

Sanggeni Galih Wardhana

## PREDIKSI KECEPATAN GELOMBANG GESER (VS) MENGGUNAKAN ...

## Sources Overview

12%

OVERALL SIMILARITY

1	journal.sekawan-org.id INTERNET	1%
2	Universitas Riau on 2021-10-14 SUBMITTED WORKS	<1%
3	repository.ub.ac.id INTERNET	<1%
4	repositori.uin-alauddin.ac.id INTERNET	<1%
5	Universitas Islam Lamongan on 2021-06-10 SUBMITTED WORKS	<1%
6	yunusmuhammad007.medium.com INTERNET	<1%
7	Pangeran Brandon Imran, Erwin Fernanda, Sthevanie Dhita Sudrazat. "PENGOLAHAN DATA LANDSAT DAN GRAVITASI SEBAGAI INDIKASI PANAS..." CROSSREF	<1%
8	UIN Syarif Hidayatullah Jakarta on 2020-02-26 SUBMITTED WORKS	<1%
9	ejournal.undip.ac.id INTERNET	<1%
10	serval.unil.ch INTERNET	<1%
11	widuri.raharja.info INTERNET	<1%
12	repository.uinib.ac.id INTERNET	<1%
13	www.sisfotenika.stmikpontianak.ac.id INTERNET	<1%
14	Ashpublications.Org INTERNET	<1%
15	repository.upi.edu INTERNET	<1%
16	www.mdpi.com INTERNET	<1%
17	jge.eng.unila.ac.id INTERNET	<1%
18	text-id.123dok.com INTERNET	<1%

19	jtsiskom.undip.ac.id	INTERNET	<1%
20	zombiedoc.com	INTERNET	<1%
21	Universitas Pelita Harapan	SUBMITTED WORKS	<1%
22	Universiti Teknologi Petronas on 2014-06-10	SUBMITTED WORKS	<1%
23	core.ac.uk	INTERNET	<1%
24	docplayer.info	INTERNET	<1%
25	Fakultas Ekonomi Universitas Indonesia on 2019-03-29	SUBMITTED WORKS	<1%
26	Universitas Brawijaya on 2016-12-23	SUBMITTED WORKS	<1%
27	www.zora.uzh.ch	INTERNET	<1%

**Excluded search repositories:**

None

**Excluded from document:**

Bibliography

Citations

**Excluded sources:**

None

# PREDIKSI KECEPATAN GELOMBANG GESER (VS) MENGUNAKAN *MACHINE LEARNING* DI SUMUR X

## *PREDICTION OF SHEAR WAVE VELOCITY (VS) USING MACHINE LEARNING IN WELL X*

Sanggeni Galih Wardhana<sup>1\*</sup>, M Aldi<sup>2</sup>, Indra Rivaldi Siregar<sup>3</sup>

<sup>1,2,3</sup>Teknik Geofisika, Fakultas Teknologi Eksplorasi dan Produksi, Universitas Pertamina

Received: 2021, November 29<sup>th</sup>

Accepted: 2022, February 18<sup>th</sup>

### Keywords:

*Machine learning*,

R<sup>2</sup> value score;

Shear wave velocity;

Well X and Y.

### Correspondent Email:

[sanggeniwardhana@gmail.com](mailto:sanggeniwardhana@gmail.com)

### How to cite this article:

Wardhana, S.G., Aldi, M., & Siregar, I.R. (2022). Prediksi Kecepatan Gelombang Geser (Vs) Menggunakan *Machine Learning* di Sumur X. *Jurnal Geofisika Eksplorasi*, 08(01), 67-77.

**Abstrak.** Salah satu parameter yang sangat penting dalam produksi dan eksplorasi minyak dan gas untuk mengetahui properti fisik batuan adalah kecepatan gelombang geser (Vs). Akan tetapi, dalam tahap akuisisi data ini tidak selalu tersedia karena keterbatasan biaya. Salah satu metode alternatif yang dapat dilakukan untuk mengestimasi kecepatan gelombang geser adalah memanfaatkan pembelajaran mesin (*machine learning*). *Machine learning* memiliki komputasi yang cepat setelah sekumpulan data dicoba untuk memahami pola dan faktor yang dapat mempengaruhi nilai parameter yang diprediksi, contoh kasus adalah kecepatan gelombang geser. Penelitian ini menggunakan data dari sumur X dan Y yang dapat diakses secara bebas di *website* SEG Wiki. Algoritma *machine learning* yang digunakan yaitu KNN, ANN, dan SVR. Beberapa parameter yang digunakan dalam mengestimasi nilai Vs berdasarkan nilai korelasi yang paling tinggi yaitu *Depth*, DTCO, NPHI, ECGR, dan ATRT, serta DTSM. Proses pembangunan model *machine learning* diperoleh dengan membagi data menjadi 70% sebagai data latih dan 30% sebagai data tes. Hasil prediksi kecepatan gelombang geser menunjukkan algoritma KNN lebih baik daripada ANN dan SVR dengan perolehan nilai *R<sup>2</sup> score* KNN = 0.9686, ANN = 0.9643, dan SVR = 0.9697 untuk sumur X dan *R<sup>2</sup> score* KNN = 0.6170, ANN = 0.4680, dan SVR = 0.5800 untuk sumur Y. Hasil prediksi pada sumur Y menunjukkan indikasi terjadinya *overfitting* akibat model terlalu mempelajari data percobaan.

**Abstract.** One of the essential parameters in oil and gas production and exploration to determine the physical properties of rocks is the shear wave velocity (Vs). However, this data is not always available in the acquisition stage due to limited costs. One alternative using a method to estimate the shear wave velocity is machine learning. Machine learning has fast computing after a trained data set to understand the patterns and factors that can affect the value of the predicted parameter, for example is shear wave velocity. This study uses wells X and Y data, which can be accessed freely on the SEG Wiki website. The Algorithms machine learning used

---

are KNN, ANN, and SVR. Several parameters used to estimate the value of  $V_s$  based on the highest correlation value are Depth, DTCO, NPFI, ECGR, ATRT, and DTSM. The building process of the machine learning model is obtained by dividing the data into 70% as training data and 30% as test data. The results of shear wave velocity prediction show that the KNN algorithm is better than ANN and SVR with the  $R^2$  score of KNN = 0.9686, ANN = 0.9643, and SVR = 0.9697 for wells X and  $R^2$  score of KNN = 0.6170, ANN = 0.4680, and SVR = 0.5800 for well Y. Prediction results in well Y indicate overfitting because the models too much studied the training data.

