

Analisis Jaringan Syaraf Tiruan *Backpropagation* untuk Prediksi Kelas Tanah , N-SPT, dan Kohesi Tak Terdrainase

Ridwan Setiadi

Jurusan Teknik Sipil Fakultas Teknik Sipil dan Perencanaan Universitas Gunadarma
e-mail: ridwansetiadi94@gmail.com

Sri Wulandari

Jurusan Teknik Sipil Fakultas Teknik Sipil dan Perencanaan Universitas Gunadarma
e-mail: sri_wulandari@staff.gunadarma.ac.id

Abstrak

Jaringan syaraf tiruan (JST) adalah salah satu alat yang dapat digunakan untuk memprediksi sesuatu yang terlalu sulit untuk dimodelkan atau terlalu rumit diprogram melalui algoritma komputer biasa. Pada dasarnya, JST dibangun lalu dilatih untuk mengenali pola data pelatihan yang terdiri dari input dan target. Idealnya, JST yang sukses memiliki nilai kuadrat error rata-rata atau mean squared-error (MSE) yang kecil dan jumlah data pelatihan yang berhasil dikenali besar. Ada tiga JST yang dibuat dalam penelitian ini, yaitu JST klasifikasi tanah, JST prediksi N-SPT, dan JST prediksi kohesi tak terdrainase. JST klasifikasi tanah memiliki MSE pelatihan senilai 0,0351 dan mampu mengenali 56 dari total 57 atau 98,2% data pelatihan sementara MSE pengujiannya senilai 0,6534 dan mampu mengenali 6 dari 10 atau 60% data pengujian. JST prediksi N-SPT memiliki MSE pelatihan senilai 0,368 dan mampu mengenali 29 dari 37 atau 78,38% data pelatihan sementara MSE pengujiannya senilai 1,4697 dan mampu mengenali 6 dari 10 atau 60% data pengujian. JST prediksi kohesi tak terdrainase memiliki MSE pelatihan senilai 0,0059 dan mampu mengenali 27 dari 28 atau 96,43% data pelatihan sementara MSE pengujiannya senilai 0,0225 dan mampu mengenali 9 dari 10 atau 90% data pengujian.

Kata-kata Kunci: Jaringan syaraf tiruan, klasifikasi tanah, kohesi tak terdrainase, kuadrat error rata-rata, N-SPT

Abstract

Artificial Neural Network (ANN) is a tool which can be used to predict something too complicated to be modeled or to be programmed by using ordinary computer algorithm. Basically, ANN is built then trained to recognize pattern of training data consisting of inputs and targets. Ideally, a successful ANN has low mean squared-error (MSE) and high amount of recognizable training data. There are three ANN built, they are: classifying-soil-class ANN, predicting-N-SPT ANN, and predicting-undrained-cohesion ANN. MSE from training session of classifying-soil-class ANN is 0.0351 and it can recognize 56 of 57 or 98.2% training data while MSE from testing session is 0.6534 and it can recognize 6 of 10 or 60% testing data. MSE from training session of predicting-N-SPT ANN is 0.368 and it can recognize 29 of 37 or 78.38% training data while MSE from testing session is 1.4697 and it can recognize 6 of 10 or 60% testing data. MSE from training session of predicting-undrained-cohesion ANN is 0.0059 and it can recognize 27 of 28 or 96.43% training data while MSE from testing session is 0.0225 and it can recognize 9 of 10 or 90% testing data.

Keywords: Artificial neural network, mean squared-error, N-SPT, soil class, undrained cohesion

1. Pendahuluan

Pemanfaatan Jaringan Syaraf Tiruan (JST) atau juga sering disebut *Artificial Neural Network* (ANN) untuk kepentingan prediksi telah banyak dilakukan, baik untuk kepentingan bisnis maupun pendidikan. Contoh penggunaan JST untuk kepentingan bisnis antara lain: merekomendasikan suatu konten di dunia maya (seperti pakaian, aksesoris, video, atau berita) kepada pengguna berdasarkan data yang diberikan pengguna dan sejarah penelusuran *item* yang dilakukan pengguna.

Ada banyak penelitian yang menggunakan JST atau ANN untuk memprediksi sesuatu yang biasanya terlalu rumit untuk dimodelkan. Manyuk Fauzi dan Minarni Nur Trilita (2005) menulis “Aplikasi *Artificial Neural Network* untuk Peramalan Aliran Sungai Blega”. F.

Farrokhzad, A. J. Choobbasti, dan A. Barari (2010) menulis “*Artificial Neural Network Model for Prediction of Liquefaction Potential in Soil Deposits*”. Putu Doddy Heka Ardana (2013) menulis “Aplikasi Jaringan Syaraf Tiruan (*Artificial Neural Networks*) dalam Modelisasi Curah Hujan Limpasan dengan Perbandingan Dua Algoritma Pelatihan (Studi Kasus: Das Tukad Jogading) (139a)”.

Tujuan penulisan ini antara lain: mendesain JST untuk mengenali kelas tanah *Unified Soil Classification System* (USCS); mendesain JST untuk mengenali nilai *N* dari uji *Standard Penetration Test* (SPT) atau sering disebut N-SPT; dan mendesain JST untuk mengenali nilai kohesi tak terdrainase atau biasa disebut CU. JST pada penulisan ini dibuat dalam aplikasi MATLAB 2013a.

2. Tinjauan Pustaka

2.1 Jaringan syaraf tiruan

Jaringan Syaraf Tiruan, pada dasarnya, dapat dianggap seperti fungsi matematika. Fungsi tersebut dibangun berdasarkan data pelatihan, fungsi aktivasi, fungsi performa, dan fungsi pelatihan yang digunakan pada JST tersebut.

