PREDIKSI KECEPATAN GELOMBANG GESER (VS) MENGGUNAKAN MACHINE LEARNING

INDRA RIVALDI SIREGAR

https://github.com/insersir/Machine-Learning-for-Oil-and-Gas-Exploration

Outline

PENDAHULUAN

(Latar Belakang)

- Karakterisasi Reservoir
- Membutuhkan Data Kecepatan Gelombang Geser (Vs)
- Data Vs tidak selalu tersedia & Metode prediksi konvensional oleh Castagna (1985) cenderung kompleks
- Implementasi *machine learning*

Komputasi menggunakan Python dengan memanfaatkan module *scikit-learn* oleh Pedregosa *et al*. (2011)

PENDAHULUAN

(Tujuan)

Menentukan model *machine learning* yang paling optimal untuk memprediksi kecepatan gelombang geser dari perbandingan beberapa algoritma machine learning.

Dasar Teori

Machine Learning

Sebuah proses:

- Dari data/sekumpulan observasi yang dimiliki
- Secara otomatis mempelajari pola/aturan yang ada dengan bermacam algoritma

\+ Gunakan pola yang ditemukan untuk membuat keputusan mengenai data/observasi baru

DASAR TEORI

(Algoritma *K-Nearest Neighbor*)

KNN adalah algoritma jenis *supervised learning* yang digunakan untuk mengidentifikasi kemiripan suatu titik baru berdasarkan **jarak terdekat** dari **titik tetangganya** (Mitchel, 1997).

Oleh karena itu, **jarak** merupakan **kunci keberhasilan** dari algoritma ini.

$$d(a,b) = \sqrt{\sum_{i=1}^{n}(xi - yi)^2}$$

Keterangan:

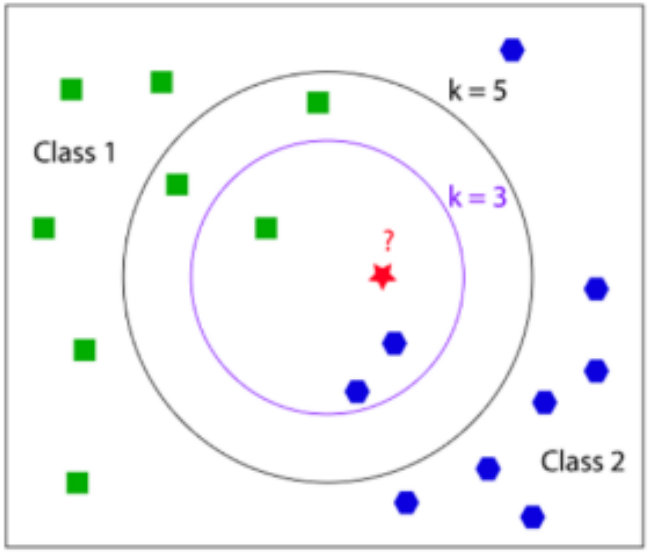
$d(a,b)$: jarak Euclidean

xi : data 1

yi : data 2

i : fitur ke-i

n : jumlah fitur



Gambar 1. Ilustrasi KNN (Mitchell, 1997)

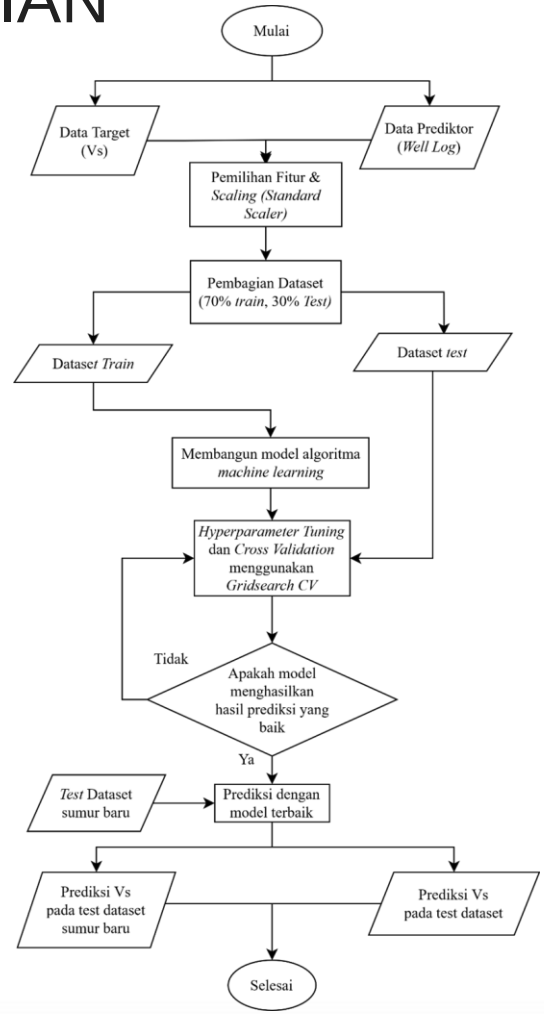
METODE PENELITIAN

(Data Penelitian)

- Data log sumur Kronos-1 dan sumur Poseidon-2 di **Lapangan NW Shelf Australia** yang dapat diakses melalui ***website* SEG Wiki**.
- Sumur Kronos-1 digunakan untuk **membangun model *machine learning*** dari beberapa algoritma.
- Sumur Poseidon-2 digunakan untuk **menguji konsistensi performa** dari setiap model *machine learning*.