---

## 1. PENDAHULUAN

### 1.1. Latar Belakang

Kecepatan gelombang geser ( $V_s$ ) merupakan parameter yang sangat penting dalam dunia geofisika baik dalam studi pengembangan produksi maupun eksplorasi minyak dan gas untuk mengetahui properti fisik batuan seperti porositas, litologi, dan parameter geomekanik formasi seperti modulus geser. Namun, data terkait kecepatan gelombang geser tersebut tidak tersedia untuk semua sumur, terutama pada sumur tua karena memerlukan biaya yang tinggi (Akhundi dkk., 2014). Mengingat pentingnya data kecepatan gelombang geser untuk karakterisasi reservoir, maka sangat penting untuk memperkirakan parameter ini menggunakan data *logging* di sumur lainnya (Dumke & Berndt, 2019).

Banyak metode telah disajikan untuk memperkirakan parameter ini dari data *logging* sumur lainnya yang dicatat di sebagian besar sumur (Anemangely dkk., 2019). Pada penelitian ini metode yang akan digunakan untuk prediksi kecepatan gelombang geser adalah machine learning dengan algoritma KNN, ANN, dan SVR. Penerapan metode *machine learning* sudah dilakukan sebelumnya dalam industri minyak dan gas bumi, seperti yang dilakukan oleh Waldeland dkk. (2018) untuk interpretasi seismic, Isebor dan Grujic (2012) untuk mengoptimasi produksi minyak dalam menganalisis ketidakpastian geologi, serta Huang dkk. (2021) untuk prediksi *tight oil recovery*.

Pada penelitian ini proses komputasi dilakukan menggunakan bahasa pemrograman Python. Kami menganalisis terlebih dahulu beberapa data log yang sangat mempengaruhi kecepatan gelombang geser berdasarkan nilai korelasinya. Lalu, kami membangun model dari beberapa algoritma dan membandingkan model tersebut yang mampu secara optimal untuk memprediksi kecepatan gelombang geser.

## 2. TINJAUAN PUSTAKA

### 2.1. Prediksi Properti Data Well

Beberapa metode sudah pernah dilakukan untuk memprediksi sifat properti pada data *well* (Shi & Zhang, 2021). Salah satunya yang dilakukan oleh Hampson dkk. (2021) menggunakan metode *multi-attribute transforms*. Penggunaan metode tersebut membutuhkan serangkaian perhitungan sampel atribut untuk memperoleh transformasi *multi-attribute*. Kemudian dilakukan validasi silang untuk memperoleh keandalan dari *multi-attribute transforms* yang diturunkan. Metode tersebut membutuhkan proses yang lama dibandingkan dengan *machine learning* yang memiliki komputasi lebih cepat. Selain itu, pada metode *machine learning* cukup memerlukan data *well* tanpa data seismik. Hal tersebut dapat meminimalisir biaya dalam kegiatan eksplorasi minyak dan gas bumi.

### 2.2. K-Nearest Neighbor (KNN)

Menurut Mitchell (1997), KNN adalah algoritma jenis *supervised learning* yang digunakan untuk mengidentifikasi kemiripan

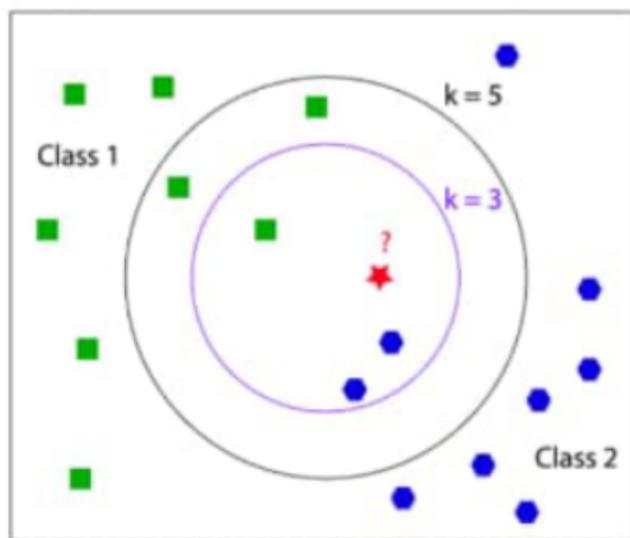
suatu titik baru berdasarkan jarak terdekat dari titik tetangganya (Gambar 1). KNN tidak membuat asumsi pada model atau generalisasi sehingga algoritma ini bebas untuk melakukan *training* pada segala jenis distribusi data. Teknik ini sangat sederhana dan mudah untuk diimplementasikan, karena hanya membutuhkan fungsi jarak untuk menghitung kemiripan Shmueli dkk., (2016).

Oleh karena itu, jarak merupakan kunci keberhasilan dari algoritma ini. Semakin jauh jarak berarti semakin tidak mirip, begitu juga sebaliknya. Beberapa metode untuk menghitung jarak antara lain Euclidean, Manhattan, Chebyshev, dan lain-lain. Akan tetapi, pada penelitian ini metode yang digunakan adalah Euclidean dengan persamaan seperti berikut.

$$d(a, b) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

Keterangan:

- $d(a, b)$  : jarak Euclidean
- $x_i$  : data 1
- $y_i$  : data 2
- $i$  : fitur ke-i
- $n$  : jumlah fitur



Gambar 1. Ilustrasi KNN (Mitchell, 1997).