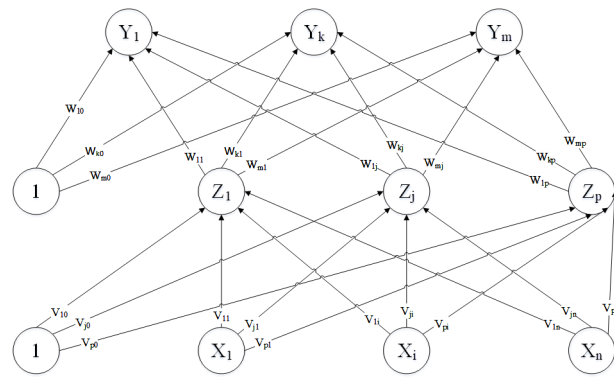
Inisialisasi JST dimulai dari menentukan jumlah *layer*, jumlah *neuron* pada tiap *layer*, fungsi aktivasi pada tiap *layer*, dan bobot yang menghubungkan suatu *neuron* dengan *neuron* yang ada pada *layer* berikutnya. Setelah proses inisialisasi selesai JST harus dilatih dengan memasukkan data pelatihan yang terdiri dari *input* pelatihan dan target pelatihan. Tujuan pelatihan tersebut adalah membuat JST mengenali pola *input-target*.

Proses pelatihan JST, pada dasarnya, adalah menerima *input* pelatihan lalu menghitung *output*-nya. Dari *output* tersebut, kesalahan (*error*) prediksi dihitung dengan fungsi performa (misalnya dengan fungsi MSE). Jika kesalahan tersebut lebih dari batas yang ditentukan, bobot dan bias JST tersebut akan dimodifikasi lalu proses perhitungan *output* dilakukan kembali. Pengulangan (iterasi) tersebut diulang hingga salah satu batas yang ditentukan terpenuhi. Batas tersebut dapat berupa batasan kesalahan minimum, batasan waktu maksimum, maupun batasan iterasi maksimum. Batasan tersebut ditentukan untuk mencegah proses pelatihan berlangsung terlalu lama atau bahkan mencegah pengulangan yang tak terhingga.

Fungsi pelatihan dan fungsi aktivasi dipilih berdasarkan format *output* JST. Jika tujuan JST adalah klasifikasi, format *output* yang cocok adalah vektor yang seluruh komponennya bernilai nol (0) kecuali satu komponen yang bernilai satu (1) sehingga fungsi pelatihan yang cocok adalah 'trainseg' dan fungsi aktivasi yang cocok adalah 'tansig'. Jika tujuan JST adalah memprediksi nilai tunggal atau skalar, fungsi pelatihan dan fungsi aktivasi yang cocok masing-masing adalah 'trainlm' dan 'tansig'.

Gambar 1 menunjukkan arsitektur JST secara umum. *Layer* (lapisan) paling bawah adalah *input layer* dan lingkaran-lingkaran yang ada di *layer* tersebut adalah *input neuron*. *Input layer* hanya berjumlah satu sedangkan jumlah *neuron*-nya bergantung pada jumlah set *input* yang digunakan JST tersebut.

Layer kedua dari bawah adalah *hidden layer* dan lingkaran-lingkaran yang ada di dalamnya adalah *neuron* pada *layer* tersebut. *Hidden layer* dapat berjumlah satu atau lebih. JJ Siang (2005) menyatakan bahwa hasil teoritis yang didapat menunjukkan bahwa jaringan dengan sebuah *layer* tersembunyi sudah cukup bagi *backpropagation* untuk mengenali sembarang perkawanan antara masukan dan target dengan tingkat ketelitian yang ditentukan tetapi penambahan jumlah *layer* tersembunyi kadang membuat pelatihan lebih mudah.



Gambar 1. Arsitektur backpropagation (JJ Siang, 2005)

Layer paling atas adalah *output layer* dan lingkaran-lingkaran yang ada di *layer* tersebut adalah *output neuron*. *Output layer* hanya berjumlah satu sedangkan jumlah *neuron*-nya bergantung pada jumlah set *output* yang dikeluarkan JST tersebut. Cara kerja JST dijelaskan sebagai berikut.

1. Mentransformasi nilai *input* yang berinterval besar menjadi interval kecil dengan fungsi transformasi tertentu, misalnya dari interval bilangan real positif (0, inf) menjadi interval (0,1). Proses ini terintegrasi dan dijalankan secara otomatis secara *default* saat membangun JST di MATLAB 2013a.
2. Memasukkan nilai atau set *input* yang telah ditransformasi ke *input neuron*.
3. Menghitung nilai *neuron* pada lapisan di atasnya melalui cara berikut.
 - a. Menghitung nilai 'net' melalui **Persamaan (1)** berikut ini.

$$net = b + \sum_i x_i w_i \quad (1)$$
 - b. Menghitung nilai *neuron* 'f(net)' melalui fungsi aktivasi berikut.

$$f(net) = \frac{1}{1 + e^{-(net)}} \quad (2)$$
 atau

$$f(net) = \frac{2}{1 + e^{-(net)}} - 1 \quad (3)$$
 atau

$$f(net) = net \quad (4)$$
4. Mengulangi tahap 3 untuk menghitung nilai *neuron* pada *hidden layer* berikutnya, bila *hidden layer* berjumlah lebih dari satu, sampai *neuron* pada *output layer*.

Persamaan (2) dikenal dengan nama fungsi sigmoid biner, **Persamaan (3)** dikenal dengan nama fungsi sigmoid bipolar, sedangkan **Persamaan (4)** dikenal dengan nama fungsi identitas. Dalam MATLAB 2013a, fungsi sigmoid biner dikenal dengan nama 'logsig', fungsi sigmoid bipolar dikenal dengan nama 'tansig', dan fungsi identitas dikenal dengan nama 'purelin'. Secara *default*, fungsi aktivasi JST dalam MATLAB 2013a untuk perintah inisialisasi 'patternnet'

adalah fungsi sigmoid bipolar. Jika JST diinisialisasi melalui perintah 'feedforwardnet', fungsi aktivasi pada *output layer* adalah fungsi identitas sedangkan pada *layer* lainnya, fungsi aktivasinya adalah fungsi sigmoid bipolar.

