Penerapan Metode ARIMA dan Additive Outlier untuk Pendeteksian Anomali dalam Data Monitoring Operasi Transmisi Gas di Jaringan Pipa

Ahmad Azwar Annas[1], Widi Astuti[2], Aditya Firman Ihsan[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]ahmadazw@student.telkomuniversity.ac.id, [2]widiwdu@telkomuniversity.ac.id,
[3]adityaihsan@telkomuniversity.ac.id

Abstrak

Penelitian ini menerapkan metode ARIMA dan *Additive Outlier* untuk pendeteksian anomali dalam data *monitoring* operasi transmisi gas di jaringan pipa yang diambil sekitar bulan Agustus 2020 hingga bulan Juli 2021. Data yang digunakan berasal dari sebuah perusahaan minyak dan gas yang beroperasi di wilayah Laut Natuna. Penelitian ini menerapkan metode ARIMA pada aset tertentu dengan fokus pada variabel tekanan yang menghasilkan model ARIMA(0,1,0) sebagai model terbaik untuk pendeteksian anomali pada data tekanan gas karena model tersebut memiliki nilai AIC paling kecil. Berdasarkan metrik evaluasi, model tersebut memiliki hasil yang unik karena metrik yang digunakan sensitif terhadap data anomali di mana anomali dalam data yang digunakan dalam penelitian ini tidak dibersihkan karena tujuan utama dari penelitian ini adalah pendeteksian anomali tersebut. Peneliti menyarankan untuk pengembangan model pada variabel operasional lainnya dan membandingkan model ARIMA dengan model lain dalam *machine learning*.

Kata kunci : arima, additive outlier, transmisi gas, jaringan pipa

Abstract

This research applies the ARIMA and Additive Outlier method for anomaly detection in gas transmission operation monitoring data in pipelines taken from August 2020 to July 2021. The data used comes from an oil and gas company operating in the Natuna Sea region. This research applies the ARIMA method on a specific asset with a focus on the pressure variable which results in the ARIMA(0,1,0) model as the best model for anomaly detection in gas pressure data because the model has the smallest AIC value. Based on the evaluation metrics, the model has unique results because the metrics used are sensitive to anomalous data where anomalies in the data used in this study are not cleaned because the main purpose of this study is the detection of such anomalies. The researcher suggests developing the model on other operational variables and comparing the ARIMA model with other models in machine learning.

Keywords: arima, additive outlier, gas transmission, pipe networks

1. Pendahuluan

Latar Belakang

Sistem transmisi gas melalui jaringan pipa merupakan elemen kunci dalam infrastruktur energi yang penting. Keberhasilan operasi transmisi gas tidak hanya bermanfaat bagi pengguna industri tetapi juga perumahan. Pemantauan operasi ini sangat penting untuk menjaga keamanan dan keselamatan dalam jaringan pipa dan untuk memastikan pasokan gas yang stabil [4] [6]. Operasi ini memiliki kendala utama yaitu mengidentifikasi dan mengatasi anomali dalam data. Namun, data yang didapatkan dalam operasi ini seringkali rumit dan dalam keadaan tertentu dapat mengandung anomali atau *outlier* yang menunjukkan kejadian tidak biasa atau potensi masalah dalam jaringan pipa. *Additive Outlier* adalah anomali yang sering ditemui dalam data operasi transmisi gas. Banyak faktor yang dapat menyebabkan anomali ini. Contohnya, jika tekanan dalam pipa gas turun secara drastis di bawah kisaran normal 1088.48 hingga 1896.42 psi, hal tersebut dapat menunjukkan adanya kebocoran atau pecahnya pipa gas yang dapat mengarah ke situasi yang berbahaya. Sebaliknya, jika tekanan dalam pipa gas tiba-tiba berada di atas kisaran normal hal tersebut dapat mengindikasikan bahwa ada kemungkinan penyumbatan atau kerusakan di katup yang dapat merusak pipa. Identifikasi *Additive Outlier* dalam data operasi transmisi gas ini dapat menghambat analisis dan prakiraan yang tepat yang dapat menghambat proses pengambilan keputusan dalam ope- rasi transmisi gas [11]. Metode ARIMA (*Autoregressive Integrated Moving Average*) dipilih pada penelitian ini karena metode tersebut memiliki akurasi yang tinggi. Selain itu metode ARIMA juga fleksibel dan dapat disesuaikan sesuai dengan atribut dari data yang menjadikannya pilihan terbaik untuk menganalisis data operasi transmisi gas [15].

Topik dan Batasannya

Berdasarkan latar belakang yang diberikan, penelitian ini bertujuan untuk menggunakan teknik pemodelan ARIMA untuk mengidentifikasi dan menganalisis *Additive Outlier* dalam data tekanan gas yang dikumpulkan dari operasi transmisi gas di wilayah Laut Natuna. Kumpulan data yang digunakan dalam penelitian ini dimulai dari 1 Agustus 2020 hingga 31 Juli 2021, dengan fokus khusus pada variabel tunggal tekanan gas.

Tujuan

Penelitian ini bertujuan untuk meningkatkan akurasi deteksi *Additive Outlier* dalam data operasi transmisi gas yang pada akhirnya dapat mempercepat respons terhadap anomali yang teridentifikasi. Selain itu penelitian ini berupaya untuk membuat model ARIMA yang dapat diterapkan oleh operator dalam operasi transmisi gas dalam jaringan pipa supaya dapat meningkatkan pengawasan dan administrasi kegiatan operasi transmisi gas. Hasil dari penelitian ini diharapkan dapat meningkatkan keamanan dan efektivitas operasi transmisi gas dalam jaringan pipa secara signifikan dan juga membantu dalam pengambilan keputusan yang tepat dalam pengelolaan operasi transmisi gas.

2. Studi Terkait

Banyak penelitian yang telah dilakukan sebelumnya, contohnya salah satu penelitian oleh Ahmar, dkk. Membahas pendeteksian dan pembersihan data yang mengandung *Additive Outlier* pada model ARIMA. Hasil penelitian tersebut menunjukan bahwa ada peningkatan dalam penerapan teknik tersebut [3]. Penelitian lain yang berfokus dalam konteks ekonomi menunjukan bahwa penerapan metode SARIMA (*Seasonal Autoregressive Integrated Moving Average*) pada data yang mengandung outlier memberikan hasil yang tidak akurat [2]. Selain itu, metode ARIMA juga mampu menangani data yang kompleks ketika diterapkan dalam prakiraan lingkungan untuk memprediksi tren yang dipengaruhi oleh perubahan iklim [10]. Ada penelitian lain yang membandingkan beberapa model seperti LSTM, ARIMA, dan hybrid dimana ARIMA unggul dalam prediksi tren PDB karena dapat menangani data non-stasioner dengan baik [8]. Terakhir, metode ARIMA juga diterapkan dalam penelitian yang dilakukan oleh Vignesh, dkk dalam data transportasi dan sinyal digital dimana pemilihan model yang tepat berdasarkan sifat data itu penting untuk meningkatkan akurasi prakiraan [5].

2.1 Landasan Teori

2.1.1 *Data Mining*

Menurut penelitian yang dilakukan oleh Purwadi, dkk. *Data Mining* merupakan kegiatan untuk mendapatkan informasi pola dalam sekumpulan data yang besar menggunakan berbagai pendekatan seperti ilmu kecerdasan buatan, teknik statistik dan *machine learning* yang bertujuan untuk mendapatkan pengetahuan lebih lanjut mengenai data [13].

2.1.2 *Time Series*

Time Series adalah data yang diambil pada interval waktu yang teratur dari pengukuran sebuah variabel. Data ini sering digunakan untuk menganalisis tren dan pola dalam data, memperkirakan nilai masa depan, dan mengidentifikasi faktor-faktor yang mempengaruhi variabel tersebut. Beberapa contoh dari data *Time Series* meliputi harga saham, suhu, tingkat pengangguran, tingkat inflasi mata uang yang dikumpulkan setiap hari, setiap bulan, setiap tahun, atau pada interval waktu teratur lainnya [19].

Data *Time Series* biasanya menunjukkan pola musiman, tren, dan fluktuasi acak. Ada beberapa teknik khusus yang diperlukan untuk menganalisis data tersebut. Beberapa teknik yang sering digunakan dalam analisis data *Time Series* seperti metode *smoothing*, regresi, ARIMA (*Autoregressive Integrated Moving Average*), dan pemodelan *neural network*. Analisis data *Time Series* sering digunakan dalam berbagai bidang seperti ekonomi, meteorologi, dan ilmu sosial. Teknik analisis *Time Series* juga digunakan dalam *machine learning* untuk memprediksi nilai masa depan dari sebuah variabel berdasarkan data historisnya [18].

2.1.3 *Forecasting*

Forecasting atau peramalan adalah strategi untuk memprediksi nilai masa depan berdasarkan data historis dan pengetahuan mengenai data saat ini. Metodologi yang digunakan dalam *Forecasting* dapat dibagi menjadi dua kategori yaitu metode kualitatif dan metode kuantitatif. Ketika prediksi dibentuk berdasarkan informasi subjektif atau penilaian dari ahli, maka pendekatan kualitatif akan digunakan. Sebaliknya, jika peramalan menggunakan data numerik atau alat analisis statistik, maka metode kuantitatif akan digunakan [14].

2.2 Data Anomali

Proses deteksi data anomali adalah proses mengidentifikasi potensi masalah pada data awal, seperti keberadaan *missing* data (elemen data yang hilang), ketidaklengkapannya dalam runtun waktu, dan normalisasi data. Tahap ini menjadi persiapan untuk memastikan bahwa data yang akan digunakan dalam penelitian telah siap sebagai input dalam berbagai proses peramalan. Sebelum data diimplementasikan dalam model dilakukan pembersihan data yang melibatkan seleksi data, identifikasi elemen data hilang dan lainnya [12].

Proses deteksi data anomali merupakan aktivitas analisis data yang penting untuk mengidentifikasi informasi yang tidak normal dalam aliran lalu lintas jaringan. Langkah ini bertujuan untuk mendukung manajemen dan penanganan masalah keamanan dalam jaringan [9].