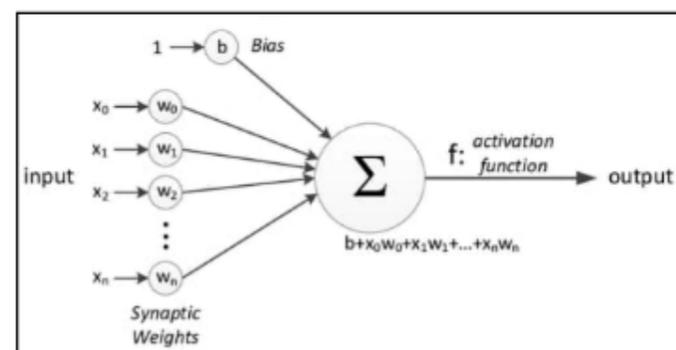
### 2.3. Artificial Neural Network (ANN)

ANN adalah set algoritma yang bekerja seperti jaringan saraf otak manusia, terdiri dari sejumlah elemen pemrosesan informasi (neuron) yang saling terhubung satu sama lain

untuk menyelesaikan suatu permasalahan (Bredkk., 2018; Wang et al., 2022).

Data dari *input layer* ( $x_0, x_1, x_2$ ) masuk ke dalam neuron, setiap input tersebut dikalikan terlebih dahulu dengan *weight* ( $w$ ), lalu seluruhnya dijumlahkan. Nilai  $w$  berperan sebagai faktor yang menentukan pengaruh dari setiap data input dan berubah saat proses belajar hingga model yang dihasilkan ANN mendekati target yang diinginkan.

Hasil penjumlahan sebelumnya dapat ditambahkan bias ( $b$ ) yang berfungsi untuk mengatur hasil output agar lebih akurat. Setelah itu, neuron akan dimasukkan ke sebuah fungsi yang disebut *activation function* ( $f$ ) (Gambar 2). Fungsi tersebut berperan dalam mengatur apakah neuron harus aktif atau tidak. Oleh karena itu, dalam algoritma ini penggunaan dari arsitektur jaringan, pembobotan, dan *activation function* harus dipilih secara optimal (Wang dkk., 2016).



Gambar 2. Ilustrasi model *single layer neural network* (Widiaputra, 2016).

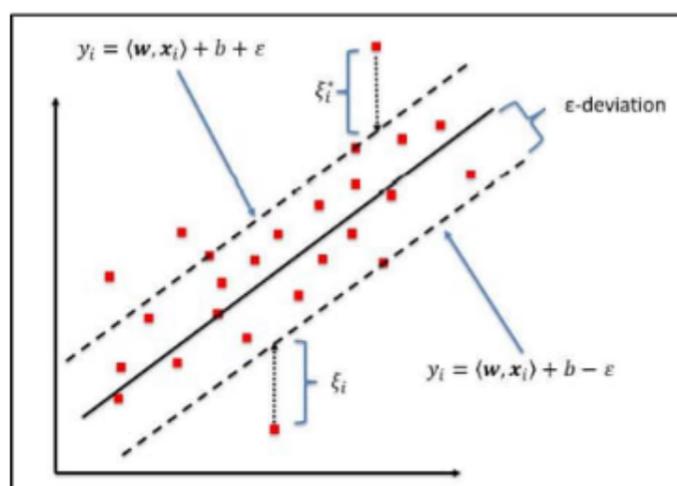
### 2.4. Support Vector Regressor (SVR)

Algoritma SVR merupakan algoritma untuk mengatasi kasus regresi yang dikembangkan dari metode *Support Vector Machine* (SVM) untuk menghasilkan bilangan riil dan kontinu (Ju & Hong, 2013). Menurut (Sholkopf & Smola, 2002), SVR bertujuan untuk menemukan sebuah fungsi  $f(x)$  sebagai suatu garis pemisah (*hyperplane*) yang sesuai dengan semua data input dengan mengusahakan nilai kesalahan sekecil mungkin (Gambar 3).

Keuntungan menggunakan SVR yaitu cocok untuk data set berdimensi tinggi atau kasus non linier menggunakan fungsi Kernel. SVR akan menghasilkan *trend* data yang bergelombang

mengikuti jalur data yang terbentuk sehingga prediksi data akan lebih akurat dibandingkan regresi linier yang hanya berupa garis lurus (Agustina dkk., 2018). Beberapa *hyperparameter* yang penting dalam algoritma SVR adalah sebagai berikut.

- a. *Constraint violation* (1) adalah parameter untuk menentukan seberapa besar deviasi dari parameter epsilon yang masih bisa ditoleransi.
- b. *Epsilon* ( $\epsilon$ ), berperan dalam mengontrol lebar zona regresi.
- c. *Gamma* ( $\gamma$ ), berhubungan dengan permasalahan *underfitting* dan *overfitting*.



Gambar 3. Ilustrasi SVR (Kleynhans dkk., 2017).

## 9. METODE PENELITIAN

### 3.1. Data Penelitian

Data yang digunakan pada penelitian ini terdiri dari data Sumur X dan Y di lapangan NW Shelf Australia yang dapat diakses secara bebas melalui website SEG Wiki. Kedua data tersebut sudah memiliki data kecepatan gelombang geser yang akan digunakan untuk pembuatan dan uji coba model *machine learning*.

### 12. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut.

- a. Prediksi kecepatan gelombang geser difokuskan pada Bahasa pemrograman Python.
- b. 20 Data yang digunakan pada penelitian ini adalah data sumur X dan Y.

- c. Algoritma *machine learning* 8 yang digunakan pada penelitian ini adalah KNN, ANN, dan SVR.