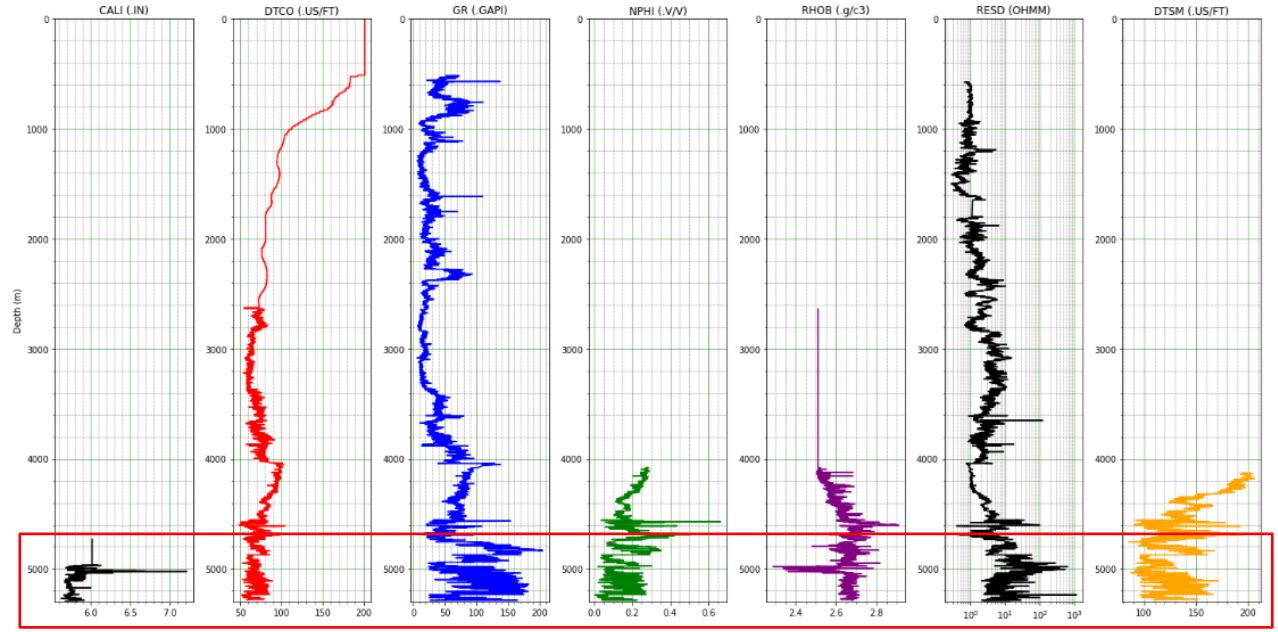
METODE PENELITIAN

Diagram Alir Penelitian



METODE PENELITIAN

Dataset (Initial)



Dataset terdiri dari:

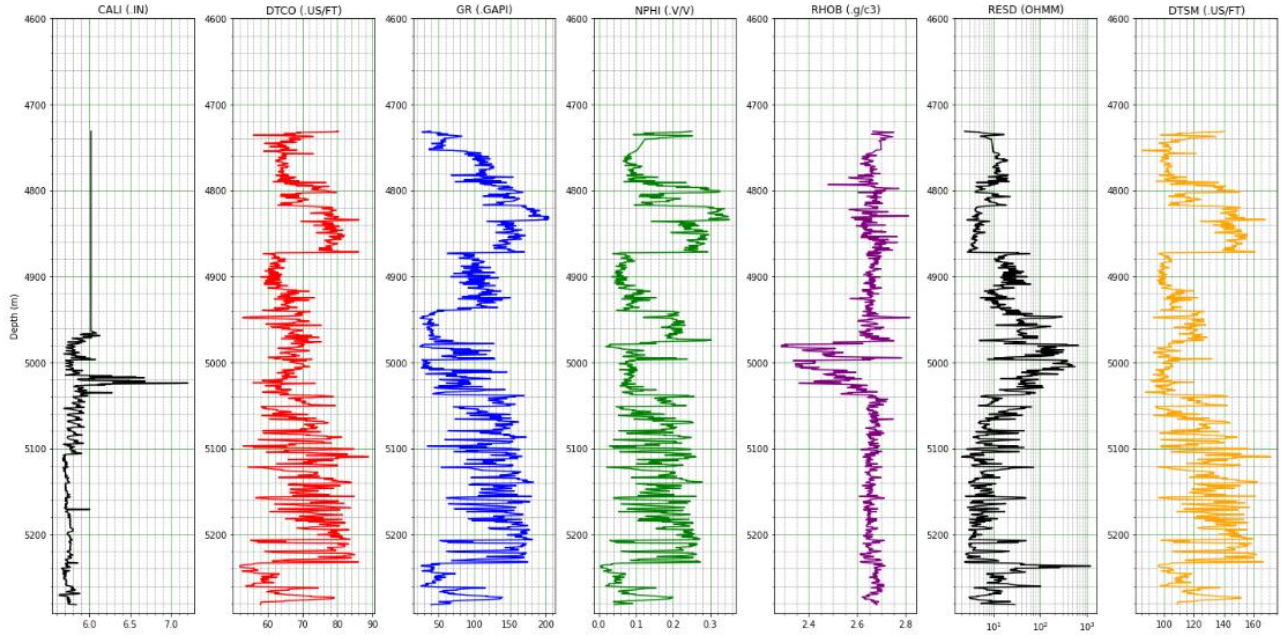
- DEPT .M : Index TVD MSL
- CALI .IN : Caliper
- DTCO .US/F : Delta-T Compressional
- GR .GAPI : Gamma Ray
- NPHI .V/V : Neutron Porosity
- RHOB .G/C3 : Bulk Density
- RESD .OHMM : Deep Resistivity
- **DTSM .US/F : Delta-T Shear**

Gambar 5. Data log prediktor/feature dan target sumur Kronos-1 (Original)

Semua data log yang beririsan dimulai dari DEPTH sekitar 4700 m

METODE PENELITIAN

Dataset (After)



Gambar 5. Data log prediktor dan target sumur Kronos-1 (After)

Dataset terdiri dari:

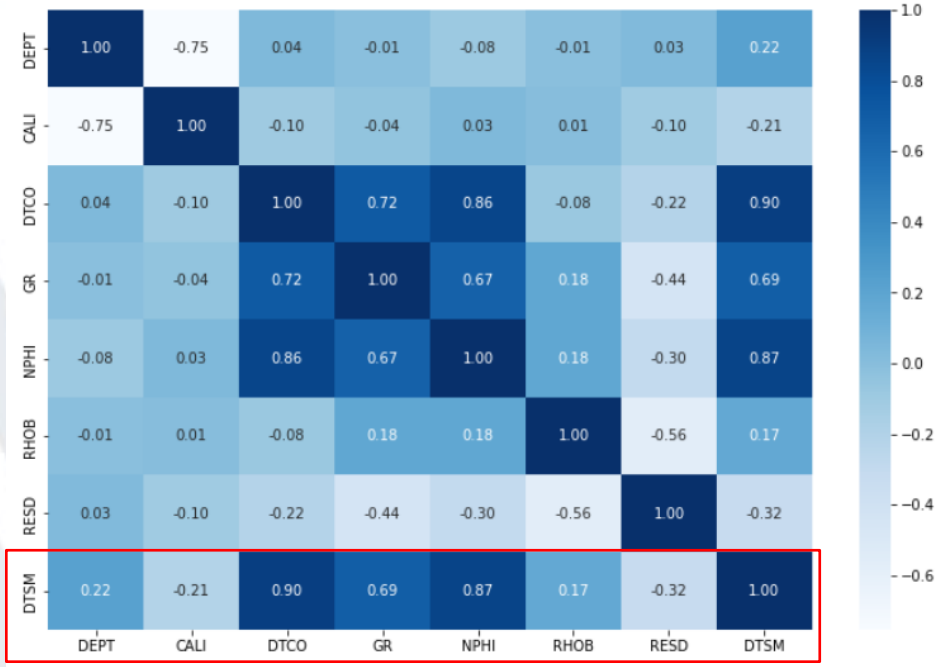
- DEPT .M : Index TVD MSL
- CALI .IN : Caliper
- DTCO .US/F : Delta-T Compressional
- GR .GAPI : Gamma Ray
- NPHI .V/V : Neutron Porosity
- RHOB .G/C3 : Bulk Density
- RESD .OHMM : Deep Resistivity
- **DTSM .US/F : Delta-T Shear**

Jumlah Baris: 3614
Jumlah Kolom: 7

→ **Ketersediaan Data Terbatas**

METODE PENELITIAN

Korelasi Pearson Antar Data Log Sumur



Gambar 4. Korelasi *heatmap* data log sumur Kronos-1

Tabel 1. Nilai korelasi data log Sumur Kronos-1

Log	Correlation with DTSM	Rank
DTCO	0.9	1
NPHI	0.87	2
GR	0.69	3
RESD	0.32	4
CALI	0.21	5
RHOB	0.17	6

DROP →

CALI (*caliper*)
GR (*gamma ray*)
DTCO (*Delta T-compressional*)
NPHI (*neutron porosity*)

RHOB (*bulk density*)
RESD (*deep resistivity*)
DTSM (*Delta T-shear*)

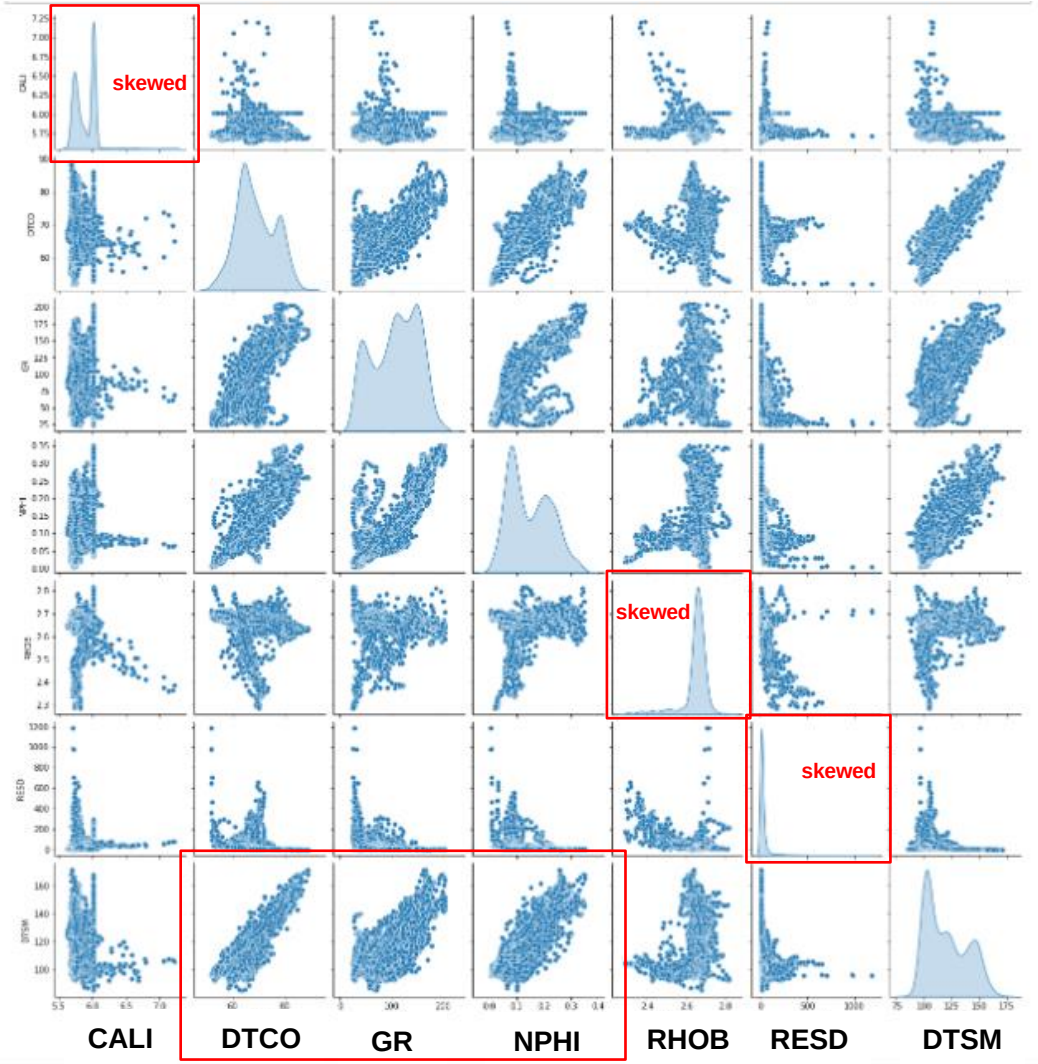
Pada umumnya, nilai korelasi yang rendah tidak digunakan sebagai feature/prediktor

CALI di-drop karena sumur validasi tidak memiliki data CALI

METODE PENELITIAN

Exploratory Data Analysis – Pair Plot

- RESD dan CALI memiliki distribusi data yang **right skewed**, sehingga bisa dilakukan **transformation**.
- RHOB distribusinya **left skewed**, bisa ditransformasi juga.
- DTCO, NPHI, dan GR menunjukkan korelasi positif yang cukup kuat dengan DTSM.



