,2	2,8	TIDAK LULUS	0
13	8,0	6,2	0,9	TIDAK LULUS	0
14	9,0	5,0	2,1	TIDAK LULUS	0
15	1,5	4,5	8,1	TIDAK LULUS	0
16	1,5	6,8	6,8	TIDAK LULUS	0
17	1,5	7,2	8,2	TIDAK LULUS	0
18	1,5	9,0	3,5	TIDAK LULUS	0
19	1,5	8,5	5,6	TIDAK LULUS	0
20	0,0	0,0	0,0	TIDAK LULUS	0
21	8,0	8,0	6,0	TIDAK LULUS	0

Tabel 2. Beberapa data *training* yang menyatakan mahasiswa lulus

1	Nilai Tugas	Nilai UTS	Nilai UAS	Kelulusan	Target
157	10,0	7,2	7,1	LULUS	1
158	10,0	6,2	4,7	LULUS	1
159	9,0	6,8	4,2	LULUS	1
160	10,0	8,0	6,2	LULUS	1
161	9,0	5,0	6,3	LULUS	1
162	9,0	8,5	7,2	LULUS	1
163	9,0	5,0	6,1	LULUS	1
164	10,0	7,2	6,3	LULUS	1
165	9,0	7,2	6,2	LULUS	1
166	10,0	7,2	7,5	LULUS	1
167	10,0	6,8	4,9	LULUS	1
168	10,0	7,2	6,3	LULUS	1
169	9,0	6,2	5,8	LULUS	1
170	10,0	5,0	4,2	LULUS	1
171	8,0	5,5	5,0	LULUS	1
172	8,0	7,2	6,3	LULUS	1
173	10,0	4,0	5,8	LULUS	1
174	8,0	7,2	5,9	LULUS	1
175	9,0	6,8	5,9	LULUS	1
176	10,0	6,2	4,8	LULUS	1

Beberapa nilai awal diberikan sebagai acuan untuk menghitung nilai perbaikan dari bobot dan bias, sebagai berikut:

$$\begin{aligned} \text{Nilai input ke 1} &= X_1 = 9,0 \\ \text{Nilai input ke 2} &= X_2 = 5,0 \\ \text{Nilai input ke 3} &= X_3 = 2,8 \\ \text{Bobot} &= W_1 = W_2 = W_3 = 0,5 \\ \text{Bias} &= b = 0,5 \\ \text{Learning rate} &= \eta = 0,1 \end{aligned}$$

dimana nilai bobot, bias dan *learning rate* diambil sembarang untuk kemudian dikoreksi

melalui iterasi hingga mendapatkan nilai terbaik dalam perhitungan data *testing*. Dari nilai-nilai awal tersebut diperoleh nilai awal lainnya sebagai berikut:

$$\text{Sigma} = V_k = (9,0 * 0,5) + (5,0 * 0,5) + (2,8 * 0,5) + 0,5 = 8,9$$

$$\text{Sigmoid} = y = 1 / (1 + \exp(-8,9)) = 0,999864$$

dimana:

$$\text{Sigmoid} = y = \frac{1}{1 + e^{-x}}$$

e merupakan eksponensial dengan besaran = 2,71828182846, sedangkan x adalah hasil penghitungan sigma di atas (Lucas, 2017). Kemudian dari nilai awal di atas kita hitung:

$$\text{error} = (0 - 0,999864)^2 = 0,999727$$

dimana:

$\text{error} = (\text{LabelTarget} - \text{Sigmoid})^2$ merupakan nilai yang akan dikecilkan atau minimalisir dengan *Stochastic Gradient Descent* (SGD) (MacKay, 2005). Lalu kita hitung:

$$\Delta W_1 = -2 * (0 - 0,999864) * 0,999864 * (1 - 0,999864) * 9,0 = 0,002453996$$

$$\Delta W_2 = -2 * (0 - 0,999864) * 0,999864 * (1 - 0,999864) * 5,0 = 0,001363331$$

$$\Delta W_3 = -2 * (0 - 0,999864) * 0,999864 * (1 - 0,999864) * 2,8 = 0,000763466$$

$$\Delta b = -2 * (0 - 0,999864) * 0,999864 * (1 - 0,999864) = 0,000272666$$

dimana:

$$\Delta W = \frac{\delta \text{error}}{\delta \text{Bobot}} = (\delta \text{error} / \delta \text{Sigmoid}) * (\delta \text{Sigmoid} / \delta \Sigma) * (\delta \Sigma / \delta \text{Bobot})$$

$$\Delta W = \frac{\delta \text{error}}{\delta \text{Bobot}} = -2 * (\text{LabelTarget} - \text{Sigmoid}) * \text{Sigmoid} * (1 - \text{Sigmoid}) * \text{input}$$

$$\Delta b = -2 * (\text{LabelTarget} - \text{Sigmoid}) * \text{Sigmoid} * (1 - \text{Sigmoid})$$

(Sena, 2017). Sedangkan *prediction* diperoleh dari pengkondisian nilai sigmoid, bila > 0,5 diberikan nilai 1 dan bila tidak sesuai dengan kondisi tersebut diberikan nilai 0.

Untuk data ke 2, perbedaan perhitungan terjadi hanya pada nilai bobot 1, bobot 2, bobot 3 dan bias, sedangkan perhitungan nilai-nilai lainnya sama dengan perhitungan pada data pertama. Perhitungan yang berbedanya adalah sebagai berikut:

$$\text{Bobot baru ke 1} = 0,5 - 0,1 * 0,002453996 = 0,4997546$$

$$\text{Bobot baru ke 2} = 0,5 - 0,1 * 0,001363331 = 0,499863667$$

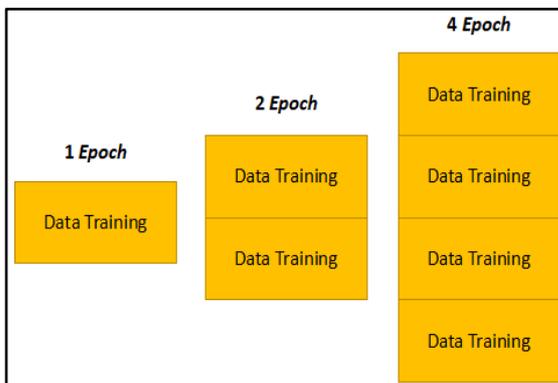
$$\text{Bobot baru ke 3} = 0,5 - 0,1 * 0,000763466 = 0,499923653$$

Nilai baru bias = $0,5 - 0,1 * 0,000272666 = 0,499972733$

dimana:

Bobot baru = Bobot lama - $\eta \frac{\delta_{error}}{\delta_{Bobot}}$ (Vincent, 2019). Untuk data ke 3 hingga data terakhir, dihitung seperti halnya perhitungan pada data ke 2. Semakin banyak data *training* yang tersedia maka akan semakin baik pula pembelajaran yang dilakukan oleh sistem dalam menentukan model yang terbaik.

Algoritma Perceptron mengenal adanya iterasi (*epoch*) untuk mendapatkan nilai bobot terbaik sehingga nilai “target” dan “prediksi” hasilnya akan sama persis. Deskripsi *epoch* dapat dilihat pada gambar berikut.



Gambar 3. Visualisasi *epoch* (Vincent, 2019)

Dalam *epoch* pertama, yang dilakukan adalah tahap awal dimana target dan prediksi disandingkan, sehingga dapat terlihat perbedaan antara nilai 0 dan 1 dari kedua parameter tersebut. Dengan melakukan *epoch* berulang kali, akan dapat dicapai tujuan untuk menyamakan nilai target dengan prediksi dengan perolehan nilai terbaik dari bobot dan bias. Pada penelitian ini, penulis harus melakukan *epoch* sebanyak 2.559 kali untuk mendapat nilai target dan prediksi yang sama, sehingga pada titik ini bobot yang terbaik diperoleh. Model data yang diambil adalah pada data terakhir dari *epoch* tersebut. Berikut adalah potongan tabel dari hasil *epoch* terakhir.