### 3.3. Diagram Alir Penelitian

Proses melakukan prediksi kecepatan gelombang geser ( $V_s$ ) menggunakan *machine learning* dimulai dengan cara menyiapkan dan memilih data input yang digunakan sebagai prediktor. 21 Data tersebut terdiri dari data *well log* dan data yang akan diprediksi yaitu kecepatan gelombang geser yang diperoleh pada data *well log* (Gambar 5) dan (Tabel 1). Data *well log* yang digunakan terdiri dari data prediktor yaitu Depth, DTCO (Delta-T Compressional), NPHI (Neutron Porosity Hydrogen Index), ECGR (EDTC Corrected Gamma-Ray), dan ATRT (*Array Induction Two Foot Rt*). Kemudian, data *output* yang akan diprediksi berupa DTSM (Delta-T Shear). Data parameter tersebut memiliki tingkat korelasi yang lebih tinggi dibandingkan data lainnya yaitu  $\geq 0.3$ . Tingkat korelasi tersebut adalah dasar yang diterapkan pada tahapan pemilihan fitur prediktor. Setelah prediktor terpilih, data akan dinormalisasi untuk menyamakan skala nilai sehingga tiap parameter memberikan kontribusi yang seimbang terhadap pembuatan model *machine learning*.

Tabel 1. Rank Log

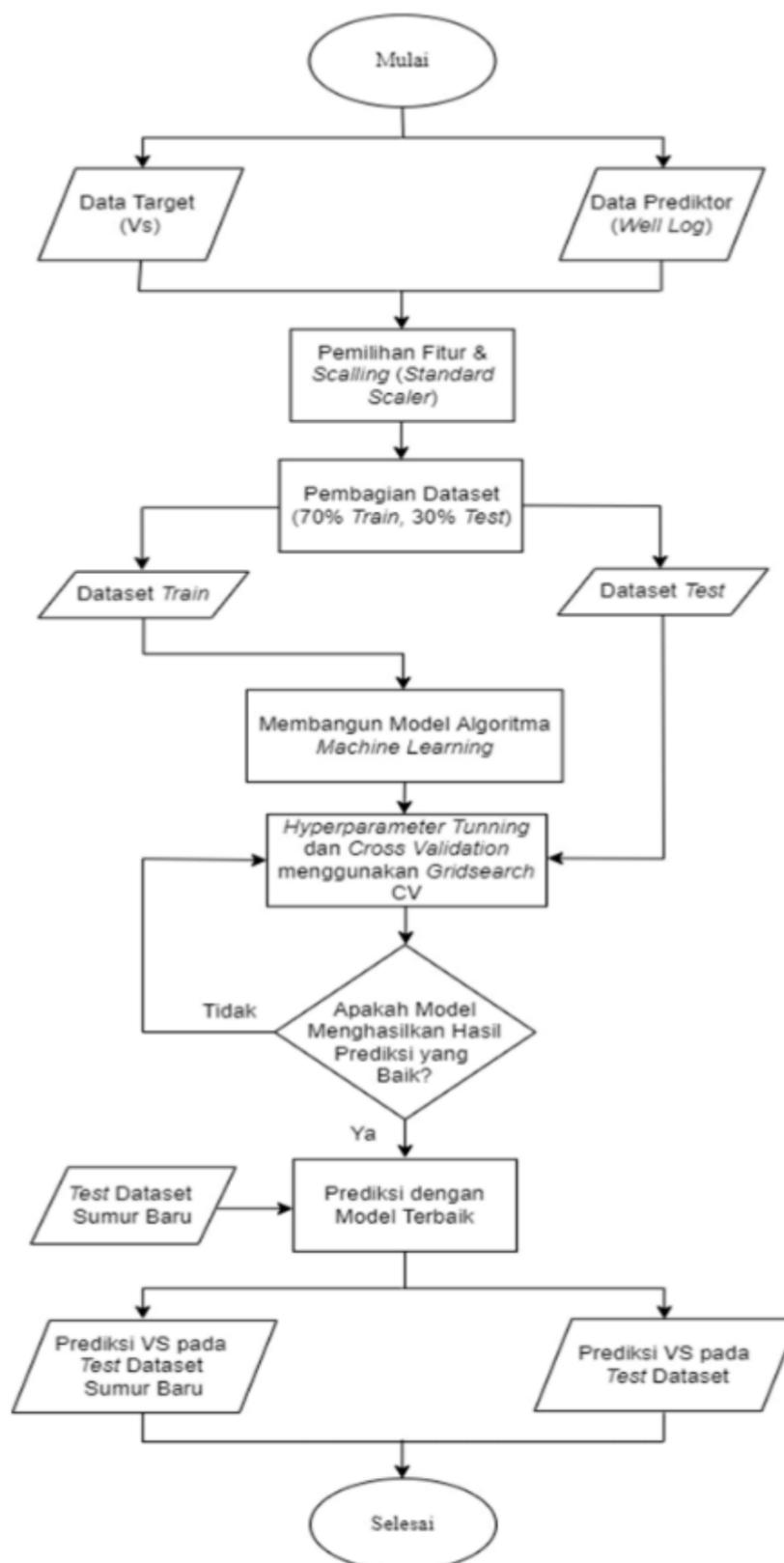
Rank	Nama Log	Korelasi dengan DTSM
1	DTCO	0.94
2	NPHI	0.85
3	DEPT	0.48
4	ECGR	0.46
5	ATRT	0.42

Dataset akan dibagi dengan persentase 70% sebagai dataset *training* dan 30% dataset *test*, hal tersebut bertujuan agar hasil prediksi cenderung mendekati *training data* apabila terjadi *overfitting* (Maulana dkk., 2019). Model dari *machine learning* akan dibangun menggunakan dataset *training* yang telah ditentukan. Pada penelitian ini, algoritma KNN dan SVR menggunakan *grid-search cross*