2.2.1 *Additive Outlier*

Dalam analisis data *time series*, terkadang kita menemukan titik data yang berbeda secara signifikan dari pola keseluruhan. Ini disebut sebagai *Additive Outlier*. Bayangkan kita mengamati data tekanan gas pada pipa setiap hari. Pada umumnya, tekanan ini berada pada tingkat yang stabil, tetapi suatu hari, mungkin karena masalah teknis atau kesalahan pengukuran, ada lonjakan tekanan yang tidak biasa. Lonjakan ini adalah contoh dari *Additive Outlier*. *Additive Outlier* adalah anomali yang mempengaruhi satu titik data tertentu dan tidak merefleksikan perubahan jangka panjang atau tren dalam data. Deteksi *Additive Outlier* penting karena membantu kita mengidentifikasi kejadian atau kesalahan yang memerlukan perhatian khusus. Dengan menangani *outlier* ini, kita bisa memastikan bahwa analisis data kita lebih akurat dan tidak dipengaruhi oleh kesalahan atau kejadian sementara [7].

Ketika melakukan analisis data *time series*, kita sering menemukan titik data yang berbeda secara signifikan dibandingkan dengan pola keseluruhan data. Hal tersebut dapat disebut sebagai *Additive Outlier*. Ketika kita sedang mengamati tekanan gas dalam jaringan pipa pada umumnya tekanan yang kita amati berada pada tingkat yang stabil. Namun, suatu hari terjadi lonjakan tekanan yang tidak biasa. Lonjakan tersebut adalah contoh dari *Additive Outlier*. *Additive Outlier* adalah anomali yang mempengaruhi satu titik data tertentu yang tidak ada refleksinya dalam pola data dalam jangka panjang. Deteksi *Additive Outlier* penting karena dapat membantu kita mengidentifikasi kejadian yang memerlukan perhatian khusus. [7].

Secara matematis, jika Y_t adalah pengamatan pada waktu t, dan ε_t adalah *noise* atau *error* yang diasumsikan mengikuti distribusi normal, maka *Additive Outlier* dapat dimodelkan sebagai berikut:

$$Y_t = \mu_t + AO_t + \varepsilon_t$$

dimana μ_t adalah komponen tren atau musiman dari deret waktu, dan AO_t adalah komponen *outlier* yang bersifat aditif pada waktu t tertentu. Biasanya, AO_t bernilai nol untuk semua t kecuali pada titik di mana *outlier* terjadi.

2.2.2 ARIMA

Model ARIMA (*Autoregressive Integrated Moving Average*) adalah teknik yang digunakan untuk menganalisis data *time series*, yaitu data yang diambil secara teratur dari waktu ke waktu, dan untuk memprediksi tren di masa depan. Model ini bekerja dengan menggabungkan tiga konsep utama:

1. ***Autoregressive (AR)***: Ini adalah bagian di mana model menggunakan nilai-nilai masa lalu dari data untuk memprediksi nilai masa depan. Jika kita ingin memprediksi suhu besok, kita bisa melihat suhu pada hari-hari sebelumnya untuk membuat prediksi. Secara matematis, ini digambarkan sebagai:

   $$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + e_t$$

   Dimana Y_t adalah nilai saat ini, c adalah konstanta, ϕ adalah koefisien, dan e_t adalah kesalahan yang tidak dapat diprediksi.

2. ***Integrated (I)***: Proses ini melibatkan pengurangan tren dalam data agar data menjadi stasioner. Stasioner berarti data tidak memiliki tren atau pola yang berubah seiring waktu. Untuk membuat data menjadi stasioner, kita menghitung perbedaan antara nilai saat ini dan nilai sebelumnya.

3. ***Moving Average (MA)***: Ini melibatkan penggunaan kesalahan dari prediksi sebelumnya untuk memprediksi nilai saat ini. Dengan kata lain, jika ada kesalahan dalam prediksi sebelumnya, bagian ini membantu memperbaikinya. Secara matematis, ini digambarkan sebagai:

   $$Y_t = c + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} + e_t$$

   Dimana θ adalah koefisien dari kesalahan prediksi sebelumnya.

Model ARIMA dinyatakan sebagai ARIMA(p, d, q), dimana

- p adalah urutan dari bagian *autoregressive*,
- d adalah jumlah *differencing* yang dilakukan untuk membuat data menjadi stasioner,
- q adalah urutan dari bagian *moving average* [1] [16].

Mari kita lihat contoh sederhana dengan menerapkan ARIMA(1,1,1) ke dalam sebuah data *time series* perekaman tekanan gas perjam.

1. **Autoregressive(AR(1))**: Model ARIMA akan menggunakan nilai tekanan gas pada jam sebelumnya untuk memprediksi tekanan gas saat ini.

   Contoh: Jika tekanan gas pada satu jam yang lalu adalah 100 psi, model mungkin akan memprediksi bahwa nilai pada jam ini dengan mempertimbangkan nilai tekanan gas sebesar 100 psi pada satu jam yang lalu.

2. **Integrated(I(1))**: Model ARIMA akan menggunakan perbedaan antara tekanan gas pada jam saat ini dan satu jam yang lalu untuk menghilangkan tren apa pun.

   Contoh: Jika tekanan gas pada satu jam yang lalu adalah 100 psi dan tekanan gas pada jam saat ini adalah 102 psi, model ARIMA akan bekerja dengan selisih sebanyak +2 psi.

3. **Moving Average(MA(1))**: Model ARIMA akan menggunakan kesalahan pada prediksi satu jam yang lalu untuk menyesuaikan prediksi pada jam saat ini.

   Contoh: Jika prediksi pada satu jam yang lalu meleset sebesar +1 psi, model akan memperhitungkan kesalahan tersebut untuk memperbaiki prediksi saat ini.

2.2.3 *Mean Squared Error*

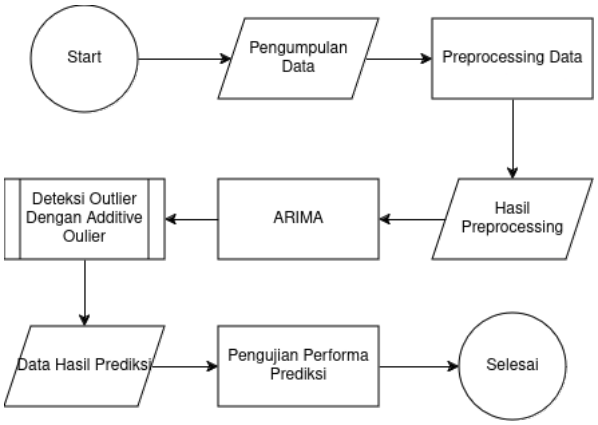
Mean Squared Error (MSE) sering digunakan sebagai metrik dalam pemodelan statistik dan *machine learning* untuk mengevaluasi akurasi dari sebuah model prediksi. MSE mengukur rata-rata kuadrat dari kesalahan, yang merupakan perbedaan antara nilai prediksi dengan nilai aktual [17]. Secara matematis MSE dapat dinyatakan sebagai:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(A_i - P_i)^2$$

Dimana n adalah total jumlah pengamatan, A_i adalah nilai aktual pada pengamatan ke-i, dan P_i adalah nilai prediksi pada pengamatan ke-i. MSE akan selalu bernilai positif, semakin kecil nilai MSE makan semakin baik kinerja model tersebut. Nilai MSE yang kecil dapat menunjukkan bahwa prediksi model mendekati nilai sebenarnya yang dapat disimpulkan bahwa model memiliki tingkat akurasi yang tinggi, Sebaliknya, jika nilai MSE terbilang besar maka model memiliki banyak kesalahan dalam prediksi.

3. Perancangan Sistem

Dataset yang digunakan dalam penelitian ini merupakan data operasional transmisi gas dalam jaringan pipa yang diperoleh dari sebuah perusahaan minyak dan gas yang beroperasi di wilayah Laut Natuna. Setelah pengumpulan data selesai dilakukan maka data tersebut akan masuk ke proses *pre-processing* yang kemudian data tersebut akan dibagi menjadi data *train* dan data *test*. Di mana, data *train* akan digunakan untuk melatih model ARIMA dan data *test* akan digunakan untuk evaluasi model yang didapatkan.



Gambar 1. Alur Proses

3.1 *Dataset*

Dataset yang akan digunakan dalam penelitian ini direkam mulai dari bulan Agustus 2020 hingga bulan Juli 2021, ada beberapa *variable* di dalam *dataset* seperti:

- **Tanggal (DATE STAMP)**: Menunjukkan waktu pengambilan data yang dicatat dalam format tanggal dan waktu.
- **ID Aset (ASSET ID)**: Mengidentifikasi aset spesifik dalam jaringan pipa yang sedang dipantau.
- **Tekanan (PRESSURE)**: Mengukur tekanan gas dalam pipa pada waktu tertentu.
- **Suhu (TEMPERATURE)**: Mencatat suhu gas dalam pipa saat data diambil.
- **Tingkat Energi (ENERGY RATE)**: Mengukur tingkat energi yang terkait dengan aliran gas.
- **Tingkat Volume (VOLUME RATE)**: Mengukur volume gas yang mengalir melalui pipa.

Contoh data mentah dari *dataset* ini dapat dilihat pada Tabel 1

Date Stamp	***Asset ID***	***Pressure***	***Temperature***	***Energy Rate***	***Volume Rate***
2020-08-01 01:00:00	133071	590.06	82.60	24.73	22.52
2020-08-01 01:00:00	133004	1478.23	88.57	41.88	41.56
2020-08-01 01:00:00	133060	1269.56	84.88	201.54	183.81
2020-08-01 01:00:00	133002	1470.80	81.05	201.54	171.31
2020-08-01 01:00:00	133003	1420.27	85.06	82.61	73.71

Tabel 1. Contoh Data Mentah

3.2 Pengolahan Data Awal

Untuk memastikan data yang digunakan berkualitas dan memenuhi syarat sebagai data *time series*, akan dilakukan beberapa pengolahan data, dalam penelitian ini *variable* '*PRESSURE*' dan '*ASSET_ID*' pipa dengan kode '133002' akan dipilih sebagai fokus dari penelitian. Sebelumnya variabel '*DATE_STAMP*' akan di format ulang ke dalam bentuk yang lebih mudah digunakan untuk penelitian. Selain itu akan dilakukan *Augmented Dickey-Fuller*

Test pada data yang digunakan untuk mengecek apakah perlu dilakukan *Differencing* terlebih dahulu sebelum data dapat digunakan untuk membuat model. Dan yang terakhir data akan dibagi dalam rasio 90:10 untuk *training* dan *test*.