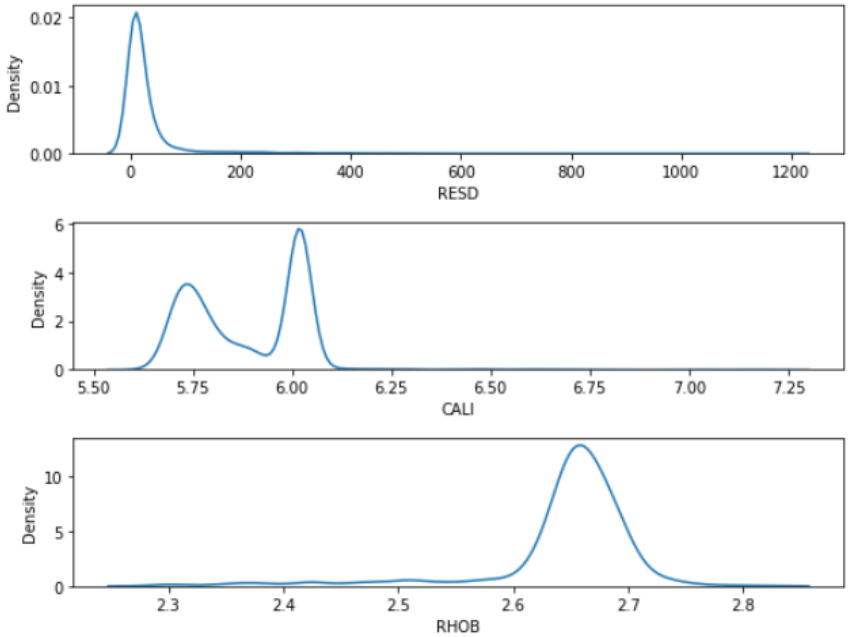
METODE PENELITIAN

Feature Transformation pada data yang skewed distribution

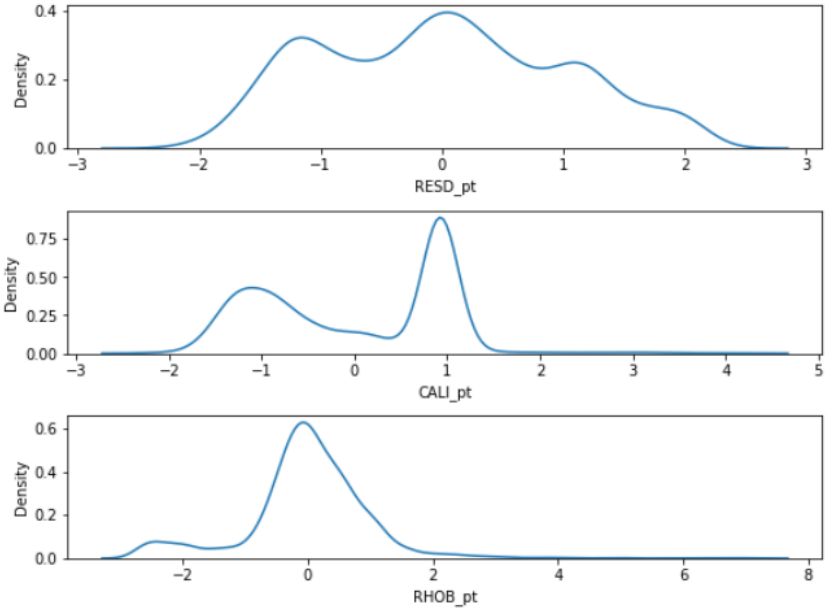
Kernel Density Estimate (KDE) Plot

Transformasi menggunakan metode yeo-johnson (bekerja dengan baik pada left and right skewed distribution)

Before Transformation



After Transformation

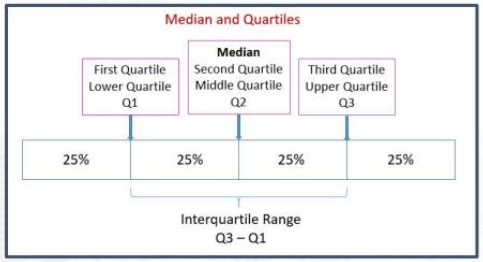


Yeo & Johnson (2000) https://www.jstor.org/stable/2673623

METODE PENELITIAN

Handling Outlier

Menghapus Outlier dengan Metode IQR



IQR: lebar Q3-Q1

Outlier: Lebih ekstrim dari 1.5 IQR dari Q1 atau Q3

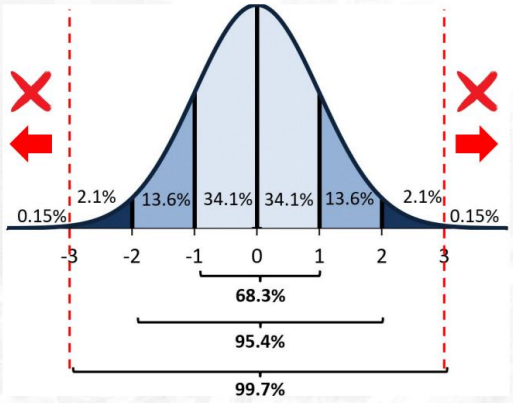
Menghapus Outlier dengan Metode Z-score

Z-score: berapa kali *standard deviation* jarak sebuah nilai dari rata-rata kolom

Outlier: abs(Z-score) > 3

- Kita membuang ~0.3% data paling ekstrim (asumsi data berdistribusi normal)