Pada **Gambar 1**, tiap *neuron* (kecuali *input neuron*), memiliki panah yang menghubungkan *neuron* tersebut dengan *neuron* sebelumnya. Nilai *neuron* 'f(net)' ditentukan dari nilai *neuron* sebelumnya dan bobot yang berada pada panah yang menghubungkan dengan *neuron* sebelumnya. Misalnya, jika **Persamaan (1)** diaplikasikan pada **Gambar 1**, nilai 'net' untuk *neuron* Z_1 didapatkan melalui **Persamaan (5)**.

$$net = 1 \times v_{10} + X_1 \times v_{11} + X_i \times v_{1i} + X_n \times v_{1n} \quad (5)$$

Nilai 'net' dari **Persamaan (5)** kemudian dimasukkan ke dalam fungsi aktivasi sigmoid (**Persamaan (2)**) atau **Persamaan (3)**) atau ke dalam fungsi identitas (**Persamaan (4)**) untuk mendapatkan nilai *neuron* Z_1 'f(net)'. Nilai 'f(net)' suatu *neuron* menandakan seberapa besar pengaruh *neuron* tersebut terhadap *neuron* di atasnya.

2.2 Pemrograman jaringan syaraf tiruan dengan MATLAB 2013a

Pembuatan JST dilakukan dalam aplikasi MATLAB 2013a sehingga ada beberapa ketentuan yang harus dipatuhi agar JST dapat terbentuk, misalnya format *input* dan *output*. Sisi positifnya adalah, algoritma fungsi pelatihan, aktivasi, dan performa tidak perlu dibuat menalinkan hanya perlu menggunakan fungsi bawaan dari aplikasi MATLAB 2013a. Berikut ini adalah tahapan pembentukan JST tersebut.

1. Memasukkan *input* dan target pelatihan.

Proses ini dilakukan dengan meng-assign suatu variabel dan menetapkan matriks *input* ke dalam variabel *input* dan matriks target ke dalam variabel *output* tertentu. Misalnya, jika variabel yang menyimpan matriks *input* pelatihan adalah 'inp_p' dan yang menyimpan matriks target pelatihan adalah 'tar_p', algoritmenya adalah sebagai berikut.

```
inp_p = [matriks_input_pelatihan];
tar_p = [matriks_target_pelatihan];
```

2. Memasukkan *input* dan target pengujian.

Proses ini sama dengan seperti tahap 1. Misalnya, variabel yang menyimpan matriks *input* pengujian bernama 'inp_t' dan yang menyimpan matriks target pelatihan bernama 'tar_t'.

```
inp_t = [matriks_input_pengujian];
tar_t = [matriks_target_pengujian];
```

3. Inisialisasi jaringan syarat tiruan.

Ada dua perintah inisialisasi JST dalam penulisan ini, bergantung pada format *output* JST yang diinginkan. Pembentukan JST dilakukan melalui salah satu dari dua algoritme berikut.

```
net = feedforwardnet([S1 S2 ... SN]);
```

atau dengan perintah berikut.

```
net = patternnet([S1 S2 ... SN]);
```

$S_1, S_2,$ sampai S_N masing-masing adalah jumlah *neuron* pada *hidden layer* 1, *hidden layer* 2, dan *hidden layer* ke-N. Misalnya, jika JST memiliki dua *hidden layer* dengan 12 *neuron* di *hidden layer* 1 dan 5 *neuron* di *hidden layer* 2, algoritmenya menjadi seperti berikut.

```
net = feedforwardnet([12 5]);
```

atau dengan perintah berikut.

```
net = patternnet([12 5])
```

4. Menetapkan parameter pelatihan.

Parameter pelatihan meliputi jumlah epoch maksimum (*epochs*) yang disimpan ke dalam variabel 'p_e', durasi maksimum (*time*), dan nilai eror minimum (*goal*). Penetapan nilai-nilai tersebut dilakukan melalui algoritme berikut.

```
net.trainParam.epochs = p_e;
```

```
net.trainParam.time = p_w;
```

```
net.trainParam.goal = p_g;
```

5. Memulai pelatihan JST.

Pelatihan dimulai dengan algoritme berikut.

```
[net,tr] = train(net,inp_p,tar_p);
```

6. Menghitung *output* JST hasil pelatihan.

Dengan memasukkan *input* pelatihan (inp_p) ke dalam JST yang telah dilatih, *output* pelatihan dapat diperoleh. Misalnya, jika *output* pelatihan disimpan ke dalam variabel 'y_p', algoritme perhitungan *output* tersebut menjadi seperti berikut.

```
y_p = net(inp_p);
```

7. Menghitung *squared-error* dan MSE antara target-*output* pelatihan.

Squared-error (SE) dihitung dengan mengkuadratkan selisih antara target dengan *output* pelatihan. Jika data pelatihan berjumlah 20, ada 20 SE yang harus dihitung. MSE dihitung dengan merata-ratakan seluruh SE tersebut. Jika SE disimpan ke dalam variabel 'se_p' dan MSE disimpan ke dalam variabel 'mse_p', algoritme perhitungan SE dan MSE adalah sebagai berikut.

```
se_p = sum((tar_p - y_p).^2);
```

```
mse_p = mean(se_p);
```

8. Memulai pengujian JST.

Proses ini sama dengan tahap ke-6 tetapi *output* pengujian disimpan ke dalam variabel 'y_t'. Pengujian dilakukan melalui algoritme berikut.

```
y_t = net(inp_t);
```

9. Menghitung SE dan MSE target-*output* pengujian.

Proses ini sama dengan tahap ke-7 namun target dan *output* yang digunakan adalah target pengujian yang

disimpan dalam variabel 'tar_t' dan *output* pengujian dalam 'y_t'. SE pengujian disimpan ke dalam variabel 'se_t' dan MSE disimpan ke dalam variabel 'mse_t'. Berikut ini adalah algoritmenya.

```
se_t = sum((tar_t - y_t).^2);
```

```
mse_t = mean(se_t);
```

2.3 Tanah

Dalam pengertian teknik secara umum, tanah didefinisikan sebagai material yang terdiri dari agregat (butiran) mineral-mineral padat yang tidak tersementasi (terikat secara kimia) satu sama lain dan dari bahan-bahan organik yang telah melapuk (yang berpartikel padat) disertai dengan zat cair dan gas yang mengisi ruang-ruang kosong di antara partikel-partikel padat tersebut. Ukuran partikel tanah sangat beragam dengan variasi yang cukup besar. Tanah umumnya dapat disebut sebagai kerikil (*gravel*), pasir (*sand*), lanau (*silt*), atau lempung (*clay*), tergantung pada ukuran partikel yang paling dominan pada tanah tersebut.