3.3 *ARIMA*

Setelah proses pengolahan data awal selesai dilakukan, selanjutnya akan dilakukan *training* model ARIMA. Data yang digunakan merupakan data tekanan gas pada jaringan pipa.

3.4 Deteksi *Additive Outlier*

Setelah *training* model dilakukan, *residual* dari model akan dianalisis di mana jika *residual* dari model melebihi tiga standar deviasi dari rata-rata maka akan dianggap sebagai *Additive Outlier*.

3.5 Evaluasi Model

Model ARIMA yang telah dibangun kemudian akan diukur performanya menggunakan metriks evaluasi MSE (*Mean Squared Error*) yang akan memberikan ukuran rata-rata kesalahan akurasi prediksi dari model.

4. Evaluasi

4.1 *Exploratory Data Analysis*

4.1.1 Gambaran Umum Data

Dataset yang digunakan dalam penelitian ini memiliki 61.313 baris dengan 21 variabel. Setiap baris data berhubungan dengan catatan dengan waktu untuk berbagai parameter yang terkait dengan operasi transmisi gas di jaringan pipa. Variabel-variabel tersebut meliputi 'DATE_STAMP', 'ASSET_ID', 'PRESSURE', 'TEMPERATURE', 'ENERGY_RATE', 'VOLUME_RATE', dan berbagai metrik komposisi gas seperti 'C1', 'C2', 'C3', dan lain-lain.

Tabel 2 memberikan gambaran umum mengenai struktur dataset termasuk tipe data dan julah entri bukan nol untuk setiap variabel. Pada variabel 'C1' dan 'HCDP' mengindikasikan adanya pengukuran pada variabel tersebut yang hilang atau tidak tersedia selama periode perekaman.

Variable	**Data Type**	**Non-Null Count**
DATE_STAMP	datetime64	61,313
ASSET_ID	int64	61,313
PRESSURE	float64	61,313
TEMPERATURE	float64	61,313
ENERGY_RATE	float64	61,313
VOLUME_RATE	float64	61,313
C1	float64	0
C2	float64	61,313
C3	float64	61,313
IC4	float64	61,313
NC4	float64	61,313
IC5	float64	61,313
NC5	float64	61,313
C6	float64	61,313
C7	float64	61,313
C8	float64	61,313
C9	float64	61,313
N2	float64	61,313
CO2	float64	61,313
H2O	float64	61,313
HCDP	float64	0

Tabel 2. Struktur Data

Dataset tersebut juga memiliki tujuh 'ASSET_ID' yang berbeda seperti '133001', '133002', '133060' dan lainnya.

4.1.2 *Summary Statistics*

Tabel 3 memberikan gambaran yang lebih rinci menganai *dataset*. Contohnya pada variabel ‘PRESSURE‘ memiliki nilai rata-rata 1169,38 dengan rentang nilai dari -5,27 hingga 1676,45, variabel tersebut juga memiliki standar deviasi sebesar 420,87. Variabilitas dalam variabel tersebut dapat menunjukkan potensi anomali yang mungkin memerlukan penyelidikan lebih lanjut.

Variable	Count	Mean	Min	25%	50%	75%	Max	Std. Dev.
DATE_STAMP	61313	2021-01-30	2020-08-01	2020-10-31	2021-01-30	2021-05-01	2021-07-31	NaN
ASSET_ID	61313	133030.14	133001.00	133002.00	133004.00	133070.00	133071.00	32.10
PRESSURE	61313	1169.38	-5.27	590.37	1396.58	1502.54	1676.45	420.87
TEMPERATURE	61313	89.26	0.00	77.60	83.86	98.18	173.19	16.70
ENERGY_RATE	61313	129.70	0.00	62.15	84.33	199.27	91865.32	534.87
VOLUME_RATE	61313	118.02	0.00	58.17	78.82	184.17	32426.25	203.82
C1	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN
C2	61313	29.35	0.00	3.83	4.36	5.19	102885.95	1602.32
C3	61313	1.91	0.00	1.68	1.94	2.46	5.74	0.87
IC4	61313	0.61	0.00	0.52	0.60	0.71	85.75	1.18
NC4	61313	0.07	0.00	0.01	0.05	0.13	1.74	0.07
IC5	61313	12.76	0.00	0.03	0.11	0.22	36884.96	682.23
NC5	61313	0.07	0.00	0.01	0.05	0.13	1.74	0.07
C6	61313	0.07	0.00	0.00	0.05	0.09	56.31	0.82
C7	61313	21.08	0.00	0.00	0.02	0.03	92217.63	1392.80
C8	61313	14.26	0.00	0.00	0.01	0.01	62840.27	713.52
C9	61313	0.01	0.00	0.00	0.00	0.00	12.12	0.33
N2	61313	0.44	0.00	0.37	0.40	0.50	1.40	0.16
CO2	61313	20.20	0.00	1.68	2.04	2.16	64007.62	866.16
H2O	61313	0.01	0.00	0.01	0.02	0.02	15.45	0.06
HCDP	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Tabel 3. *Summary Statistics*

4.1.3 Analisis Data yang Hilang

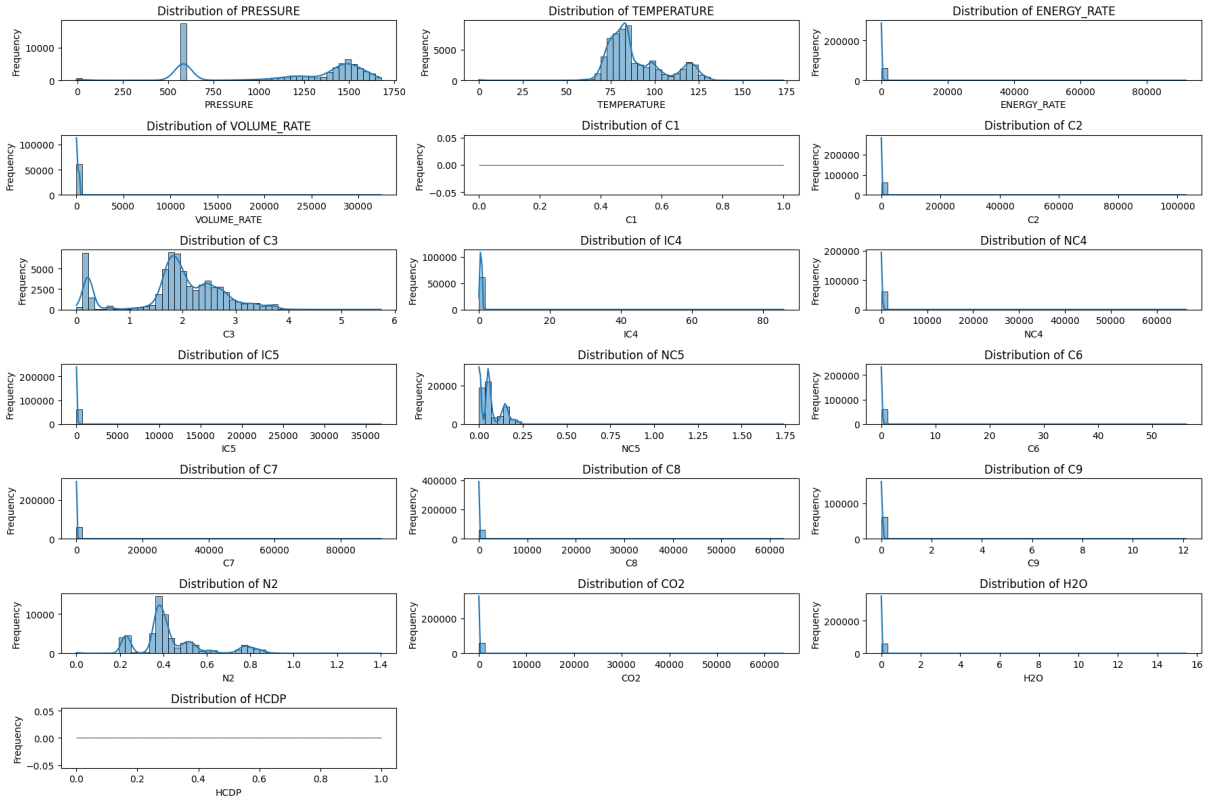
Analisis dari *dataset* menunjukkan bahwa kebanyakan variabel memiliki data yang lengkap dengan 0% nilai yang hilang. Namun, dua variabel ‘C1‘ dan ‘HCDP‘ yang merupakan pengecualian dimana 100% datanya hilang. Hal ini dapat mengindikasikan bahwa tidak ada pengamatan pada dua variabel tersebut selama pengumpulan data.

Walaupun kebanyakan variabel memiliki data yang lengkap, dengan ketiadaan data pada variabel ‘C1‘ dan ‘HCDP‘ dapat membatasi cakupan analisis untuk variabel-variabel tertentu.

Secara keseluruahan, dataset ini memiliki ketersediaan data yang cukup dan hanya dibatasi oleh variabel ‘C1‘ dan ‘HCDP‘ yang hilang.

4.1.4 Analisis Distribusi Variabel-variabel Utama

Gambar 2 mengilustrasikan distribusi berbagai variabel utama dalam operasi transmisi gas. Untuk memberikan contoh, variabel ‘PRESSURE‘ menunjukkan distribusi bimodal, dengan satu puncak yang menonjol dengan nilai disekitar 500 dan puncak lainnya dengan nilai sekitar 1500. Contoh lainnya, variabel ‘TEMPERATURE‘ menunjukkan distribusi yang miring ke kanan, dengan sebagian besar nilai berada di sekitar 75 dan 100 derajat, variabel tersebut kemungkinan juga memiliki beberapa *outlier* di sekitar 173 derajat.