$$z = \frac{x - \mu}{\sigma}$$

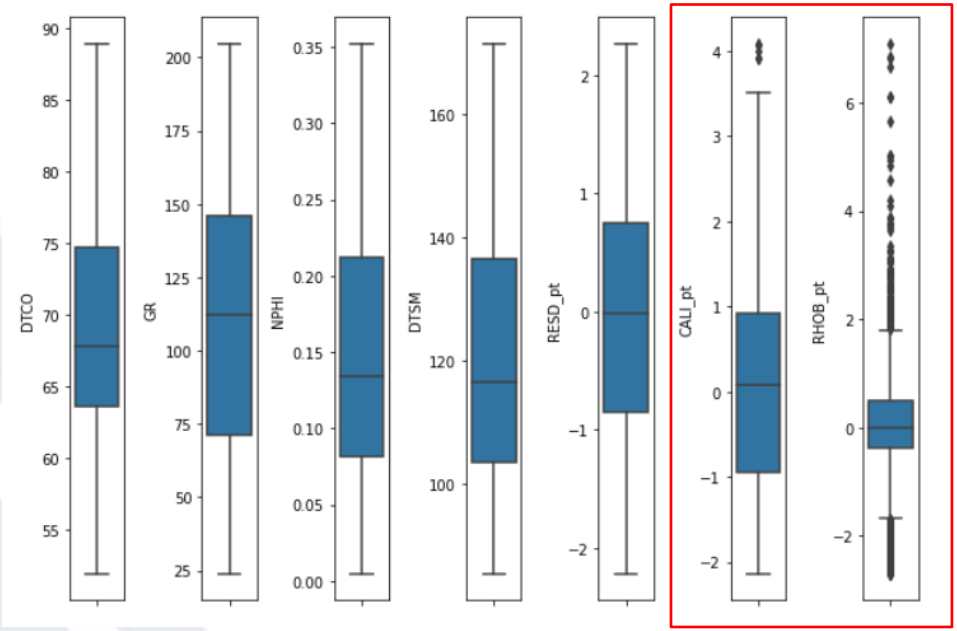


METODE PENELITIAN

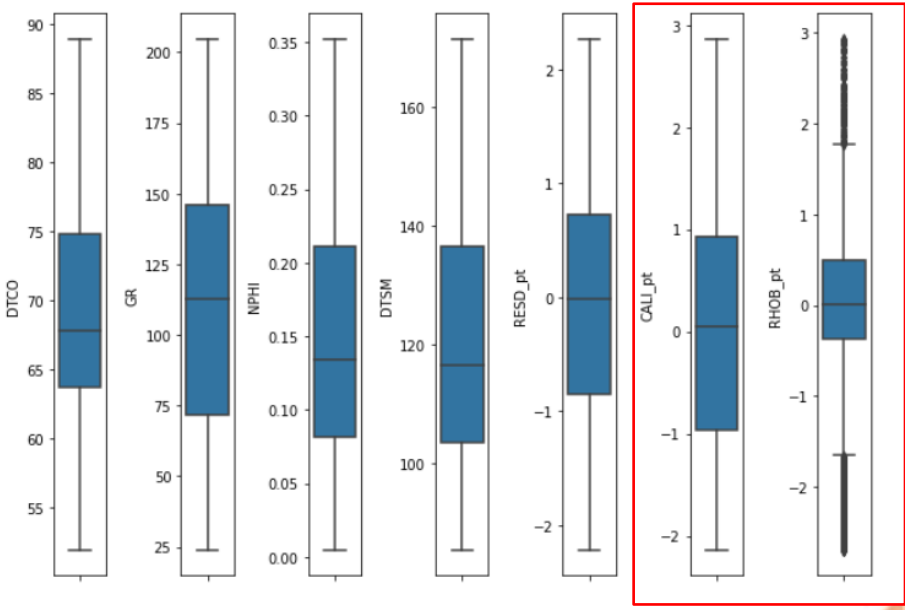
Handling Outlier menggunakan metode z-score

Kernel Density Estimate (KDE) Plot

Boxplot sebelum Handling Outlier



Boxplot setelah Handling Outlier



- **Jumlah data sebelum handling outlier : 3614**
- **Jumlah data setelah handling outlier : 3568 (berkurang 1.27 %)**

Handling outlier -> membuang data outlier

METODE PENELITIAN

Train-Test Split Data

Data Sumur Kronos-1 yang sudah **di-preprocessing** di-split menjadi:

- 70% Training Dataset (2498 baris)
- 30% Testing Dataset (1070 baris)

Standarisasi

Training Dataset dilatih pada algoritma machine learning

Performanya dievaluasi pada Testing Dataset

METODE PENELITIAN

Modeling and Evaluation

No	Model	Train			Test		
		MAE	RMSE	R2 Score	MAE	RMSE	R2 Score
1	Linear Regression	4.66	6.21	0.88	4.71	6.17	0.88
2	Lasso	5.28	6.51	0.84	5.42	6.57	0.84
3	Ridge	4.66	6.21	0.88	4.71	6.17	0.88
4	XGBoost	0.84	1.21	1.00	2.84	4.18	0.95
5	KNN	1.95	3.17	0.97	2.52	3.88	0.96
6	Extra Tree Regression	0.00	0.00	1.00	2.41	3.87	0.96

Model semakin baik ketika:

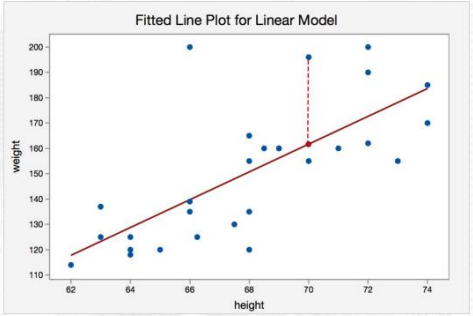
- **R2 Score mendekati 1.**
- **RMSE dan MAE semakin rendah.**

3 model terbaik

METODE PENELITIAN

Model Evaluation

Evaluasi Model: Regresi



Evaluasi yang biasa digunakan adalah dengan menghitung jarak antara hasi prediksi dengan posisi asalnya (error)

1. RMSE (root mean square error)
2. MAE (mean absolute error)

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2}$$

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

Error yang lebih kecil lebih baik.

CASE 1: Evenly distributed errors

ID	Error	|Error|	Error^2
1	2	2	4
2	2	2	4
3	2	2	4
4	2	2	4
5	2	2	4
6	2	2	4
7	2	2	4
8	2	2	4
9	2	2	4
10	2	2	4

MAE	RMSE
2.000	2.000

CASE 2: Small variance in errors

ID	Error	|Error|	Error^2
1	1	1	1
2	1	1	1
3	1	1	1
4	1	1	1
5	1	1	1
6	3	3	9
7	3	3	9
8	3	3	9
9	3	3	9
10	3	3	9

MAE	RMSE
2.000	2.236

CASE 3: Large error outlier

ID	Error	|Error|	Error^2
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
10	20	20	400