2.4 Kadar air alami

Tanah terdiri dari butiran padat dan rongga pori. Pada keadaan tidak jenuh air, rongga pori tersebut berisi udara dan air. Sedangkan dalam keadaan jenuh air, seluruh pori tersebut terisi air. Rasio antara berat air dan berat butiran padat suatu tanah disebut sebagai kadar air. Kadar air alami tanah, sesuai namanya, adalah kadar air saat tanah berada di lapangan atau dalam keadaan tidak terganggu (*undisturbed*).

2.5 Batas atterberg tanah

Atterberg (1911) membagi kedudukan atau kondisi fisik tanah lempung berdasarkan kadar airnya, yakni: padat, semi padat, plastis, dan cair. Kadar air yang menjadi batas antara keempat kondisi itu disebut dengan batas susut, batas plastis, dan batas cair. Selisih antara batas cair dan batas plastis adalah indeks plastisitas. Dalam penulisan ini, batas cair, batas plastis, dan indeks plastisitas yang digunakan dalam JST.

2.6 Gradasi butiran tanah

Pada dasarnya, butiran padatan tanah terdiri dari butiran kasar dan halus. Berikut ini adalah ketentuan jenis tanah berdasarkan ukuran butiran dengan mengacu pada *Unified Soil Classification System* (USCS).

1. Kerikil, bila diameter butiran sebesar 4,75 - 75 mm.
2. Pasir, bila diameter butiran sebesar 0,075 - 4,75 mm.
3. Butiran halus, bila diameter butiran lebih kecil dari 0,075 mm.

Dalam penulisan ini, butiran halus dipisahkan menjadi lanau dan lempung. Jenis tanah disebut lanau jika ukuran butiran 0,002 - 0,075 mm sedangkan disebut lempung jika ukuran butiran kurang dari 0,002 mm.

Gradasi butiran tanah terdiri dari empat (4) nilai dimana masing-masing nilai menunjukkan rasio berat kerikil terhadap berat seluruh tanah, rasio berat pasir terhadap berat seluruh tanah, rasio berat lanau terhadap berat seluruh tanah, dan rasio berat lempung terhadap berat seluruh tanah. Jika keempat rasio tersebut dijumlahkan, nilainya akan menjadi satu (1). Nilai gradasi butiran tanah biasanya berupa persentase namun dalam JST yang dibuat pada penulisan ini, gradasi butiran harus dalam bentuk desimal. Nilai gradasi butiran tersebut didapat dari analisis saringan dan uji hidrometer.

2.7 Berat volume tanah

Berat volume atau berat unit (*unit weight*) basah atau lembab adalah rasio antara berat tanah keseluruhan (termasuk berat air dan udara dalam tanah tersebut meskipun berat udara dapat diabaikan) dengan volume tanah keseluruhan (termasuk volume air dan udara di dalamnya). Satuan yang digunakan dalam JST ini adalah metrik ton per meter kubik (ton/m^3).

2.8 Berat spesifik tanah

Berat spesifik atau *specific gravity* atau berat jenis (G_s) adalah rasio antara berat volume butiran padat dengan volume air pada temperatur 4°C. Nilai ini tidak memiliki satuan.

2.9 Kelas tanah

Kelas tanah yang digunakan dalam penulisan ini didasarkan pada USCS. Simbol-simbol yang digunakan dalam USCS antara lain.

1. G = kerikil (*gravel*)
2. S = pasir (*sand*)
3. C = lempung (*clay*)
4. M = lanau (*silt*)
5. O = lanau atau lempung organik
6. PT = tanah gambut dan tanah organik tinggi (*peat*)
7. W = gradasi baik (*well-graded*)
8. P = gradasi buruk (*poorly-graded*)
9. H = plastisitas tinggi (*high-plasticity*)
10. L = plastisitas rendah (*low-plasticity*)

Kelas tanah yang ada pada USCS adalah GW, GP, GM, GC, SW, SP, SM, SC, ML, CL, MH, CH, OL, OH, dan PT. Tanah dikelompokkan ke dalam salah satu dari ke-15 kelas tersebut berdasarkan hasil analisis saringan dan hidrometer.

Sebelum digunakan dalam program JST, kelas tanah harus diubah menjadi suatu vektor yang semua komponennya bernilai nol (0) kecuali satu komponen yang bernilai satu (1). Komponen yang bernilai satu tersebut merupakan komponen yang terasosiasi dengan kelas tanah. Berikut ini adalah ketentuan merubah kelas tanah menjadi vektor kelas tanah.

1. Menetapkan urutan komponen kelas, dimana dalam

penulisan ini, yaitu: GW, GP, GM, GC, SW, SP, SM, SC, ML, CL, MH, CH, OL, OH, dan PT.

2. Berdasarkan urutan di atas, tulis angka satu (1) pada komponen yang merupakan kelas tanah lalu tulis angka nol (0) pada komponen-komponen lainnya.

Contohnya, jika kelas tanah adalah GW, komponen pertama diisi nilai satu (1) sedangkan komponen lainnya diisi nilai nol (0) sehingga vektor kelas GW adalah $[1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$. Jika kelas tanah adalah CH, komponen ke-12 diisi nilai satu (1) sedangkan komponen lainnya diisi nilai nol (0) sehingga vektor kelas CH adalah $[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]$.