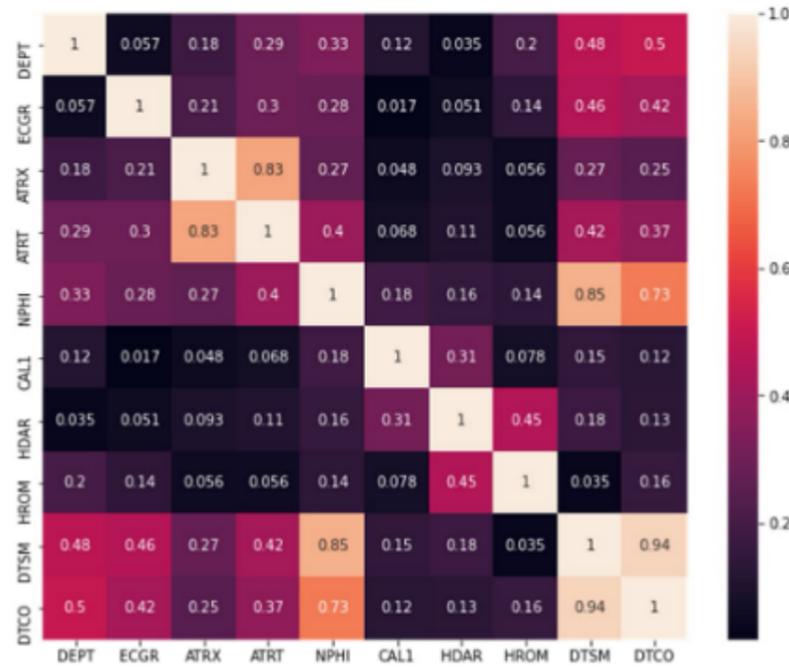
*validation* sebagai teknik untuk melakukan *hyperparameter tuning*. Teknik ini membantu dalam proses pemilihan nilai *hyperparameter* terbaik secara iteratif dalam suatu interval tertentu (Gambar 4). *Cross validation* dilakukan dengan membagi data validasi menjadi beberapa *fold*, kemudian akan dicari nilai rata-rata akurasi yang diperoleh dari hasil prediksi pada setiap *fold*. Pada algoritma ANN, proses *hyperparameter tuning* dan *cross validation* dilakukan secara eksperimental

karena menggunakan *library* yang berbeda, sehingga tidak dilakukan *grid search cross validation*.

Setelah dilakukan *cross validation*, model yang dihasilkan oleh *hyperparameter* dengan prediksi terbaik dapat ditentukan. Pada penelitian ini, prediksi pada sumur X digunakan untuk membangun model *machine learning* yang terbaik, lalu model tersebut diaplikasikan untuk memprediksi nilai kecepatan gelombang geser di sumur Y.



Gambar 4. Diagram alir penelitian.



Gambar 5. Hasil Korelasi Antar Parameter.

#### 4. HASIL DAN PEMBAHASAN

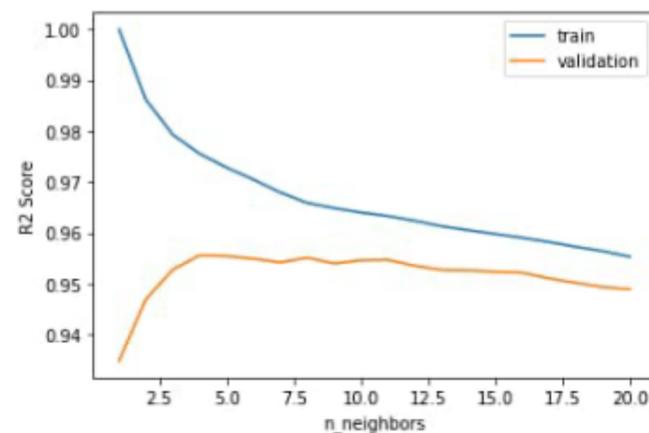
##### 4.1. Hasil Algoritma KNN

*Hyperparameter Tuning* pada algoritma KNN dengan memvariasikan nilai  $N$  (jumlah tetangga terdekat) yang akan digunakan untuk memprediksi nilai titik baru. Nilai  $N$  yang cocok akan menghasilkan skor *test* yang optimal. Skor *test* tertinggi adalah saat nilai  $N = 4$  dengan skor dataset *test* dan *train* masing-masing sebesar 0.9556 dan 0.9755 (Tabel 2). Kemudian, pada Gambar 6 dapat dilihat bahwa nilai  $N$  yang sangat kecil dapat menyebabkan model *overfitting* yang menunjukkan bahwa hasil prediksi dari dataset *training* mirip dengan dataset *training*, sedangkan nilai  $N$  yang terlalu besar akan menyebabkan model akan semakin *underfitting* yang ditunjukkan oleh semakin menurunnya skor *train* dan *test*.

Tabel 2. Performa Prediksi menggunakan KNN berdasarkan Variasi nilai  $n\_neighbors$  ( $N$ ).

$N$	Test ( $R^2$ Score)	Train ( $R^2$ Score)
1	0.9348	1.0000
2	0.9469	0.9862
3	0.9528	0.9793
4	0.9556	0.9755
5	0.9554	0.9729

6	0.9549	0.9705
7	0.9542	0.9679
8	0.9551	0.9659
9	0.9540	0.9649
10	0.9546	0.9640
11	0.9547	0.9633
12	0.9535	0.9624
13	0.9527	0.9613
14	0.9526	0.9605
15	0.9524	0.9598
16	0.9521	0.9590
17	0.9511	0.9582
18	0.9502	0.9573
19	0.9493	0.9564
20	0.9489	0.9553



Gambar 6. Grafik Performa Prediksi menggunakan KNN berdasarkan  $n\_neighbors$  ( $N$ )

#### 4.2. Hasil Algoritma ANN

Dalam menentukan model terbaik untuk prediksi gelombang geser, digunakan variasi kombinasi antara beberapa *hyperparameter* seperti *hidden layer*, *batch size*, *epoch*, dan *activation function*. Percobaan dengan kombinasi *hyperparameter* tersebut digunakan untuk *hyperparameter tuning* sehingga didapatkan model ANN terbaik. Parameter *hidden layer* pada ANN harus disesuaikan dengan data yang digunakan sehingga terhindar dari *underfitting* ataupun *overfitting*.

Berdasarkan Tabel 3, nilai *batch size* dan *epoch* sangat mempengaruhi hasil dari prediksi. Nilai *epoch* yang terlalu kecil dapat menyebabkan model semakin *overfitting* karena terlalu mirip dengan data *training*, sedangkan nilai *epoch* yang terlalu besar dapat menyebabkan model semakin *underfitting* yaitu akurasi skor pada data *test* yang semakin rendah daripada skor pada data *training*.