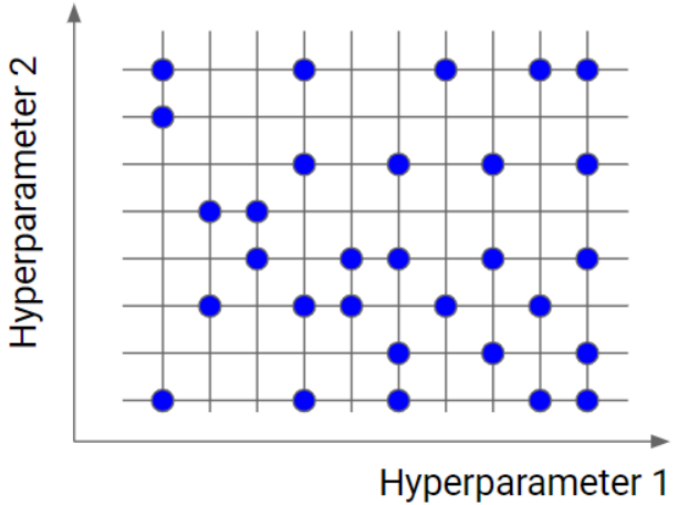
MAE	RMSE
2.000	6.325

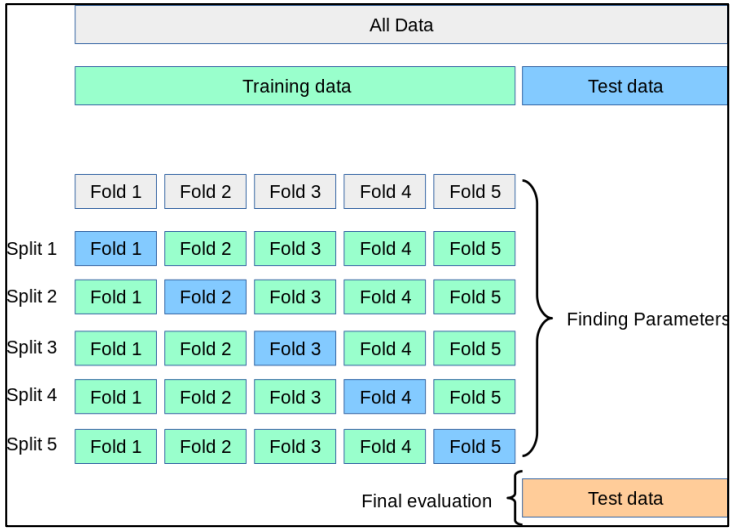
RMSE mempunyai keuntungan dengan memberikan error yang besar jika terdapat outlier, sehingga menghasilkan pengukuran yang tepat untuk beberapa kasus yang lebih sensitif

METODE PENELITIAN

Hyperparameter Tuning

Hyperparameter Tuning: mencari hyperparameter/parameter terbaik dari suatu model ML. Cara mencarinya adalah dengan **metode grid search** atau **random search** dari kombinsi tiap parameter.

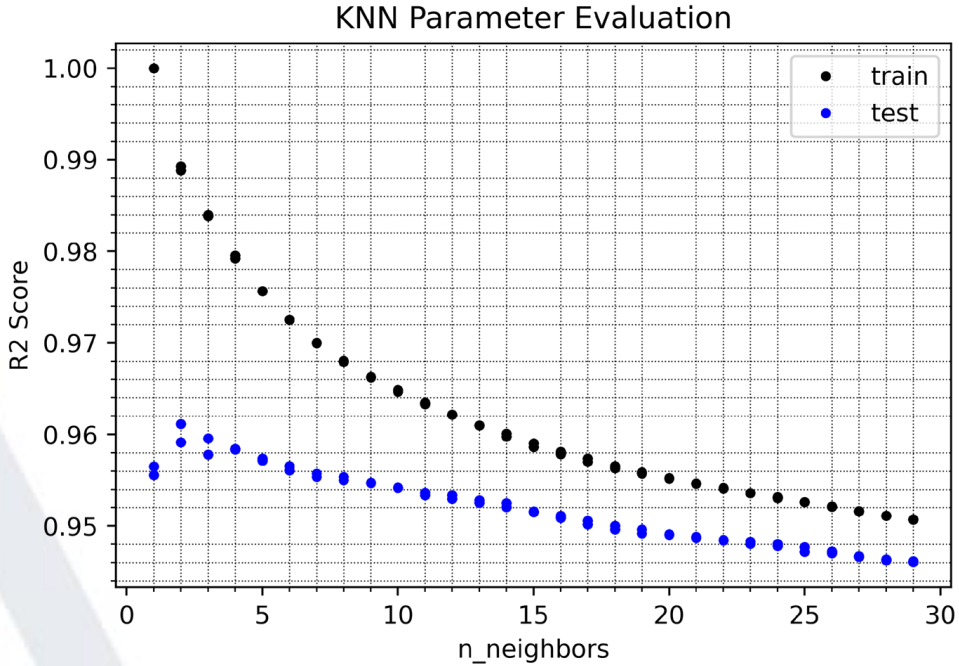




Ilustrasi *K-Fold Cross Validation* (Pedregosa *et al*., 2011)

Hasil dan Pembahasan

Hasil Algoritma KNN

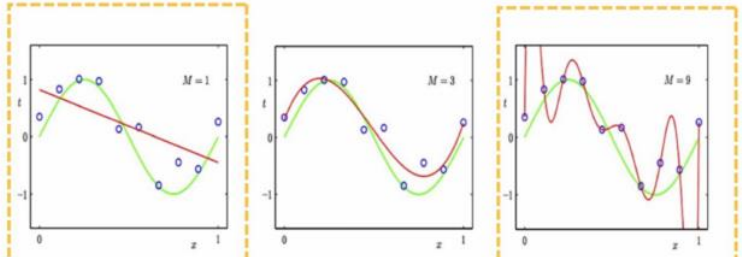


Gambar 6. Grafik Performa Prediksi menggunakan KNN berdasarkan n_neighbors (N)

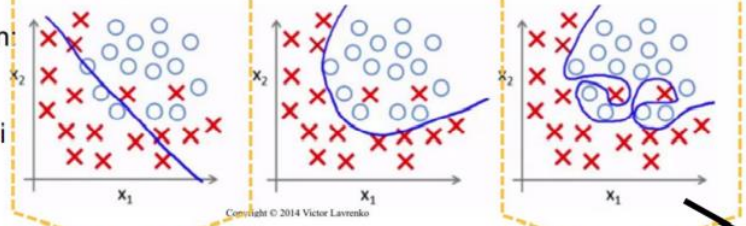
- *Hyperparamater tuning,* memvariasikan jumlah tetangga terdekat (N) dari interval 1 - 30.
- Nilai ***N*** = 1 -> ***overfitting*** (model pintar di data train, tetapi bodoh di data test).
- Nilai ***N*** terbaik diperoleh pada saat ***N=2***.
- Nilai ***N > 2*** performa menurun.