2.10 N-SPT tanah

Uji SPT terdiri atas uji pemukulan tabung belah dinding tebal ke dalam tanah, disertai dengan pengukuran jumlah pukulan yang dibutuhkan untuk memasukkan tabung belah sedalam 300 mm secara vertikal. Palu (beban jatuh atau biasa disebut hammer) dengan berat 63,5 kg dijatuhkan secara bebas (*free-fall*) dan berulang dari ketinggian 0,76 m.

Pelaksanaan pengujian dibagi dalam tiga tahap, yaitu berturut-turut setebal 150 mm untuk masing-masing tahap. Tahap pertama dicatat sebagai dudukan, sementara jumlah pukulan untuk memasukkan tahap ke-dua dan ke-tiga dijumlahkan untuk memperoleh nilai pukulan N atau perlawanan SPT. Dengan kata lain, nilai N dari SPT adalah jumlah pukulan yang dibutuhkan untuk menembus suatu lapisan tanah sedalam 300 mm.

Nilai N-SPT berada pada interval bilangan bulat positif dari satu hingga 50 sehingga *output* JST prediksi N-SPT perlu dibatasi agar mengeluarkan *output* yang berada dalam interval tersebut. Oleh karena itu, target JST N-SPT harus dirubah menjadi vektor dengan cara yang sama seperti mengubah kelas tanah menjadi vektor kelas tanah. Cara merubah nilai N-SPT menjadi vektor kelas N-SPT adalah sebagai berikut.

1. Menetapkan urutan komponen kelas, dimana dalam penulisan ini, yaitu: 1, 2, 3, ... , 48, 49, 50.
2. Berdasarkan urutan di atas, tulis angka satu (1) pada komponen yang merupakan nilai N-SPT lalu tulis angka nol (0) pada komponen-komponen lainnya.

[illegible]

2.11 Kohesi tak terdrainase tanah

Kohesi adalah ketertarikan antarpartikel yang sama sedangkan 'tak terdrainase' merupakan istilah yang menunjukkan bahwa kadar air tanah tidak berubah atau air tanah tidak dalam keadaan mengalir. Kohesi tak

terdrainase (C_u) adalah suatu koefisien yang menunjukkan tegangan tarik antar partikel tanah saat air tanah dalam keadaan tidak mengalir (*undrained*). Nilai ini dapat diperoleh dari uji triaksial.

Dalam JST, nilai ini bisa langsung digunakan sebagai target tanpa perlu mengalami perubahan seperti pada kelas tanah dan N-SPT. Berbeda dari N-SPT yang bersifat diskrit, nilai C_U bersifat kontinu pada interval bilangan positif. Satuan yang digunakan untuk C_U pada penulisan ini adalah kg/cm^2 .

3. Hasil Jaringan Syaraf Tiruan

3.1 Jaringan syaraf tiruan klasifikasi kelas tanah

Input pelatihan yang digunakan untuk jaringan ini adalah batas cair/ *liquid limit* (LL), batas plastis/ *plastic limit* (PL), indeks plastisitas/ *plasticity index* (PI), fraksi kerikil, fraksi pasir, fraksi lanau, dan fraksi lempung. Semua data tersebut dalam bentuk desimal.

Data pelatihan yang digunakan untuk pelatihan JST klasifikasi tanah ini berjumlah 57 sedangkan data pengujiannya berjumlah 10. Jumlah *neuron* pada *hidden layer* 1 adalah 17 sedangkan jumlah *neuron* pada *hidden layer* 2 adalah 3. Berikut ini adalah detail mengenai parameter pelatihan JST ini.

Fungsi aktivasi = ‘tansig’

Fungsi pelatihan = 'trainscg'

Fungsi performa = 'mse'

Epoch maksimum = 1500

Waktu maksimum = inf

Rangkuman pelatihan JST ini ditunjukkan pada **Tabel 1**. Setelah dilatih, proses selanjutnya adalah pengujian. Pengujian ini berfungsi untuk menilai seberapa baik JST memprediksi data yang selain dari data pelatihannya. Rangkuman pengujian ditunjukkan pada **Tabel 2**.

Nilai maksimum pada vektor *output* data ke-7 hanya sebesar 0,4 pada komponen ML dan 0,4 pada komponen MH. Kedua nilai tersebut tidak mendekati satu (1) sehingga JST dianggap tidak mampu mengenali data pengujian ke-7. Oleh sebab itu, bagian *output* data pengujian ke-7 pada **Tabel 2** dikosongkan.

3.2 Jaringan syaraf tiruan prediksi N-SPT

Sama seperti JST klasifikasi kelas tanah, *input* pelatihan yang digunakan untuk jaringan ini adalah batas cair/ *liquid limit* (LL), batas plastis/ *plastic limit* (PL), indeks plastisitas/ *plasticity index* (PI), fraksi kerikil, fraksi pasir, fraksi lanau, dan fraksi lempung. Semua data tersebut dalam bentuk desimal.

Data pelatihan yang digunakan untuk pelatihan JST ini berjumlah 37 sedangkan data pengujiannya berjumlah 10. JST ini didesain memiliki 7 neuron di *input layer*, 17 neuron di *hidden layer 1*, 15 neuron di *hidden layer 2*, 15 neuron di *hidden layer 3*, 15 neuron di *hidden layer 4*, 14 neuron di *hidden layer 5*, dan 50 neuron di *output layer*. Berikut ini adalah detail mengenai