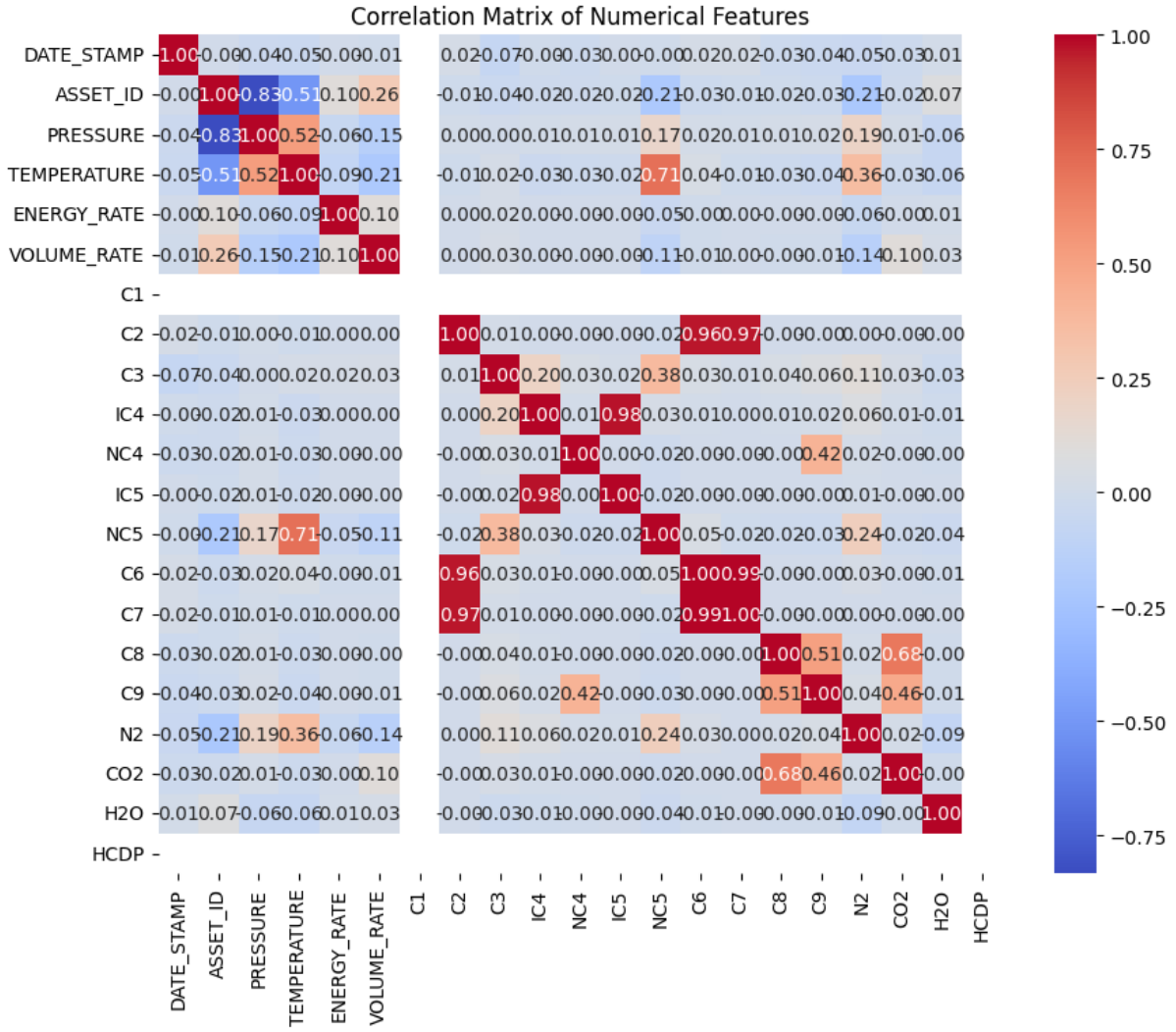


Gambar 2. Distribusi Variabel Utama

4.1.5 *Correlation Matrix*

Gambar **??** mengilustrasikan korelasi dari fitur numerik dalam *dataset*. Matriks ini menyoroti tingkat hubungan antar variabel. Analisis dari matriks tersebut memperlihatkan korelasi negatif sebesar -0.83 dimana dengan meningkatnya 'ASSET_ID' akan cenderung membuat 'PRESSURE' menurun. Hal tersebut dapat mengindikasikan perbedaan kondisi operasi atau karakteristik dari aset dalam jaringan pipa. Adapun korelasi positif seperti 'VOLUME_RATE' dan 'ENERGY_RATE' dengan nilai korelasi sebesar 1.00 yang menunjukkan bahwa kedua variabel ini berkorelasi hampir sempurna.

Secara keseluruhan *Correlation Matrix* dapat menunjukkan pengetahuan lebih lanjut mengenai hubungan antara berbagai variabel dalam operasional transmisi gas di jaringan pipa. Hal tersebut penting dalam pengembangan model prediktif untuk mengidentifikasi area potensial di mana perubahan pada satu variabel dapat memiliki dampak yang signifikan terhadap variabel lainnya.

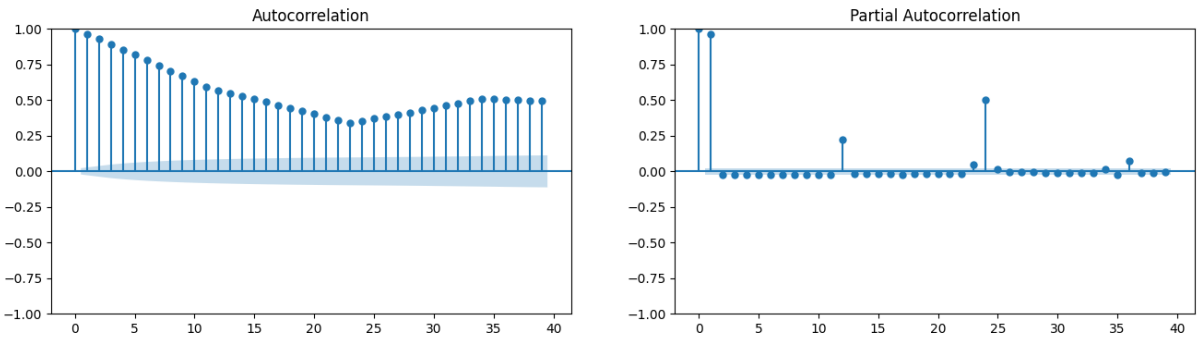


Gambar 3. *Correlation Matrix*

4.2 Hasil Pengujian

Sebelum model dibuat, data akan di filter terlebih dahulu dengan memilih salah satu 'ASSET_ID' dan dalam konteks penelitian ini '133002' dipilih sebagai aset yang akan digunakan. Proses filter tersebut mengurangi jumlah data yang awalnya ada 61313 baris menjadi 8759 baris. Akan dilakukan analisis kepada data yang telah di filter untuk mencari tahu apakah data tersebut sudah stasioner atau belum.

Gambar 4 menunjukkan dua grafik: yang pertama adalah *Autocorrelation* dan yang lainnya adalah *Partial Autocorrelation*. Pada grafik *Autocorrelation* menunjukkan rangkaian yang menurun mulai mendekati satu dan menurun ke nilai positif seiring dengan meningkatnya lag, yang menunjukkan bahwa efek memori dalam data berkurang secara bertahap. Pada grafik *Partial Autocorrelation* terdapat lonjakan yang mencolok pada lag pertama. Fitur tersebut menunjukkan bahwa data *time series* memiliki ketergantungan yang signifikan pada jeda yang pendek.



Gambar 4. Autocorrelation & Partial Autocorrelation

Peneliti melakukan *Augmented Dickey-Fuller Test* pada data yang digunakan dalam penelitian ini untuk mengecek apakah data sudah stasioner atau belum. Hasil dari tes tersebut memberikan *p-value* sebesar 8.931. Hal tersebut menjelaskan bahwa data yang digunakan sudah stasioner karena jika *p-value* dari data melebihi 0.05 kita dapat mengambil kesimpulan bahwa data tersebut adalah data stasioner dan kita tidak perlu melakukan proses *Differencing* pada data sebelum melakukan pemodelan.

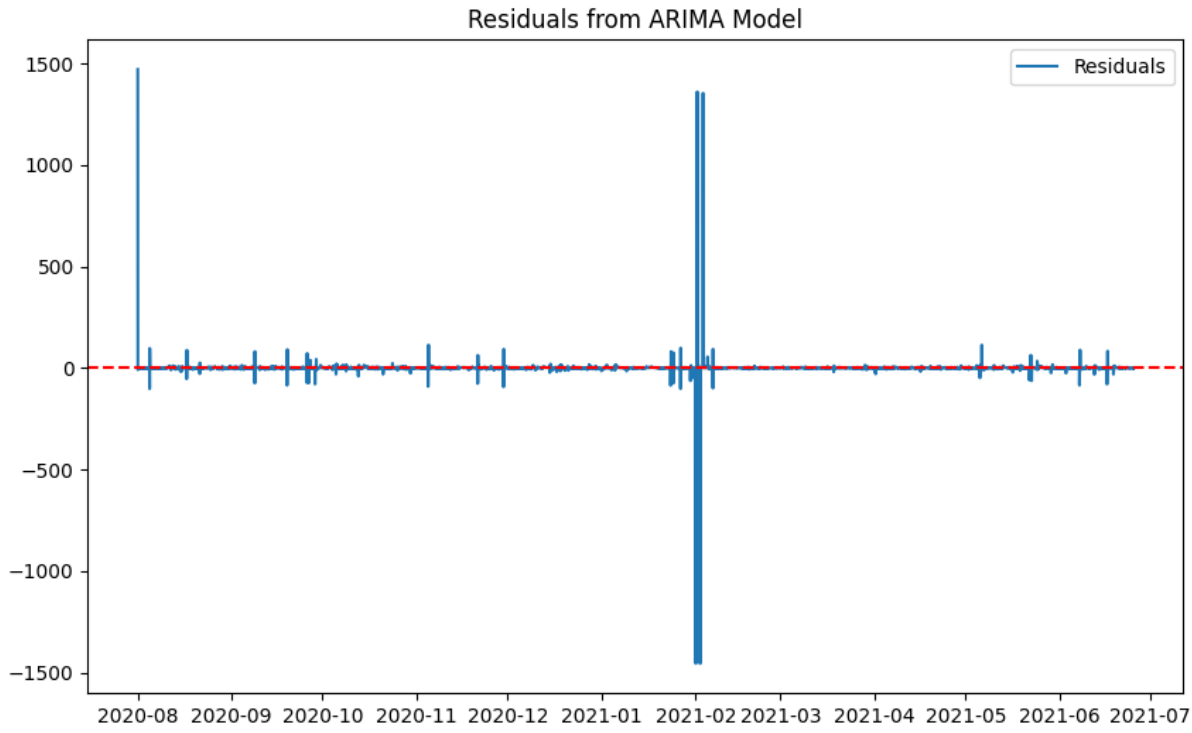
Peneliti melakukan identifikasi model ARIMA terbaik dengan meminimalkan *Akaike Information Criterion* (AIC). Identifikasi ini mencakup pencarian berbagai kombinasi parameter *autoregressive*(p) dan *moving average*(q) dengan asumsi *differencing* (d=1). Setiap konfigurasi model dievaluasi nilai AIC-nya dimana jika model memiliki nilai AIC yang rendah maka model tersebut memiliki kecocokan terhadap data yang lebih baik. Tabel 4 menunjukkan beberapa perbandingan parameter ARIMA.