Pada prediksi gelombang geser dengan algoritma ANN, *activation function* adalah parameter yang sangat penting untuk disesuaikan, karena hanya terdapat beberapa fungsi yang cocok digunakan pada kasus regresi. Nilai *input* dan pembobotan pada setiap *input nodes* pada arsitektur ANN akan sangat mempengaruhi *activation function* dari neuron pada algoritma ANN. Pada penelitian ini digunakan *activation function* ReLu (*Rectified Linear Unit*) dan Linear.

Pada Tabel 3, model terbaik yang digunakan adalah pada saat skenario pertama dengan parameter *hidden layer* sebesar (7, 3, 1) dengan *batch size* dan *epoch* masing masing bernilai 10 dan 250, serta menggunakan *activation function* ReLu. Model tersebut menghasilkan skor pada data *train* sebesar 0.9687 dan nilai skor pada data *test* sebesar 0.9585.

Tabel 3. Performa Variasi Model Arsitektur dan *Hyperparameter* pada ANN.

Skenario	<i>Hyperparameter</i>			16 <i>Activation Function</i>	Akurasi ( $R^2$ Score)	
	<i>Hidden Layer Combination</i>	<i>Batch size</i>	<i>Epoch</i>		<i>Train</i>	<i>Test</i>
1	(7,3,1)	10	250	ReLu	0.9687	0.9585
2	(7,3,1)	15	250	ReLu	0.9569	0.9539
3	(7,3,1)	20	250	ReLu	0.9559	0.9503
4	(7,3,1)	10	300	ReLu	0.9654	0.9573
5	(7,3,1)	15	300	ReLu	0.9644	0.9584
6	(7,3,1)	20	300	ReLu	0.9504	0.9379
7	(10,5,1)	10	250	ReLu	0.9655	0.9583
8	(10,5,1)	15	250	ReLu	0.9636	0.9531
9	(10,5,1)	20	250	ReLu	0.9609	0.9564
10	(10,5,1)	10	300	ReLu	0.9671	0.9583
11	(10,5,1)	15	300	ReLu	0.9657	0.9552
12	(10,5,1)	20	300	ReLu	0.9592	0.9515
13	(7,3,1)	10	250	27 Linear	0.9536	0.9421
14	(7,3,1)	15	250	Linear	0.9537	0.9419
15	(7,3,1)	20	250	Linear	0.9538	0.9422
16	(7,3,1)	10	300	Linear	0.9538	0.9423
17	(7,3,1)	15	300	Linear	0.9537	0.9419
18	(7,3,1)	20	300	Linear	0.9537	0.9424
19	(10,5,1)	10	250	10 Linear	0.9536	0.943
20	(10,5,1)	15	250	Linear	0.9534	0.943
21	(10,5,1)	20	250	Linear	0.9538	0.9422
22	(10,5,1)	10	300	Linear	0.9522	0.9386
23	(10,5,1)	15	300	Linear	0.9535	0.9416

4.3. Hasil Algoritma SVR

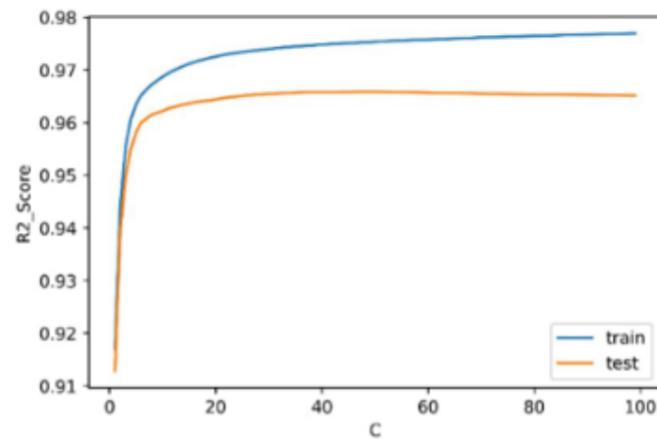
Pada algoritma SVR, parameter penting yang harus disesuaikan adalah koefisien C. Koefisien ini sangat mempengaruhi ukuran margin pada suatu *hyperplane* SVR. Nilai C yang kecil dapat memperlebar *margin* yang memungkinkan peningkatan *error*. Namun, nilai C yang besar dapat mempersempit margin sehingga mengurangi kesalahan dalam prediksi dataset *training*. Pada tahap pengujian atau validasi dataset *test*, model terbaik pada tahap *training* belum tentu dapat menghasilkan hasil prediksi yang optimal pada tahap *test* (*overfitting*). Oleh karena itu, nilai C yang optimal harus ditentukan melalui tahap *cross validation* untuk memperoleh hasil terbaik pada tahap *test*.

Berdasarkan Tabel 4, koefisien C terbaik yang dipilih adalah saat  $C = 49$ , karena skor hasil prediksi dataset *training* dan *test* menghasilkan nilai yang cukup tinggi (*test score* = 0.965883 dan *train score* = 0.975308). Skor *test* dan *train* akan meningkat ketika nilai C semakin besar, tetapi hal tersebut juga mengakibatkan terjadinya *overfitting* karena model terlalu mempelajari data *training* (Gambar 7).

Tabel 4. Performa Prediksi menggunakan SVR berdasarkan Variasi nilai C.