Tabel 3. Beberapa data terakhir dari *epoch* ke 2.559

1	x1	x2	x3	kelulusan	target	prediction	weight1	weight2	weight3	bias
447805	9,0	7,6	4,9	LULUS	1	1	1,529763	1,270252	1,420735	-24,5098
447806	10,0	8,0	7,1	LULUS	1	1	1,529777	1,270263	1,420743	-24,5098
447807	10,0	7,2	7,1	LULUS	1	1	1,529777	1,270263	1,420743	-24,5098
447808	10,0	6,2	4,7	LULUS	1	1	1,529777	1,270263	1,420743	-24,5098
447809	9,0	6,8	4,2	LULUS	1	1	1,529822	1,270292	1,420764	-24,5098
447810	10,0	8,0	6,2	LULUS	1	1	1,530567	1,270854	1,421111	-24,5097
447811	9,0	5,0	6,3	LULUS	1	1	1,530567	1,270854	1,421111	-24,5097
447812	9,0	8,5	7,2	LULUS	1	1	1,530753	1,270958	1,421242	-24,5097
447813	9,0	5,0	6,1	LULUS	1	1	1,530753	1,270958	1,421242	-24,5097
447814	10,0	7,2	6,3	LULUS	1	1	1,531077	1,271138	1,421461	-24,5097
447815	9,0	7,2	6,2	LULUS	1	1	1,531077	1,271138	1,421461	-24,5097
447816	10,0	7,2	7,5	LULUS	1	1	1,531078	1,271139	1,421462	-24,5097
447817	10,0	6,8	4,9	LULUS	1	1	1,531078	1,271139	1,421462	-24,5097
447818	10,0	7,2	6,3	LULUS	1	1	1,531083	1,271142	1,421465	-24,5097
447819	9,0	6,2	5,8	LULUS	1	1	1,531083	1,271142	1,421465	-24,5097
447820	10,0	5,0	4,2	LULUS	1	1	1,531112	1,271167	1,421488	-24,5097
447821	8,0	5,5	5,0	LULUS	1	1	1,534497	1,272856	1,422906	-24,5093
447822	8,0	7,2	6,3	LULUS	1	1	1,558753	1,289532	1,438067	-24,5063
447823	10,0	4,0	5,8	LULUS	1	1	1,558759	1,289537	1,438071	-24,5063
447824	8,0	7,2	5,9	LULUS	1	1	1,558962	1,289619	1,438189	-24,5063
447825	9,0	6,8	5,9	LULUS	1	1	1,558979	1,289633	1,438201	-24,5063
447826	10,0	6,2	4,8	LULUS	1	1	1,558981	1,289635	1,438203	-24,5063

Dari *epoch* ke 2.559 tersebut, diambil nilai pada baris terakhir sebagai berikut:

$$W_1 = 1,558981271$$

$$W_2 = 1,28963516$$

$$W_3 = 1,438202882$$

$$b = -24,50626357$$

Nilai-nilai tersebut di atas, akan digunakan untuk penghitungan data *testing* sebagai pengujiannya.

Pengujian dilakukan dengan memasukkan data *testing* dengan beberapa skenario yang merupakan perpaduan nilai dari data input ke 1, 2 dan 3. Penulis mengambil skenario yang merupakan nilai ambang lulus dan tidak lulus dari ketiga data input dimana model data tersebut erat kaitannya dengan data *training* yang telah diberikan sebelumnya. Tabel berikut di bawah menunjukkan data *testing* dan hasilnya.