Model	**AIC**	**Waktu Eksekusi (detik)**
ARIMA(1,1,1)	inf	0.43
ARIMA(1,1,0)	77198.046	0.55
ARIMA(0,1,1)	77198.048	1.29
ARIMA(0,1,0)	77194.203	0.12

Tabel 4. Perbandingan Model

Model ARIMA(1,1,1) memiliki masalah komputasi yang menghasilkan nilai AIC yang tidak dapat didefinisikan, hal tersebut dapat menunjukkan ketidakstablian numerik atau masalah dalam proses estimasi. Model lainnya seperti ARIMA(1,1,0) dan ARIMA(0,1,1) tidak memberikan hasil yang memuaskan karena nilai AIC-nya yang tinggi.

Model yang paling sederhana yaitu ARIMA(0,1,0) memberikan nilai AIC yang paling rendah yaitu 77194.203. Yang menunjukkan bahwa, pada *dataset* ini, model tanpa komponen *autoregressive* atau *moving average* memberikan kecocokan yang paling efisien dibandingkan dengan model lainnya yang dievaluasi.



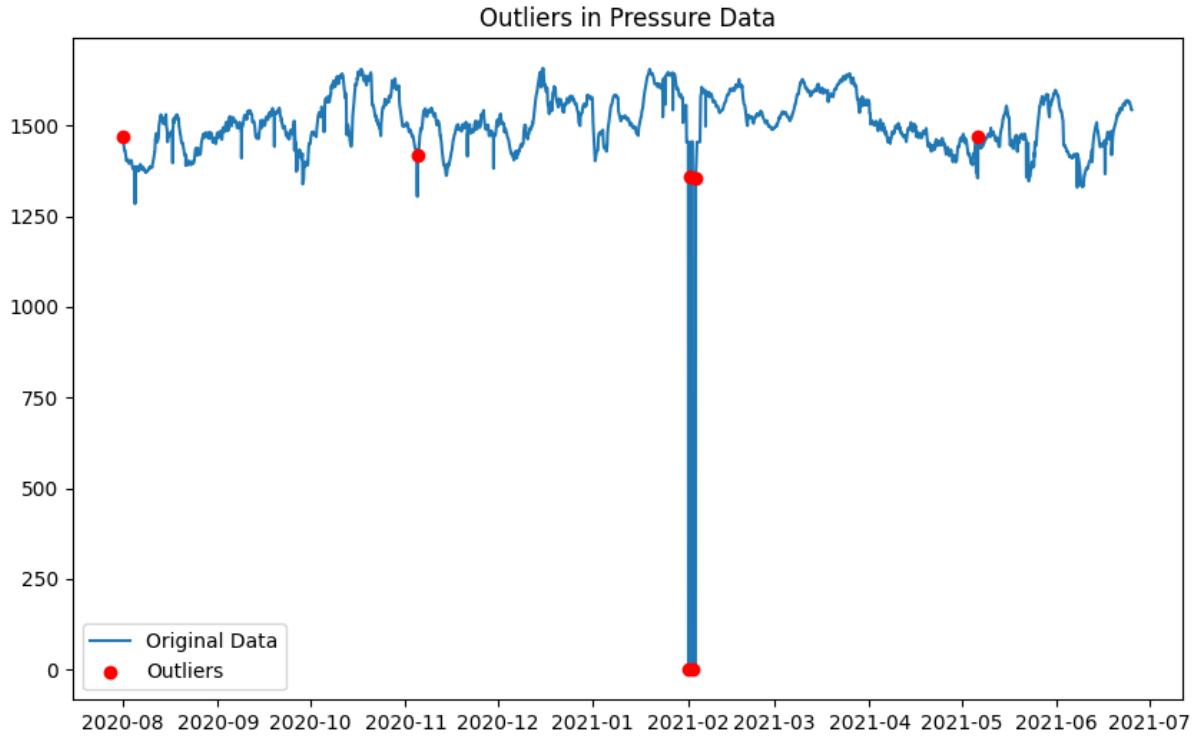
Gambar 5. Residual dari Model ARIMA

Gambar 5 mengilustrasikan *residual* yang terdapat pada model ARIMA yang telah dilatih, dimulai dari bulan Agustus 2020 hingga bulan Juli 2024. Dari ilustrasi tersebut ada beberapa hal yang dapat kita amati:

- Kebanyakan *residual* berkumpul disekitar garis nol yang berarti prediksi dari model mendekati nilai asli.
- Terdapat lonjakan yang menonjol jika dibandingkan kisaran normal dari residual, contohnya seperti di sekitar bulan Februari 2021. Hal ini menunjukkan bahwa model kemungkinan gagal memprediksi nilai aktual secara signifikan yang dapat mengindikasikan potensi masalah dalam spesifikasi model atau adanya titik data anomali dalam periode tersebut.
- Selain dari outlier yang terdeteksi, residual relatif stabil disekitar garis nol untuk sebagian besar datanya yang dapat menunjukkan kecocokan model secara umum.

Residual dari model dapat kita gunakan untuk pendeteksian *Additive Outlier* berdasarkan ambang batas yang didefinisikan sebagai tiga kali standar deviasi residual.

Gambar 6 menunjukkan hasil pendeteksian *Additive Outlier* dengan model ARIMA. Dari total 8759 baris data ada sekitar tujuh *outlier* yang terdeteksi atau sekitar 0.08% dari total data. Dapat kita lihat contoh *outlier* yang terdeteksi di sekitar bulan Februari 2021 yang berada di bawah kisaran nilai umum dari data.



Gambar 6. Deteksi Outlier

Hasil Evaluasi dari model ARIMA(0,1,0) dapat dilihat pada Tabel 5. Tabel tersebut memberikan beberapa nilai metrik evaluasi seperti MSE dan RMSE.

Metriks	Nilai	Persentase
MSE	61428,563	158,041%
RMSE	247,847	0,637%

Tabel 5. Evaluasi Model

Persentase tersebut dihitung berdasarkan nilai metrik evaluasi dibagi dengan varians dari data tes dan ada beberapa hal yang dapat diamati dari hasil evaluasi model ARIMA yang dihasilkan, diantaranya:

- MSE yang dihasilkan lumayan tinggi yang menunjukkan perbedaan substansial antara nilai yang diprediksi dan titik data aktual. Metrik ini lumayan sensitif terhadap *outlier* karena metrik ini mengkuadratkan perbedaan antara nilai yang diprediksi dan nilai yang diamati, sehingga memberikan bobot lebih pada kesalahan yang lebih besar.
- Nilai MSE yang tinggi juga dapat menunjukkan bahwa model yang digunakan kemungkinan berurusan dengan kesalahan prediksi yang tinggi yang kemungkinan disebabkan oleh anomali dalam data, parameter model yang tidak tepat, atau *overfitting* dimana modelnya terlalu sesuai dengan data ketika proses *training*.
- Meskipun RMSE masih menghasilkan kesalahan yang cukup besar per-prediksi, lebih mudah digunakan dibandingkan dengan MSE, karena RMSE memberikan skala yang lebih mudah ditafsirkan dalam unit yang sama dengan data asli sehingga mudah untuk dipahami.

4.3 Analisis Hasil Pengujian

Dalam penelitian ini, didapatkan bahwa model ARIMA(0,1,0) adalah model dengan terbaik karena model tidak memerlukan tambahan komponen *autoregressive* maupun *moving average*, pemilihan model tersebut juga memiliki alasan khusus karena model dengan parameter tersebut memiliki nilai AIC paling kecil dibandingkan dengan model dengan parameter lain. Model tersebut dapat mendeteksi sekitar 0.07% *outlier* yang terdapat dalam sekitar 8759 baris data. Evaluasi model menggunakan metrik *Mean Squared Error* (MSE) dan *Root Mean Squared*

Error (RMSE) menunjukkan hasil yang lumayan unik dikarenakan sifat MSE yang sensitif terhadap *outlier* didalam data yang berbanding terbalik dengan nilai RMSE yang walaupun masih terbilang cukup tinggi namun lebih mudah ditafsirkan karena dalam unit yang sama dengan data asli. Pendeteksian *Additive Outlier* perhitungan tiga standar deviasi dari *residual* model juga menunjukkan hasil yang baik dan dapat mendeteksi nilai ekstrim dalam data monitoring transmisi gas dalam jaringan pipa.

5. Kesimpulan

Penelitian ini bertujuan untuk menerapkan metode ARIMA dan *Additive Outlier* untuk mendeteksi anomali dalam data monitoring operasi transmisi gas dalam jaringan pipa. Dari hasil pengujian dan analisis yang telah dilakukan, dapat disimpulkan bahwa model ARIMA(0,1,0) terpilih sebagai model terbaik untuk mendeteksi anomali pada data tekanan gas. Menurut hasil evaluasi metrik MSE dan RMSE, model tersebut memiliki ruang perbaikan baik dalam penyempurnaan parameter atau mengatasi masalah kualitas data. Secara umum model dapat memprediksi dengan kesalahan yang dapat diterima, namun ada beberapa kesalahan tinggi yang perlu diatasi untuk meningkatkan efektivitas secara keseluruhan.