Tabel 1. Rangkuman pelatihan JST klasifikasi tanah

Data Ke-	Target	Output	Squared-Error
1	MH	MH	0.0000
2	CH	CH	0.0000
3	CH	CH	0.0000
4	MH	MH	0.0000
5	CH	CH	0.0000
6	CH	MH	1.9978
7	CH	CH	0.0000
8	MH	MH	0.0000
9	MH	MH	0.0000
10	MH	MH	0.0000
11	MH	MH	0.0000
12	ML	ML	0.0000
13	CH	CH	0.0000
14	MH	MH	0.0000
15	MH	MH	0.0000
16	CH	CH	0.0000
17	MH	MH	0.0001
18	CH	CH	0.0027
19	CH	CH	0.0000
20	CH	CH	0.0000
21	CH	CH	0.0000
22	CH	CH	0.0000
23	CH	CH	0.0003
24	MH	MH	0.0000
25	ML	ML	0.0000
26	ML	ML	0.0000
27	CH	CH	0.0000
28	CH	CH	0.0000
29	MH	MH	0.0001
30	CH	CH	0.0000
31	CH	CH	0.0000
32	MH	MH	0.0000
33	CH	CH	0.0000
34	CH	CH	0.0000
35	CH	CH	0.0000
36	MH	MH	0.0000
37	MH	MH	0.0000
38	MH	MH	0.0000
39	MH	MH	0.0000
40	MH	MH	0.0000
41	MH	MH	0.0000
42	MH	MH	0.0001
43	MH	MH	0.0000
44	MH	MH	0.0000
45	MH	MH	0.0000
46	MH	MH	0.0000
47	MH	MH	0.0000
48	MH	MH	0.0000
49	MH	MH	0.0000
50	MH	MH	0.0000
51	MH	MH	0.0000
52	MH	MH	0.0000
53	MH	MH	0.0000
54	MH	MH	0.0000
55	MH	MH	0.0000
56	MH	MH	0.0000
57	MH	MH	0.0000
Rata-Rata			0.0351

Tabel 2. Rangkuman pengujian JST klasifikasi tanah

Data Ke-	Target	Output	Squared-Error
1	CH	CH	0.0000
2	CH	MH	1.9970
3	MH	MH	0.0000
4	MH	MH	0.1683
5	ML	MH	1.8248
6	CH	CH	0.0000
7	MH	-	0.5440
8	ML	ML	0.0000
9	CH	MH	1.9998
10	MH	MH	0.0000
Rata-Rata			0.6534

parameter pelatihan JST ini.

1. Fungsi aktivasi = 'tansig'
2. Fungsi pelatihan = 'trainscg'
3. Fungsi performa = 'mse'
4. Epoch maksimum = 1500
5. Waktu maksimum = inf

Rangkuman pelatihan JST ini ditunjukkan pada **Tabel 3** sedangkan rangkuman pengujian ditunjukkan pada **Tabel 4**.

3.3 Jaringan syaraf tiruan prediksi kohesi tak terdrainase

Input JST ini berjumlah 25, antara lain kadar air alami (w_n), *specific gravity* (G_s), berat jenis, batas cair, batas plastis, indeks plastisitas, fraksi kerikil, fraksi pasir, fraksi lanau, fraksi lempung, dan vektor kelas dengan 15 komponen. Kadar air alami, batas cair, batas plastis, indeks plastisitas, fraksi kerikil, fraksi pasir, fraksi lanau, dan fraksi lempung dalam bentuk desimal (bukan dalam persentase). Berat jenis dalam satuan metrik ton per meter kubik (ton/m^3). *Specific gravity* tidak memiliki satuan. Vektor kelas 15 komponen memiliki format yang sama seperti vektor *output* JST klasifikasi kelas tanah, yaitu tiap komponen bernilai 0 kecuali satu komponen, yang terasosiasi dengan kelas tanah tersebut, yang bernilai 1.

Data pelatihan yang digunakan untuk pelatihan JST ini berjumlah 28 sedangkan data pengujian berjumlah 10. JST ini didesain memiliki 25 *neuron* di *input layer*, 5 *neuron* di *hidden layer* 1, dan 1 *neuron* di *output layer*. Berikut ini adalah detail mengenai parameter pelatihan JST ini.

1. Fungsi aktivasi selain *neuron output* = 'tansig'
2. Fungsi aktivasi *neuron output* = 'purelin'
3. Fungsi pelatihan = 'trainlm'
4. Fungsi performa = 'mse'
5. Epoch maksimum = 1000
6. Waktu maksimum = inf

Rangkuman pelatihan JST ini ditunjukkan pada **Tabel 5** sedangkan rangkuman pengujian ditunjukkan pada **Tabel 6**.

Tabel 3. Rangkuman pelatihan JST prediksi N-SPT

Data Ke-	Target	Output	Squared-Error
1	1	4	2.0026
2	1	1	0.0000
3	2	2	0.0000
4	2	2	0.0000
5	3	3	0.0000
6	3	3	0.0000
7	3	3	0.0336
8	4	4	0.0000
9	4	4	0.0000
10	4	4	0.0000
11	4	4	0.0000
12	5	3	1.7382
13	5	5	0.0000
14	5	5	0.0000
15	5	5	0.0000
16	6	6	0.0000
17	6	6	0.0219
18	6	6	0.0000
19	6	6	0.0000
20	7	7	0.0000
21	7	7	0.0261
22	7	4	1.9962
23	7	7	0.0000
24	7	7	0.0000
25	8	6	1.9901
26	8	6	1.0124
27	10	10	0.0000
28	10	10	0.0000
29	10	10	0.0000
30	11	11	0.0000
31	11	11	0.1022
32	13	14	1.5744
33	13	13	0.0001
34	14	10	1.1494
35	14	14	0.0000
36	16	7	1.9698
37	37	37	0.0000
Rata-Rata			0.3680

Tabel 4. Rangkuman pengujian JST prediksi N-SPT

Data Ke-	Target	Output	Squared-Error
1	2	3	1.3543
2	3	3	0.3097
3	4	3	1.9844
4	4	10	1.0701
5	5	13	1.9315
6	5	6	1.9925
7	6	6	0.7753
8	7	4	1.9998
9	10	10	0.5840
10	11	5	2.6950
Rata-Rata			1.4697