C	Test Score ( $R^2$ Score)	Test Score ( $R^2$ Score)
1	0.912829	0.917015
2	0.939030	0.944186
3	0.949660	0.955360
4	0.954984	0.960659
5	0.958232	0.963580
40	0.965816	0.974815
41	0.965827	0.974874
42	0.965830	0.974930
43	0.965829	0.974988

44	0.965834	0.975043
45	0.965848	0.975097
46	0.965863	0.975150
47	0.965880	0.975202
48	0.965882	0.975255
49	0.965883	0.975308
50	0.965878	0.975363
51	0.965871	0.975414
90	0.965265	0.976723
91	0.965257	0.976745
92	0.965247	0.976766
93	0.965235	0.976786
94	0.965221	0.976807
95	0.965206	0.976829
96	0.965182	0.976850
97	0.965158	0.976872
98	0.965136	0.976894
99	0.965129	0.976916



Gambar 7. Grafik Performa Prediksi menggunakan SVR berdasarkan Variasi Nilai C

4.4. Prediksi Vs pada Sumur X

Prediksi gelombang geser pada keseluruhan data di sumur X ditunjukkan oleh Gambar 8. Data tersebut merupakan data sumur yang digunakan untuk melakukan tahapan *training* dan *test* sehingga ketiga metode tersebut menghasilkan hasil yang sangat baik. Algoritma terbaik dalam prediksi yang dilakukan di data sumur ini adalah algoritma KNN dengan nilai  $R^2$  score = 0.9686 dan nilai  $RMSE =$

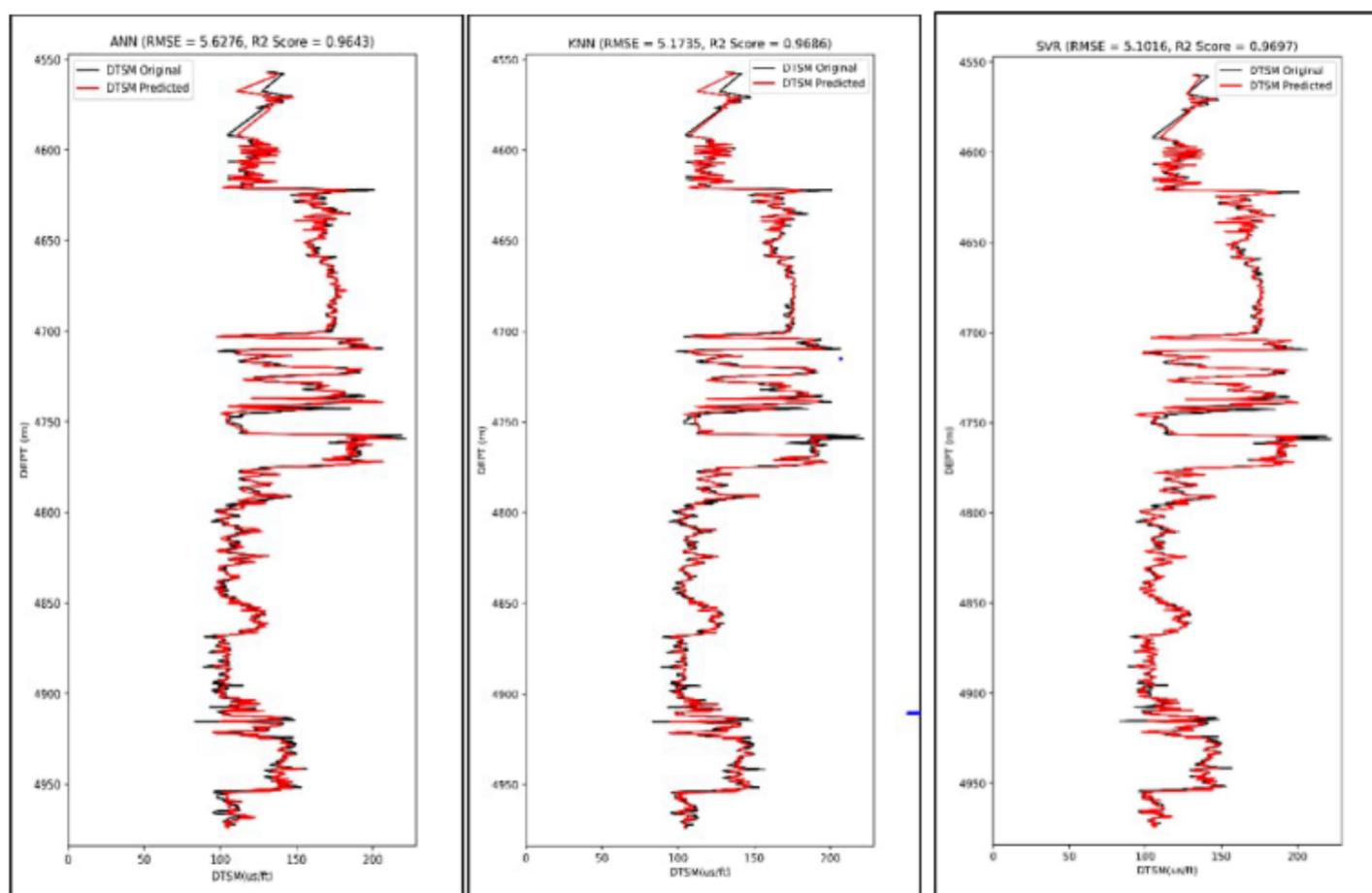
5.1735. Algoritma terbaik kedua adalah ANN dengan nilai  $R^2 \text{ score} = 0.9643$  dan nilai  $RMSE = 5.6276$ , dan algoritma terakhir adalah SVR dengan nilai  $R^2 \text{ score} = 0.9697$  dan nilai  $RMSE = 5.1016$ . Berdasarkan gambar 8, algoritma SVR kurang baik dalam memprediksi kecepatan gelombang geser yang tinggi sehingga menghasilkan skor paling rendah.