Overfitting/Underfitting

Regression:



Classification:



Model yang mempunyai nilai bias yang tinggi jika model tidak dapat menemukan relasi yang tepat antara variabel dan target output (underfit)

Sedangkan jika mempunyai variance yang tinggi jika model modelnya terlalu sensitif terhadap detail-detail kecil pada data, sehingga tidak dapat melakukan prediksi terhadap data yang lebih general

Hyperparameter Tuning Algoritma XGBoost

- Hyperparameter yang digunakan untuk mentuning best model adalah:
- eta: penyusutan step size untuk mencegah overfitting (a.k.a. learning_rate)
- gamma: minimum loss reduction yang dibutuhkan untuk membuat partisi selanjutnya pada leaf node
- max_depth: kedalaman maksimum tree
- min_child_weight: jumlah weight minimum pada sebuah "child" (partisi), semakin tinggi parameter ini, model semakin konservatif
- colsample_bytree: rasio subsample pada konstruksi tree
- lambda: regularisasi L2, semakin tinggi parameter ini, model semakin konservatif
- alpha: regularisasi L1, semakin tinggi parameter ini, model semakin konservatif
- tree_method: algoritma konstruksi tree yang digunakan pada model XGBoost

METODE PENELITIAN

Hyperparameter Tuning 3 model terbaik

	Model	Train			Test		
		MAE	RMSE	R2 Score	MAE	RMSE	R2 Score
Sebelum Hyperparameter Tuning	KNN	1.95	3.17	0.97	2.52	3.88	0.96
Setelah Hyperparameter Tuning		1.29	2.21	0.99	2.34	3.93	0.96
Sebelum Hyperparameter Tuning	Extra Tree Regressor	0.00	0.00	1.00	2.41	3.87	0.96
Setelah Hyperparameter Tuning		1.55	2.42	0.98	2.54	3.98	0.95
Sebelum Hyperparameter Tuning	XGBoost	0.84	1.21	1.00	2.84	4.18	0.95
Setelah Hyperparameter Tuning		1.46	2.17	0.99	2.78	4.13	0.95

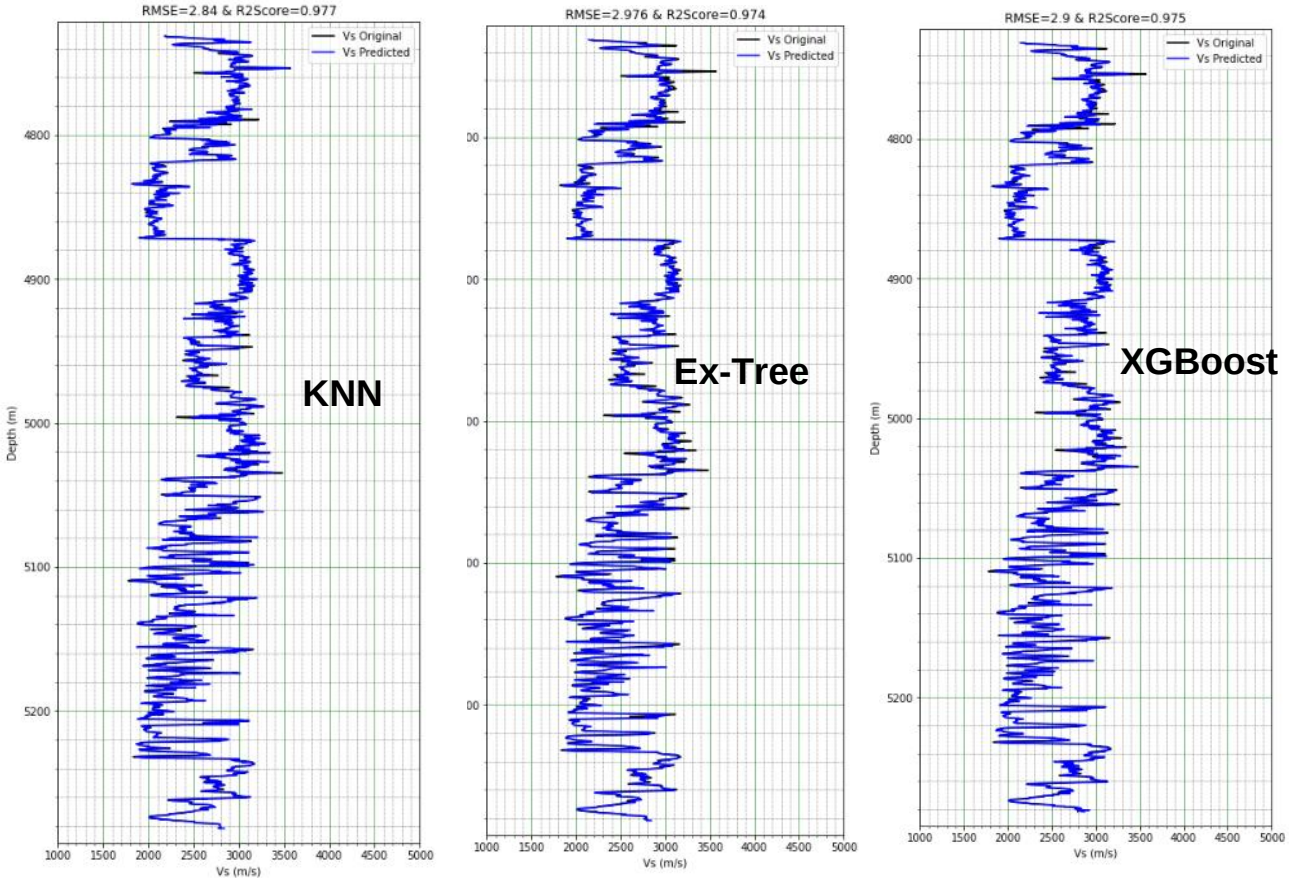
Idealnya setelah hyperparameter tuning:

Baik pada train data maupun test data, RMSE & MAE **menurun** dan R2 Score **meningkat**.