Berdasarkan hasil dan analisis penelitian ini, ada beberapa saran untuk penelitian masa depan. Pertama, pengembangan model untuk variabel operasional lainnya dapat dilakukan untuk mendapatkan pemahaman lebih lanjut mengenai operasi transmisi gas. Kedua, model ARIMA dengan Additive Outlier dapat diterapkan pada 'ASSET_ID' yang berbeda untuk menguji generalisasi model dan efektivitasnya dalam berbagai kondisi operasional. Terakhir, penelitian lebih lanjut dapat dilakukan untuk membandingkan model ARIMA dengan teknik lain dalam *machine learning* yang mungkin dapat memberikan pengertian lebih lanjut atau bahkan memperbaiki akurasi prediksi.

Dengan mempertimbangkan hasil dan saran yang disampaikan, diharapkan penelitian ini dapat memberikan kontribusi yang signifikan dalam bidang pemantuan operasional transmisi gas dan pendeteksian anomali dalam data *time series*.

Daftar Pustaka

[1] R. Adhikari and R. K. Agrawal. An introductory study on time series modeling and forecasting, 2013.

[2] R. Agustianto, I. Purnamasari, and S. Suyitno. Analisis data ketinggian permukaan air sungai mahakam daerah kutai kartanegara tahun 2010-2016 menggunakan model autoregressive integrated moving average (arima) dengan efek outlier. *EKSPONENSIAL*, 11(1):39–46, 2021.

[3] A. S. Ahmar, S. Guritno, A. Rahman, I. Minggi, M. A. Tiro, M. K. Aidid, S. Annas, D. U. Sutiksno, D. S. Ahmar, K. H. Ahmar, et al. Modeling data containing outliers using arima additive outlier (arima-ao). In *Journal of Physics: Conference Series*, volume 954, page 012010. IOP Publishing, 2018.

[4] S. S. Aljameel, D. M. Alomari, S. Alismail, F. Khawaher, A. A. Alkhudhair, F. Aljubran, and R. M. Alzannan. An anomaly detection model for oil and gas pipelines using machine learning. *Computation*, 10(8):138, 2022.

[5] V. Arumugam and V. Natarajan. Time series modeling and forecasting using autoregressive integrated moving average and seasonal autoregressive integrated moving average models. *Instrumentation, Mesures, Métrologies*, 22(4), 2023.

[6] B. Awuku, Y. Huang, and N. Yodo. Predicting natural gas pipeline failures caused by natural forces: an artificial intelligence classification approach. *Applied Sciences*, 13(7):4322, 2023.

[7] I. Fadliani, I. Purnamasari, and W. Wasono. Peramalan dengan metode sarima pada data inflasi dan identifikasi tipe outlier (studi kasus: Data inflasi indonesia tahun 2008-2014). *Jurnal Statistika Universitas Muhammadiyah Semarang*, 9(2):109–116, 2021.

[8] S. Hamiane, Y. Ghanou, H. Khalifi, and M. Telmem. Comparative analysis of lstm, arima, and hybrid models for forecasting future gdp. *Journal homepage: http://iieta. org/journals/isi*, 29(3):853–861, 2024.

[9] M. R. Kamal and M. A. Setiawan. Deteksi anomali dengan security information and event management (siem) splunk pada jaringan uii. *AUTOMATA*, 2(2), 2021.

[10] J. Kaur, K. S. Parmar, and S. Singh. Autoregressive models in environmental forecasting time series: a theoretical and application review. *Environmental Science and Pollution Research*, 30(8):19617–19641, 2023.

[11] V. Kozitsin, I. Katser, and D. Lakontsev. Online forecasting and anomaly detection based on the arima model. *Applied Sciences*, 11(7):3194, 2021.

[12] A. Q. Munir, F. Nuraini, and Y. Evrita Lusiana Utari. Deteksi anomali data prediksi untuk meningkatkan akurasi hasil peramalan data curah hujan. In *Prosiding Seminar Nasional Multidisiplin Ilmu*, volume 3, pages 73–83, 2021.

[13] P. Purwadi, P. S. Ramadhan, and N. Safitri. Penerapan data mining untuk mengestimasi laju pertumbuhan penduduk menggunakan metode regresi linier berganda pada bps deli serdang. *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer)*, 18(1):55–61, 2019.

[14] E. Purwaningsih and S. Subirman. Alternatif kebijakan perencanaan kebutuhan obat dengan menggunakan metode arima box-jenkins untuk mengatasi kelebihan stok. *Jurnal Kebijakan Kesehatan Indonesia: JKKI*, 8(1):10–17, 2019.

[15] M. Saqib, E. Şentürk, S. A. Sahu, and M. A. Adil. Comparisons of autoregressive integrated moving average (arima) and long short term memory (lstm) network models for ionospheric anomalies detection: a study on haiti (m w= 7.0) earthquake. *Acta Geodaetica et Geophysica*, pages 1–19, 2022.

[16] S. Siami-Namini and A. S. Namin. Forecasting economics and financial time series: Arima vs. lstm, 2018.

[17] S.-H. Tseng and T. Son Nguyen. Agent-based modeling of rumor propagation using expected integrated mean squared error optimal design. *Applied System Innovation*, 3(4):48, 2020.

[18] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun. Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*, 2022.

[19] J. Yoon, D. Jarrett, and M. Van der Schaar. Time-series generative adversarial networks. *Advances in neural information processing systems*, 32, 2019.

Lampiran

Pemuatan Data

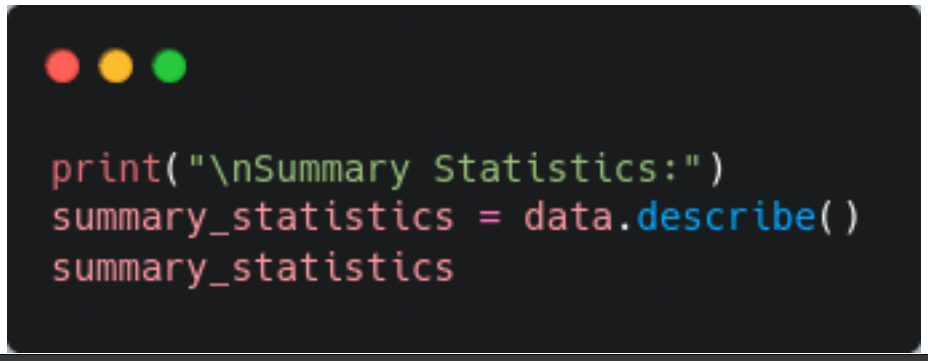
```
path_pattern = '/content/drive/MyDrive/wnts_hourly/2021.csv'

data = pd.read_csv(path_pattern)

data['DATE_STAMP'] = pd.to_datetime(data['DATE_STAMP'], errors='coerce')

data.sort_values(by='DATE_STAMP', inplace=True)

data.reset_index(drop=True, inplace=True)
```

Informasi Dasar Data

```
print("Data Information:")
print(data.info())
```

```
Data Information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 61313 entries, 0 to 61312
Data columns (total 21 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   DATE_STAMP    61313 non-null  datetime64[ns]
 1   ASSET_ID      61313 non-null  int64
 2   PRESSURE      61313 non-null  float64
 3   TEMPERATURE   61313 non-null  float64
 4   ENERGY_RATE   61313 non-null  float64
 5   VOLUME_RATE   61313 non-null  float64
 6   C1            0 non-null      float64
 7   C2            61313 non-null  float64
 8   C3            61313 non-null  float64
 9   IC4           61313 non-null  float64
 10  NC4           61313 non-null  float64
 11  IC5           61313 non-null  float64
 12  NC5           61313 non-null  float64
 13  C6            61313 non-null  float64
 14  C7            61313 non-null  float64
 15  C8            61313 non-null  float64
 16  C9            61313 non-null  float64
 17  N2            61313 non-null  float64
 18  CO2           61313 non-null  float64
 19  H2O           61313 non-null  float64
 20  HCDP          0 non-null      float64
dtypes: datetime64[ns](1), float64(19), int64(1)
memory usage: 9.8 MB
None
```

Summary Statistics



```
print("\nSummary Statistics:")
summary_statistics = data.describe()
summary_statistics
```

Summary Statistics:

	DATE_STAMP	ASSET_ID	PRESSURE	TEMPERATURE	ENERGY_RATE	VOLUME_RATE	C1	C2	C3	IC4	...
count	61313	61313.000000	61313.000000	61313.000000	61313.000000	61313.000000	0.0	61313.000000	61313.000000	61313.000000	...
mean	2021-01-30 12:00:00	133030.142857	1169.378470	89.256917	129.698079	118.024030	NaN	29.351508	1.908598	0.608650	...
min	2020-08-01 01:00:00	133001.000000	-5.273438	0.000000	0.000000	0.000000	NaN	0.000000	0.000000	0.000000	...
25%	2020-10-31 06:00:00	133002.000000	590.366821	77.603668	62.149017	58.170605	NaN	3.826353	1.684784	0.523719	...
50%	2021-01-30 12:00:00	133004.000000	1396.582031	83.859009	84.326347	78.816093	NaN	4.356140	1.940315	0.595148	...
75%	2021-05-01 18:00:00	133070.000000	1502.542480	98.175438	199.270676	184.165253	NaN	5.194573	2.459368	0.710253	...
max	2021-07-31 23:00:00	133071.000000	1676.445801	173.188477	91865.320312	32426.248047	NaN	102885.945312	5.743607	85.753571	...
std	NaN	32.095782	420.865067	16.697954	534.866794	203.815587	NaN	1602.321168	0.865248	1.177340	...