Tabel 5. Rangkuman pelatihan JST prediksi kohesi tak terdrainase

Data Ke-	Target	Output	Squared-Error
1	0.070	0.104	0.0012
2	0.100	0.064	0.0013
3	0.100	0.094	0.0000
4	0.130	0.115	0.0002
5	0.130	0.128	0.0000
6	0.150	0.154	0.0000
7	0.180	0.181	0.0000
8	0.190	0.198	0.0001
9	0.190	0.189	0.0000
10	0.220	0.221	0.0000
11	0.220	0.229	0.0001
12	0.220	0.196	0.0006
13	0.220	0.216	0.0000
14	0.230	0.277	0.0022
15	0.270	0.270	0.0000
16	0.270	0.220	0.0025
17	0.310	0.306	0.0000
18	0.320	0.321	0.0000
19	0.320	0.239	0.0066
20	0.360	0.363	0.0000
21	0.380	0.380	0.0000
22	0.420	0.418	0.0000
23	0.440	0.443	0.0000
24	0.460	0.463	0.0000
25	0.460	0.091	0.1365
26	0.500	0.501	0.0000
27	0.510	0.390	0.0143
28	0.750	0.749	0.0000
Rata-Rata			0.0059

Tabel 6. Rangkuman pengujian JST prediksi kohesi tak terdrainase

Data Ke-	Target	Output	Squared-Error
1	0.090	0.049	0.0017
2	0.120	0.279	0.0251
3	0.140	0.076	0.0041
4	0.200	0.028	0.0296
5	0.260	0.038	0.0492
6	0.300	0.402	0.0105
7	0.340	0.406	0.0044
8	0.400	0.121	0.0778
9	0.480	0.466	0.0002
10	0.670	0.520	0.0226
Rata-Rata			0.0225

4. Diskusi Hasil

Tiga jaringan syaraf tiruan yang dibuat dalam penulisan ini memiliki kinerja yang berbeda dan perlu dianalisis tidak hanya berdasarkan nilai MSE namun juga berdasarkan jumlah data pelatihan yang sukses diprediksi dan berdasarkan kecocokan antara parameter pelatihan dengan format *output* yang diinginkan.

Tabel 6. Rangkuman pengujian JST prediksi kohesi tak terdrainase

Data Ke-	Target	Output	Squared-Error
1	0.090	0.049	0.0017
2	0.120	0.279	0.0251
3	0.140	0.076	0.0041
4	0.200	0.028	0.0296
5	0.260	0.038	0.0492
6	0.300	0.402	0.0105
7	0.340	0.406	0.0044
8	0.400	0.121	0.0778
9	0.480	0.466	0.0002
10	0.670	0.520	0.0226
Rata-Rata			0.0225

4.1 Analisis jaringan syaraf tiruan klasifikasi tanah

MSE JST ini adalah 0,0351 saat pelatihan dan 0,6534 saat pengujian. Berdasarkan MSE, JST ini mampu mengenali data pelatihan karena MSE pelatihan kurang dari 0,05 namun JST ini tidak mampu mengenali data pengujian karena MSE pengujian lebih dari 0,05.

Data pelatihan yang berhasil dikenali berjumlah 56 dari total 57 data atau 98,2% data pelatihan. Data pengujian yang berhasil dikenali berjumlah 6 dari 10 atau 60% data pengujian. Besarnya data yang dapat dikenali menunjukkan hubungan yang erat antara *input* dengan target. Kecocokan fungsi pelatihan ‘*trainscg*’ juga sangat mempengaruhi performa JST ini.

Kurangnya variasi data pelatihan, terutama tanah organik menyebabkan JST ini tidak bisa digunakan untuk memprediksi tanah organik. Ketiadaan data tanah organik dalam data pelatihan menyebabkan JST ini tidak mampu mengenali jenis tanah tersebut. Oleh sebab itu, seandainya JST ini mengeluarkan *output* tanah organik, pengguna harus mengabaikan hasil tersebut.

4.2 Analisis jaringan syaraf tiruan prediksi N-SPT

MSE JST ini adalah 0,368 saat pelatihan dan 1,4697 saat pengujian. Berdasarkan MSE, JST ini tidak mampu mengenali data pelatihan maupun pengujian karena MSE keduanya lebih dari 0,005.

Jika dilihat dari nilai SE yang lebih kecil dari 0,05, 28 dari 37 atau 75,7% data pelatihan dapat diprediksi oleh JST ini sedangkan 0 dari 10 atau 0% data pengujian yang berhasil dikenali oleh JST ini.

Perbedaan nilai N-SPT sebesar satu (1) masih dapat ditoleransi untuk beberapa perhitungan, seperti daya dukung tiang maupun penurunan, sehingga *output* pelatihan dan pengujian yang memiliki selisih mutlak (*Absolute Error*) sebesar satu (1) masih dapat dikategorikan ‘sukses dikenali’. Dengan alasan tersebut, 29 dari 37 atau 78,38% data pelatihan dapat dikenali dan 6 dari 10 atau 60% data pengujian dapat dikenali.

Besarnya MSE JST ini dapat disebabkan oleh ketidakcocokan fungsi pelatihan dan format *output* yang digunakan dalam JST ini. Seperti yang telah dijelaskan sebelumnya, nilai N-SPT yang digunakan dan yang dikeluarkan jaringan ini berbentuk vektor. SE dihitung dengan menjumlahkan kuadrat selisih tiap komponen antara target dengan *output* jaringan. Sebagai akibatnya, misalnya jaringan mengeluarkan *output* N-SPT sebesar 10 sedangkan targetnya adalah 11, SE-nya akan menjadi jauh lebih besar dari 0,05, biasanya lebih dari 1. Hal demikian terjadi pada data pelatihan ke-32 serta data pengujian ke-1, 3, 6, 7, dan 9.