Tabel 4. Data *testing* (*prediction* = 0 – tidak lulus, = 1 – lulus)

1	x1	x2	x3	prediction	weight1	weight2	weight3	bias	sigma	sigmoid
2	5,5	5,8	5,8	0	1,558981	1,289635	1,438203	-24,5063	-0,11041	0,472427
3	5,8	5,5	5,8	0	1,558981	1,289635	1,438203	-24,5063	-0,0296	0,4926
4	5,8	5,8	5,5	0	1,558981	1,289635	1,438203	-24,5063	-0,07417	0,481465
5	5,6	5,8	5,8	1	1,558981	1,289635	1,438203	-24,5063	0,045492	0,511371
6	5,8	5,6	5,8	1	1,558981	1,289635	1,438203	-24,5063	0,099361	0,52482
7	5,8	5,8	5,6	1	1,558981	1,289635	1,438203	-24,5063	0,069648	0,517405

SIMPULAN DAN SARAN

Hasil penelitian ini menunjukkan nilai akurasi yang tinggi yang dapat diperoleh dalam menggunakan Algoritma *Single Layer Perceptron* dalam memprediksi kelulusan mahasiswa. Dengan skenario yang menyertakan subyektifitas penilaian dosen pada data, diperoleh hasil bahwa nilai LULUS (*prediction* = 1) didapat bila nilai minimum dari dua nilai input adalah 5,8 dan nilai lainnya minimum 5,6. Sedangkan bila nilai lainnya lebih kecil dari 5,6 maka hasilnya adalah TIDAK LULUS (*prediction* = 0). Hasil tersebut dapat lebih akurat dengan nilai minimum

kelulusan 5,6 untuk ketiga nilai input bila nilai subyektifitas dosen ditiadakan. Penelitian ini juga dapat dijadikan acuan sebagai penelitian lanjutan menggunakan berbagai algoritma *supervised machine learning* lainnya untuk membandingkan hasil yang diperoleh dengan hasil pada penelitian ini.

DAFTAR PUSTAKA

- Bennamoun, M. (2003). *Single Layer Perceptron (SLP) Classifiers*. Lecture Slide of Neural Computation, University of Western Australia.
- Dean, J. (2014). *Big data, data mining, and machine learning [internet resource]: value creation for business leaders and practitioners*. John Wiley & Sons Inc.
- Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (Third). <https://doi.org/10.1016/C2009-0-61819-5>
- Hertzmann, A., & Fleet, D. (2012). *Machine Learning and Data Mining*. Computer Science Department, University of Toronto.
- Kothari, C. R. (2004). *Research Methodology, Methods & Techniques* (Second Rev). [https://doi.org/10.1016/0022-460X\(82\)90541-7](https://doi.org/10.1016/0022-460X(82)90541-7)
- Lucas, C. (2017). *Multilayer Perceptron and Backpropagation - slide.pdf*. School Of Informatics, University of Edinburgh.
- MacKay, D. J. C. (2005). *Information Theory, Inference and Learning Algorithms*. Cambridge University Press.
- Sena, S. (2017). Pengenalan Deep Learning Part 3: BackPropagation Algorithm. Retrieved September 2, 2020, from medium.com website: <https://medium.com/@samuelsena/pengenalan-deep-learning-part-3-backpropagation-algorithm-720be9a5fbb8>
- Simeone, O. (2018). *A Brief Introduction to Machine Learning for Engineers* (p. 237). p. 237. <https://doi.org/10.1561/20000000102>
- Smola, A., & Vishwanathan, S. V. N. (2010). *Introduction to Machine Learning* (p. 234). p. 234. https://doi.org/10.1007/978-1-62703-748-8_7
- Vincent, M. (2019). Machine Learning: Single Layer Perceptron. Retrieved September 2, 2020, from medium.com website: <https://medium.com/@vincentmichael089/machine-learning-1-single-layer-perceptron-9d94c62f1970>