8 rows × 21 columns

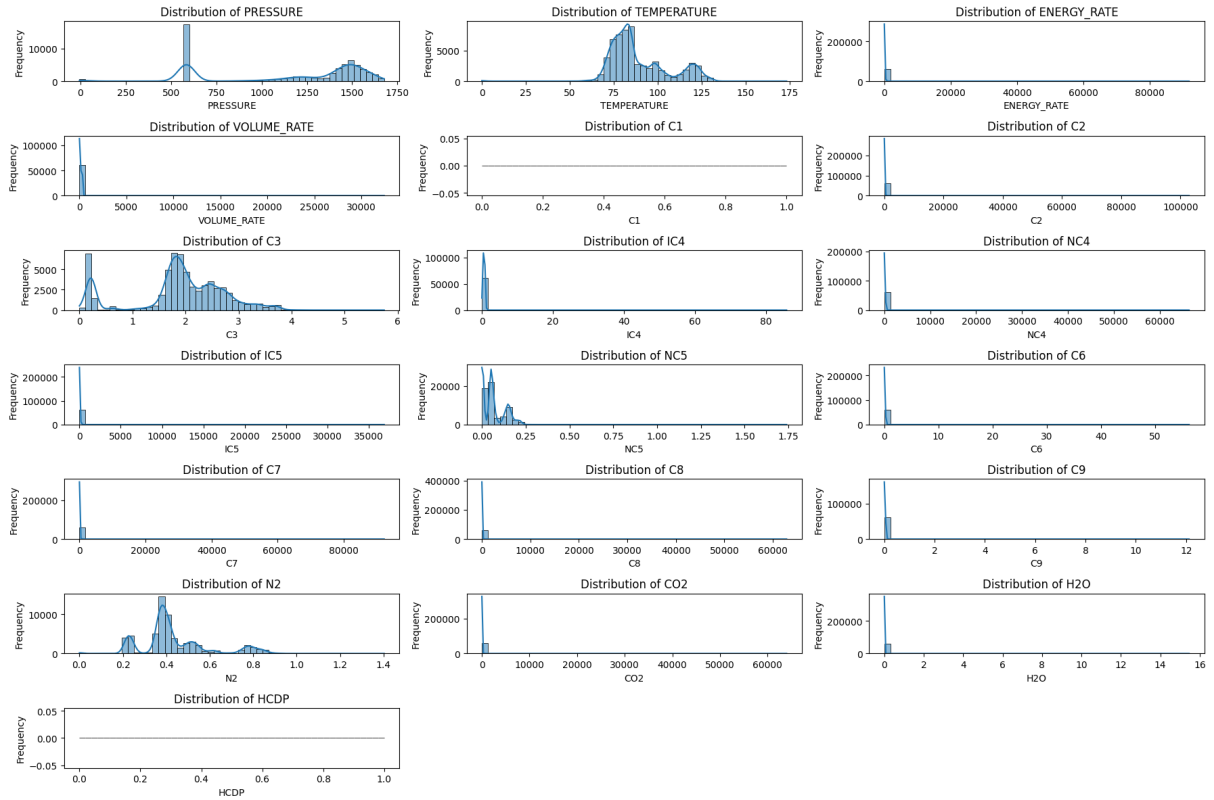
Missing Values

```python
missing_percentage = data.isnull().mean() * 100
print("\nPercentage of Missing Data in Each Column:")
print(missing_percentage)
```

```
Percentage of Missing Data in Each Column:
DATE_STAMP        0.0
ASSET_ID          0.0
PRESSURE          0.0
TEMPERATURE       0.0
ENERGY_RATE       0.0
VOLUME_RATE       0.0
C1              100.0
C2                0.0
C3                0.0
IC4               0.0
NC4               0.0
IC5               0.0
NC5               0.0
C6                0.0
C7                0.0
C8                0.0
C9                0.0
N2                0.0
CO2               0.0
H2O               0.0
HCDP            100.0
dtype: float64
```

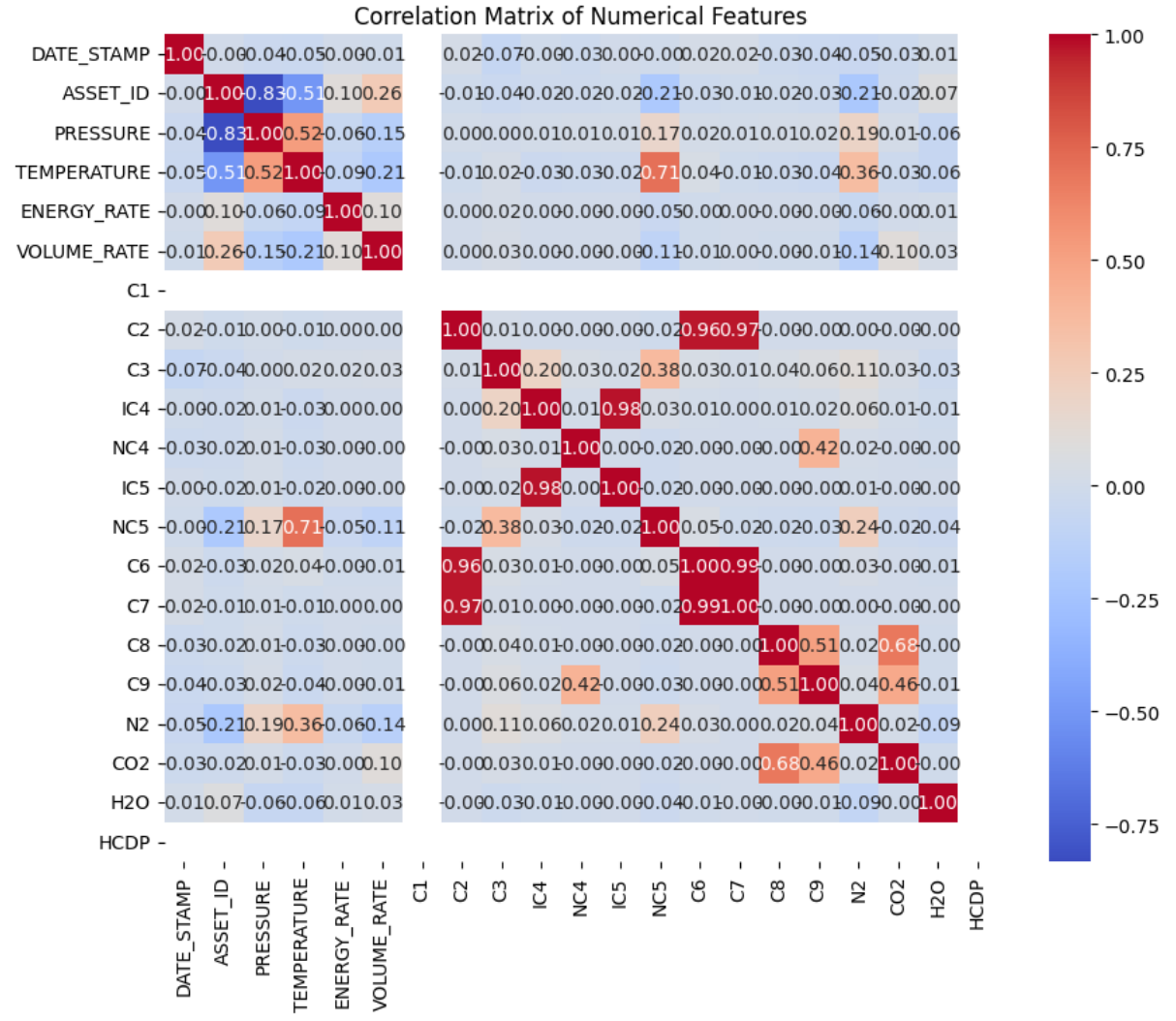
Plot Distribusi

```python
columns_to_plot = [col for col in data.columns if col not in ['DATE_STAMP', 'ASSET_ID']]

n_cols = 3
n_rows = len(columns_to_plot) // n_cols + (len(columns_to_plot) % n_cols > 0)

fig, axes = plt.subplots(n_rows, n_cols, figsize=(18, 12))
axes = axes.flatten()

for i, column in enumerate(columns_to_plot):
    if data[column].dtype != 'object':
        sns.histplot(data[column], bins=50, kde=True, ax=axes[i])
        axes[i].set_title(f'Distribution of {column}')
        axes[i].set_xlabel(column)
        axes[i].set_ylabel('Frequency')
    else:
        axes[i].remove()

for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

plt.tight_layout()
plt.show()
```



Matriks Korelasi

```
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(), annot=True, fmt=".2f", cmap='coolwarm')
plt.title('Correlation Matrix of Numerical Features')
plt.show()
```



Filter Aset

```
asset_id = 133002
asset_data = data[data['ASSET_ID'] == asset_id].copy()
asset_data.set_index('DATE_STAMP', inplace=True)

asset_data.head()
```

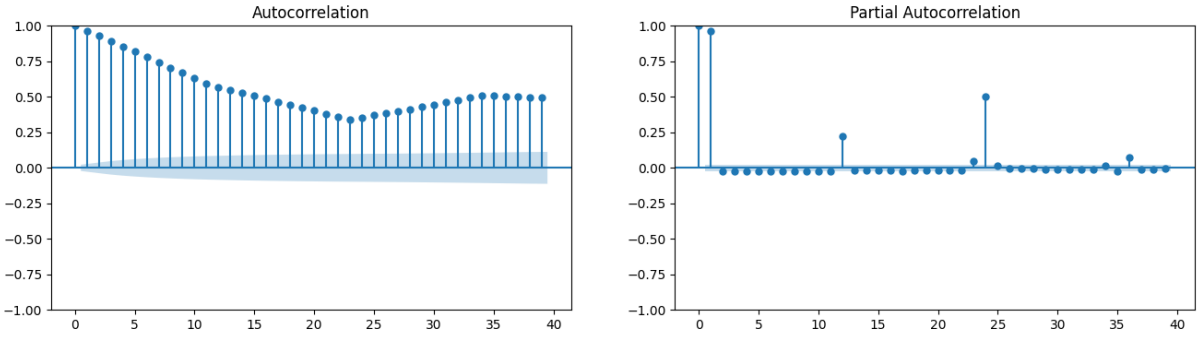
DATE_STAMP	ASSET_ID	PRESSURE	TEMPERATURE	ENERGY_RATE	VOLUME_RATE	C1	C2	C3	IC4	NC4	IC5	NC5	C6	C7	C8	C9	N2	CO2	H2O	HCDP
2020-08-01 01:00:00	133002	1470.804688	113.056580	12.383150	10.769962	NaN	7.302561	2.537969	0.781920	0.650846	0.305333	0.194397	0.176429	0.090155	0.041406	0.010556	0.848276	1.405212	0.0	NaN
2020-08-01 02:00:00	133002	1462.045654	113.168144	12.868464	11.181238	NaN	7.337192	2.559971	0.789768	0.656362	0.310611	0.198807	0.179039	0.091338	0.041786	0.010621	0.851131	1.394804	0.0	NaN
2020-08-01 03:00:00	133002	1457.055908	112.948685	12.755524	11.085944	NaN	7.335319	2.546860	0.788273	0.654451	0.310485	0.197630	0.179674	0.091613	0.043880	0.010595	0.849007	1.391734	0.0	NaN
2020-08-01 04:00:00	133002	1454.505615	112.935165	12.964414	11.272172	NaN	7.333198	2.546898	0.784753	0.652616	0.307330	0.195960	0.177928	0.090078	0.045761	0.010326	0.850651	1.396422	0.0	NaN
2020-08-01 05:00:00	133002	1452.889404	112.747292	12.685232	11.020149	NaN	7.331183	2.550877	0.785164	0.654090	0.307176	0.195879	0.176577	0.089504	0.048847	0.010259	0.849096	1.394010	0.0	NaN