Pada data pelatihan ke-32, data pengujian ke-1, 3, dan 6, meskipun setelah dikonversi ke dalam skalar, selisih antara target dengan *output* adalah 1 namun dalam bentuk vektor, SE-nya sangat jauh dari 0,05. Pada data pengujian ke-7, ada komponen vektor yang seharusnya nilainya mendekati 0 tetapi justru bernilai 0,87 sehingga memperbesar SE. Sedangkan pada data pengujian ke-9, komponen vektor terbesar memang berada pada posisi N-SPT = 10 namun nilai komponen tersebut masih jauh lebih kecil dari satu, yaitu 0,2501; penyebab lainnya adalah adanya komponen lain yang seharusnya bernilai 0 tetapi justru bernilai 0,15. Sebagai akibatnya, MSE jaringan saat pengujian menjadi jauh di atas 0,05, yaitu 1,4697, meskipun setelah dikonversi ke bentuk skalar, ada 6 data yang hanya berselisih satu dengan targetnya.

Interval N-SPT sebenarnya adalah bilangan bulat positif dari satu (1) hingga lima puluh (50) sehingga, mungkin kombinasi JST ‘*feedforwardnet*’ dengan fungsi pelatihan ‘*trainlm*’ lebih cocok dibandingkan dengan ‘*patternnet*’ dengan ‘*trainscg*’. Masalah pada penggunaan ‘*feedforwardnet*’ adalah *output*-nya yang sering berada di luar interval N-SPT. Itu sebabnya penulis memilih menggunakan ‘*patternnet*’ agar *output*-nya selalu berada di dalam interval tersebut meskipun konsumsi memorinya menjadi lebih besar karena *neuron output* JST memiliki lima puluh (50) komponen.

4.3 Analisis jaringan syaraf tiruan prediksi kohesi tak terdrainase

JST ini mampu mengenali 27 dari 28 atau 96,43% data pelatihan dimana MSE keseluruhan pada sesi pelatihan adalah 0,0059. Pada sesi pengujian, JST mampu mengenali 9 dari 10 atau 90% data pengujian dengan MSE sebesar 0,0225. Jika dilihat dari MSE dan jumlah data yang berhasil dikenali, ini adalah JST yang memiliki performa paling baik dari tiga JST yang dibuat dalam penelitian ini.

Dengan melihat nilai MSE pelatihan dan pengujian JST ini, keputusan menggunakan JST ‘*feedforwardnet*’ dengan fungsi pelatihan ‘*trainlm*’ adalah tepat. Alasan pemilihan kombinasi tersebut juga disebabkan oleh format *output* JST ini yang merupakan skalar dan berada pada interval bilangan real positif yang bersifat kontinyu (tidak seperti N-SPT yang berada pada interval bilangan bulat positif 1 – 50).

5. Kesimpulan

Berdasarkan hasil perancangan dan analisis jaringan syaraf tiruan dalam penulisan ini, penulis dapat menyimpulkan hal-hal berikut ini.

1. Jaringan syaraf tiruan (JST) untuk prediksi kelas tanah memiliki nilai MSE sebesar 0,0351 saat pelatihan dan 0,6534 saat pengujian. Dengan demikian JST dapat dilatih untuk mengenali kelas tanah dengan arsitektur ini. Arsitektur jaringan ini adalah 7 neuron di *input layer*, 17 neuron di *hidden layer 1*, 3 neuron di *hidden layer 2*, dan 15 neuron di *output layer*.
2. Jaringan syaraf tiruan (JST) untuk prediksi N-SPT memiliki nilai MSE sebesar 0,3680 saat pelatihan dan 1,4697 saat pengujian. Dengan demikian, JST dengan desain seperti ini tidak mampu memenuhi persyaratan MSE kurang dari 0,05. Arsitektur jaringan ini adalah 7 neuron di *input layer*, 17 neuron di *hidden layer 1*, 15 neuron di *hidden layer 2*, dan 15 neuron di *hidden layer 3*, 15 neuron di *hidden layer 4*, 14 neuron di *hidden layer 5*, dan 50 neuron di *output layer*.
3. Jaringan syaraf tiruan (JST) untuk prediksi kohesi tak terdrainase (C_u) memiliki nilai MSE sebesar 0,0059 saat pelatihan dan 0,0225 saat pengujian. Sehingga JST ini dapat dilatih untuk mengenali kohesi tak terdrainase (*undrained cohesion* atau C_u). Arsitektur jaringan ini adalah 25 neuron di *input layer*, 5 neuron di *hidden layer 1*, dan 1 neuron di *output layer*.

Daftar Pustaka

- Anonim, 2012, <https://statistikaontheweb.wordpress.com/2012/10/21/kapan-0-05-kapan-0-01/>. (Tanggal akses: 17 September 2017).
- Anonim, 2016, <http://www.ngekul.com/rumus-mencari-spt-pada-borring-test/>. (Diakses: 17 September 2017)
- Away, G.A., 2014, *The Shortcut of MATLAB Programming*, Informatika Bandung: Bandung.
- Das, B.M., 1990, *Principles of Foundation Engineering*, PWS-KENT Publishing Company: Boston.
- Das, B.M., 1993, *Mekanika Tanah (Prinsip-prinsip Rekayasa Geoteknis)*. Erlangga: Jakarta.
- Hardiyatmo, H.C., 2011, *Analisis dan Perancangan Fondasi II*, Gadjah Mada University Press: Yogyakarta.
- Nafisah, S., *Pengklasifikasian Jenis Tanah Menggunakan Jaringan Syaraf Tiruan dengan Algoritma Backpropagation*, Universitas Gunadarma: Jakarta.
- Poulos, H.G., & Davis, E.H., 1980, *Pile Foundation Analysis and Design*, John Wiley & Sons: New York.
- Rahim, A., 2016, <http://tambangunp.blogspot.co.id/2016/04/apa-itu-kohesi-dan-sudut-geser-dalam.html>. (Diakses: 17 September 2017).
- Ray, A.M. dkk, 2012, *Mekanika Batuan*, Institut Teknologi Bandung: Bandung.
- Siang, J.J., 2005, *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*, ANDI: Yogyakarta.
- Tjolleng, A., 2017, *Pengantar Pemrograman MATLAB*, PT Elex Media Komputindo: Jakarta.