Performa Extra Tree Reg menurun setelah hyperparameter

Hasil dan Pembahasan

Hasil Prediksi Sumur Kronos-1



Tabel 4. Perbandingan performa algoritma di sumur Kronos-1

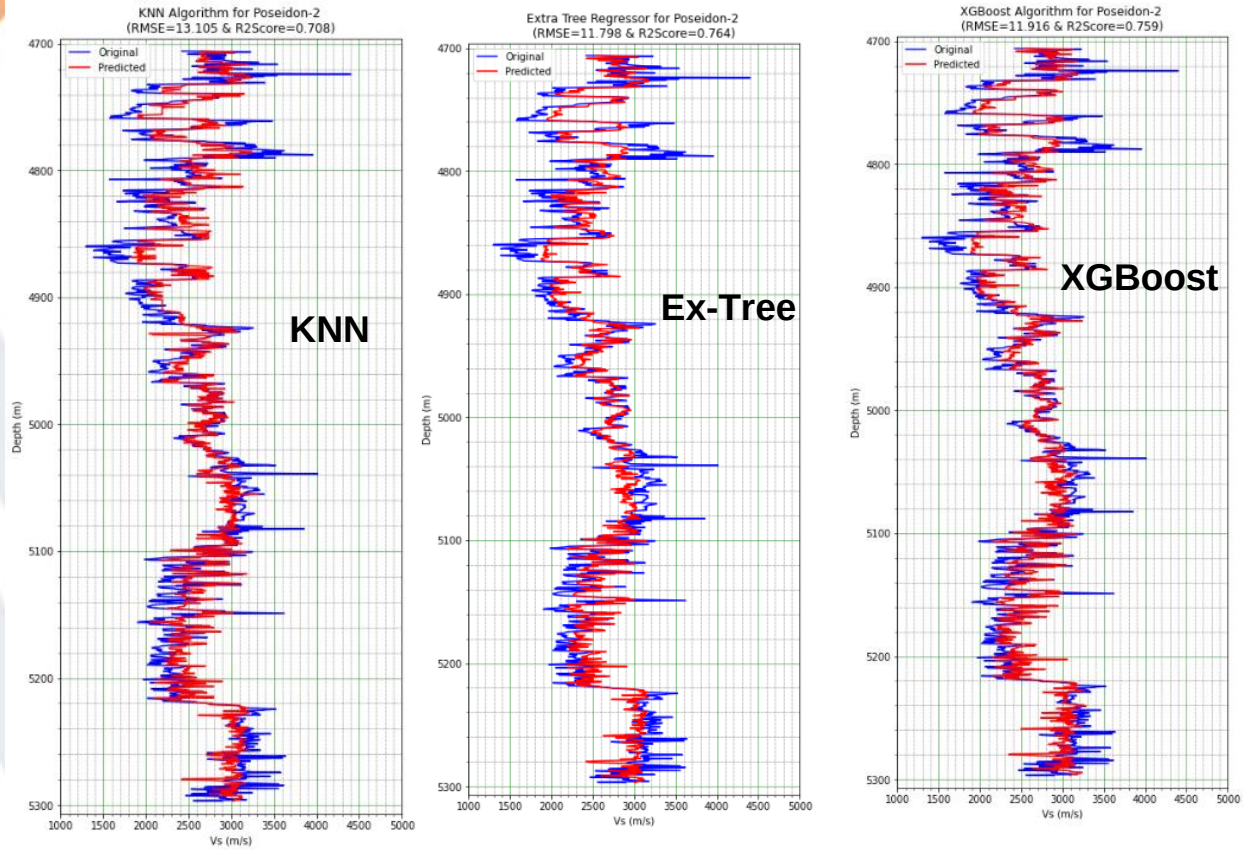
Algorithm	R2Score	RMSE	Rank
KNN	0.977	2.84	1
Ex-Tree	0.974	2.976	2
XGBoost	0.975	2.9	3

Algoritma terbaik berdasarkan hasil prediksi Vs pada Sumur Kronos-1 adalah **KNN**.

Gambar 8. Hasil prediksi Vs dari algoritma KNN, Extra Tree Regressor, dan XGBoost di sumur Kronos-1

Hasil dan Pembahasan

Hasil Prediksi Sumur Poseidon-2



Tabel 5. Perbandingan performa algoritma di sumur Poseidon-2

Algorithm	R2Score	RMSE	Rank
KNN	0.708	13.105	2
Ex-Tree	0.764	11.798	1
XGBoost	0.759	11.916	3

Algoritma terbaik berdasarkan hasil prediksi Vs pada Sumur Poseidon-2 adalah **Ex-Tree**.

Tergeneralisasi/Lebih robust

Gambar 9. Hasil prediksi Vs dari algoritma KNN, Extra Tree Regressor, dan XGBoost di sumur Poseidon-2

SARAN

Dalam penelitian selanjutnya sebaiknya model *machine learning* dap
dengan data *training* yang lebih banyak dan diuji pada beberapa sumur lainnya.

REFERENSI

Bre, F., Gimenez, J. M., Fachinotti, V. D. 2018. “Prediction of wind pressure coefficients on building surfaces using artificial neural networks”. *Energy & Buildings*, 158, 1429–1441. https://doi.org/10.1016/j.enbuild.201 7.11.045.

Castagna, J. P., Batzle, M. L., & Eastwood, R. L. 1985. “Relationships between compressional-wave and shear-wave velocities in clastic silicate rocks”. *Geophysics*, 50(4), 571-581.

Mitchell, T. M. 1997. *Machine Learning*. Burr, Ridge, IL: McGraw Hill, 45(37), 870-877.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, B., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. 2011.

Shmueli, G., Bruce, P. C., Patel, N. R. 2016. “Data Mining for business analytics”. Retrieved from In Data Mining for Business Analytics: Concepts, Techniques and Applications with XLMiners 2.

Sholkopf B. & Smola A. 2002. *Learning with Kernel*. MIT Press.

Wang, L. J., Guo, M., Sawada K., Lin, J., Zhang, J. 2016. “A comparative study of landslide susceptibility maps using logistic regression, frequency ratio, decision tree, weights of evidence and artificial neural network”. *Geosciences Journal*, 20(1), 117-136.

Widiaputra. (2016). *Artificial Neural Network*. Dosen Perbanas. https://dosen.perbanas.id/artificial-neural-network/.


THANK
YOU!
SOUTH AMERICA