#### 4.5. Prediksi Vs pada Sumur Y

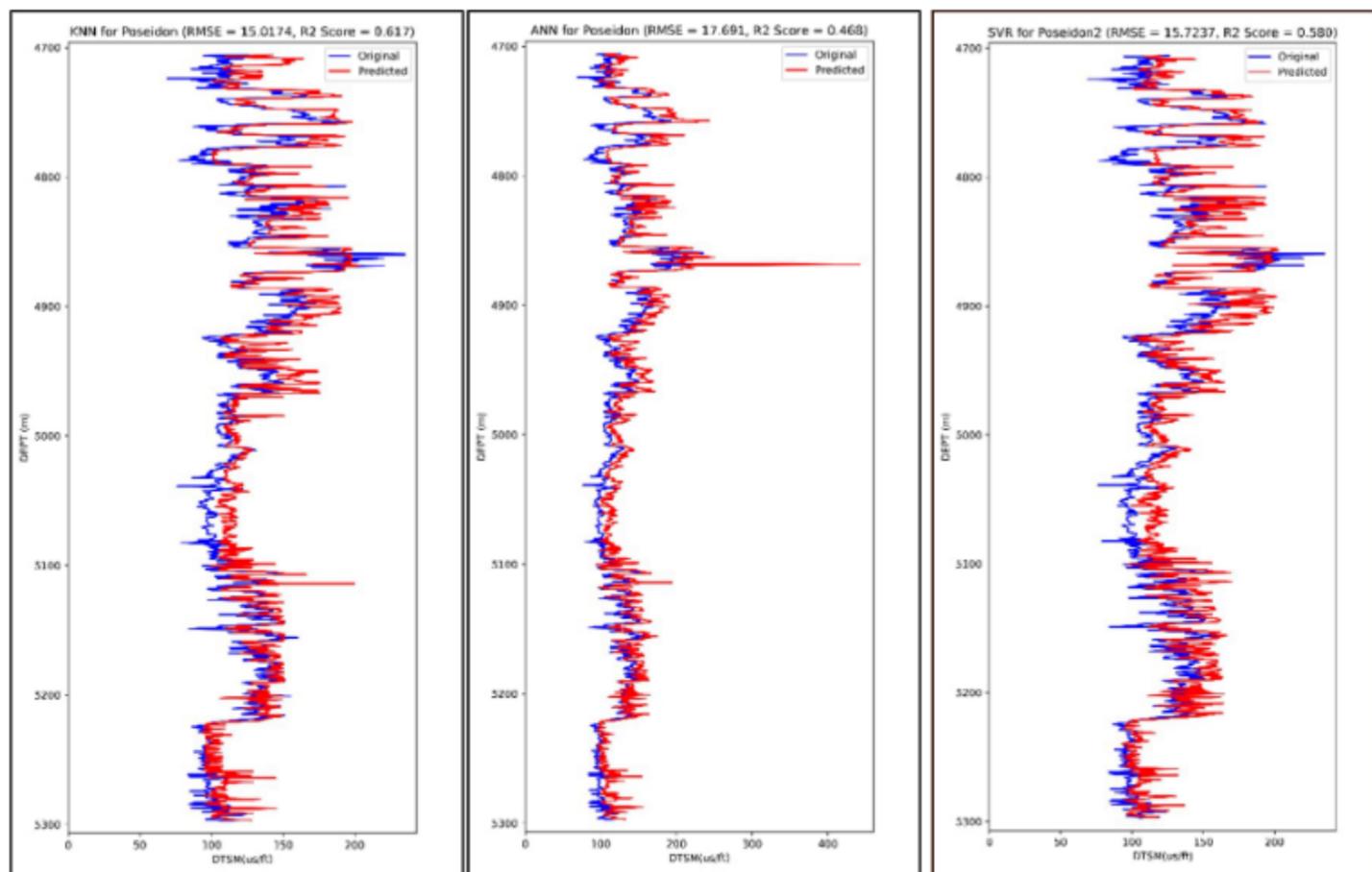
Penelitian ini juga melakukan uji kemampuan model *machine learning* yang telah dibangun dari sumur X untuk memprediksi data sumur Y. Berdasarkan hasil pada Gambar 9, dapat dilihat bahwa keseluruhan hasil prediksi menunjukkan skor prediksi yang lebih buruk daripada prediksi yang dilakukan pada sumur X. Hal ini terjadi karena model atau karakteristik data pada sumur Y memiliki perbedaan dengan sumur

X, sehingga prediksi yang dihasilkan menunjukkan akurasi yang buruk. Hal tersebut adalah salah satu tantangan yang cukup sulit pada penerapan algoritma *machine learning* untuk melakukan prediksi kecepatan gelombang geser. Kasus tersebut dapat diatasi dengan cara melakukan penambahan data *training* sehingga model dapat menghadapi karakteristik data sumur yang variatif.

Algoritma *machine learning* yang paling optimal berdasarkan prediksi pada sumur Y adalah KNN dengan error  $R^2 \text{ score} = 0.6170$  dan nilai  $RMSE = 15.0174$ . Meskipun hasil tersebut sudah cukup baik dibandingkan algoritma lainnya, tetapi *hyperparameter tuning* yang dilakukan masih hanya terbatas pada jumlah tetangga terdekat sehingga untuk meningkatkan performa dapat diuji Kembali dengan *hyperparameter* lainnya. Setelah KNN, algoritma terbaik berikutnya ialah SVR lalu diikuti oleh ANN.



Gambar 8. Hasil Prediksi Gelombang S Menggunakan Algoritma <sup>4</sup>Artificial Neural Network (ANN), K-Nearest Neighbors (KNN), dan Support Vector Regression (SVR) di Sumur X.



Gambar 9. Hasil Prediksi Gelombang S Menggunakan Algoritma <sup>4</sup>Artificial Neural Network (ANN), <sup>15</sup>K-Nearest Neighbors (KNN), dan <sup>17</sup>Support Vector Regression (SVR) di Sumur Y.

## 5. <sup>15</sup>KESIMPULAN

Berdasarkan hasil pengolahan data dan pembahasan diperoleh kesimpulan sebagai berikut.

- Parameter *input* yang digunakan untuk memprediksi kecepatan gelombang geser adalah data *well log* yang terdiri dari data prediktor yaitu *depth*, DTCO, NPFI, ECGR, dan ATRT. Kemudian, data *output* yang akan diprediksi adalah DTSM. Data parameter tersebut dipilih karena memiliki tingkat korelasi yang lebih tinggi dibandingkan data lainnya.
- Algoritma terbaik dalam memprediksi kecepatan gelombang geser dalam kasus ini

adalah KNN dengan  $R^2$  score tertinggi dan RMSE terendah dibandingkan kedua algoritma lainnya berdasarkan hasil prediksi kecepatan gelombang geser pada sumur Y.

## <sup>13</sup>UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen dan teman-teman beserta <sup>17</sup>pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