Train Test Split

```
train_size = int(len(asset_data) * 0.9)
train, test = asset_data['PRESSURE'][:train_size], asset_data['PRESSURE'][train_size:]
```

Autocorrelation & Partial Autocorrelation

```
fig, axes = plt.subplots(1, 2, figsize=(16, 4))
plot_acf(train, ax=axes[0])
plot_pacf(train, ax=axes[1])
plt.show()
```



Augmented Dickey-Fuller Test

```
adf_result = adfuller(train)
p_value = adf_result[1]
if p_value > 0.05:
    print("The time series is not stationary. Differencing is required.")
    train = train.diff().dropna()
else:
    print("The time series is stationary. No differencing is required.")

p_value
```

```
The time series is stationary. No differencing is required.
8.9135216687435e-08
```

Mencari Parameter ARIMA Terbaik

```
arima_model = auto_arima(train, start_p=1, start_q=1, start_P=1, start_Q=1,
                         max_p=5, max_q=5, max_P=5, max_Q=5, seasonal=True,
                         stepwise=True, suppress_warnings=True, D=10, max_D=10,
                         error_action='ignore', trace=True)
```

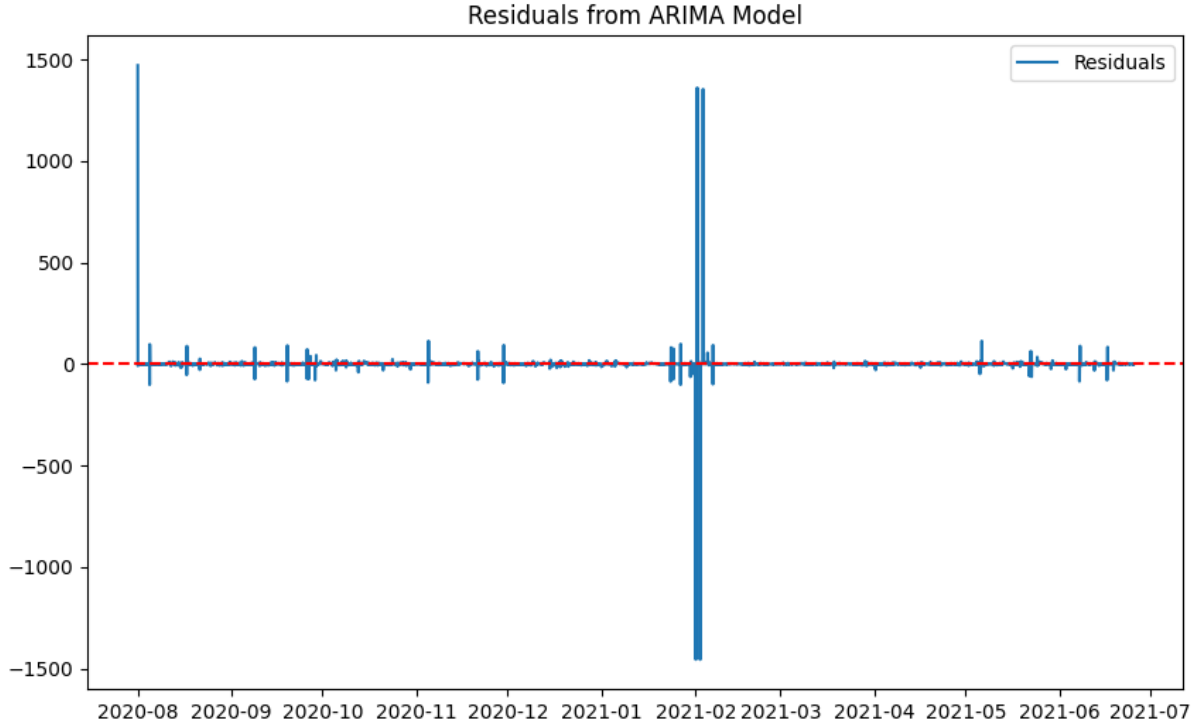
```
Performing stepwise search to minimize aic
 ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=inf, Time=9.94 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=77196.202, Time=0.23 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=77198.046, Time=0.27 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=77198.048, Time=0.70 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=77194.203, Time=0.13 sec

Best model:  ARIMA(0,1,0)(0,0,0)[0]
Total fit time: 11.299 seconds
```

Pelatihan Model ARIMA

```
arima_result = ARIMA(train, order=arima_model.order).fit()
```

Residual Model

```
residuals = arima_result.resid

plt.figure(figsize=(10, 6))
plt.plot(residuals, label='Residuals')
plt.axhline(y=0, color='red', linestyle='--')
plt.title('Residuals from ARIMA Model')
plt.legend()
plt.show()
```



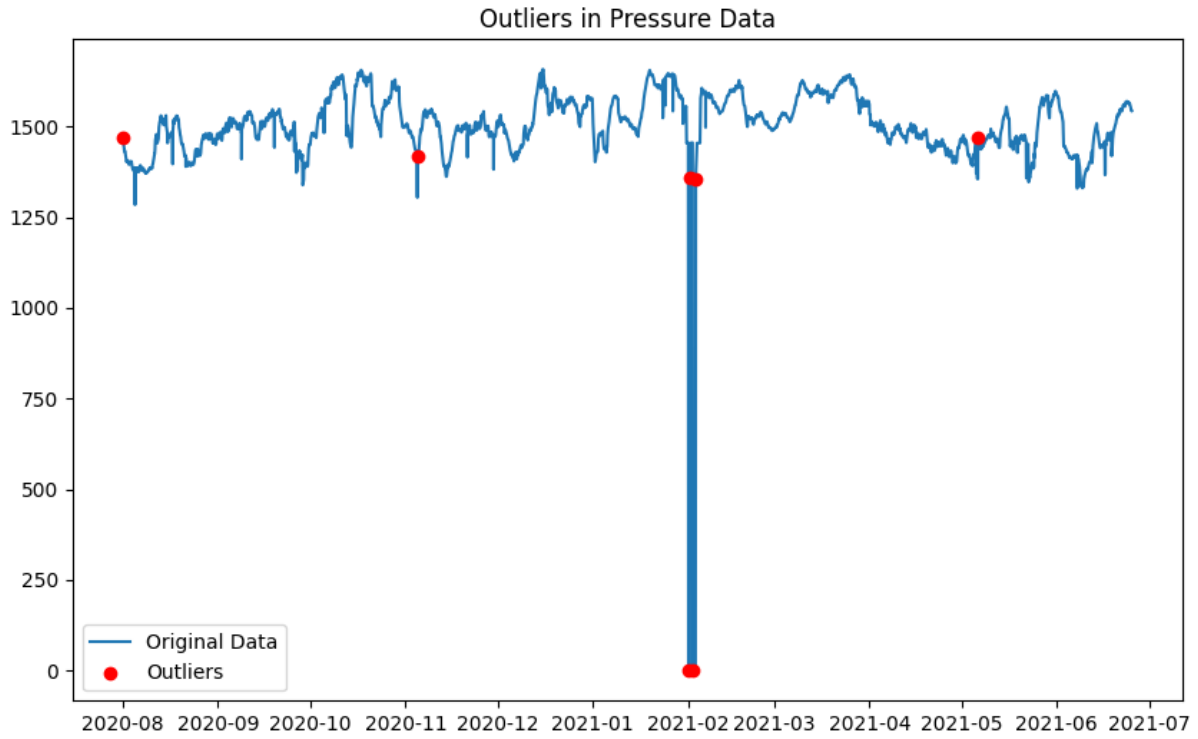
Pendeteksian Outlier

```
threshold = 3 * np.std(residuals)
outliers = residuals[np.abs(residuals) > threshold]
print("Detected Outliers:")
print(outliers)
```

```
Detected Outliers:
DATE_STAMP
2020-08-01 01:00:00    1470.804688
2020-11-04 22:00:00     113.313110
2021-02-01 02:00:00   -1455.991109
2021-02-01 13:00:00    1360.619718
2021-02-02 12:00:00   -1456.320568
2021-02-03 11:00:00    1353.706778
2021-05-06 09:00:00     113.148926
dtype: float64
```

Plot Outlier Yang Terdeteksi

```
plt.figure(figsize=(10, 6))
plt.plot(train, label='Original Data')
plt.scatter(outliers.index, train[outliers.index], color='red', label='Outliers', zorder=5)
plt.title('Outliers in Pressure Data')
plt.legend()
plt.show()
```



Persentase Outlier

```
outlier_percentage = (len(outliers) / len(asset_data)) * 100
print(f"Percentage of Detected Outliers: {outlier_percentage:.2f}%")
```

```
Percentage of Detected Outliers: 0.08%
```

Evaluasi Model

```python
forecast = arima_result.forecast(steps=len(test))

mse = mean_squared_error(test, forecast)
rmse = np.sqrt(mse)
variance = test.var()
mse_percentage_error = (mse / variance) * 100
rmse_percentage_error = (rmse / variance) * 100

print(f"MSE: {mse}")
print(f"RMSE: {rmse}")
print(f"Percentage Error (MSE): {mse_percentage_error}%")
print(f"Percentage Error (RMSE): {rmse_percentage_error}%")
```

```
MSE: 61428.56303315098
RMSE: 247.84786267618082
Percentage Error (MSE): 158.04131180503 33%
Percentage Error (RMSE): 0.637654527654806%
```