

BAB V HASIL DAN PEMBAHASAN

Hasil dan pembahasan dari penelitian ini dipaparkan secara rinci dan jelas. Setiap tahapan hasil dijelaskan dan diuji fungsinya melalui pembahasan. Tujuan utama penelitian diungkapkan pada pembahasan berdasarkan hasil yang diperoleh selama melaksanakan penelitian.

5.1. Hasil

Penelitian ini dilaksanakan berdasarkan dengan rumusan masalah, batasan penelitian, dan tujuan. Selain itu, langkah-langkah penelitian dirancang dengan menyesuaikan kondisi nyata peneliti. Adapun setelah penelitian ini dilaksanakan, maka hasil diuraikan secara nyata berdasarkan data asli yang diperoleh dari situs daring. Hasil diuraikan berdasarkan langkah komputasi yang dirancang peneliti.

5.1.1. Hasil Simulasi Pengolahan Data

Sebelum membuat pemodelan pergerakan harga batubara Newcastle berjangka menggunakan data yang sebenarnya, terlebih dahulu dilaksanakan perancangan pengolahan data menggunakan data simulasi. Data simulasi merupakan contoh data yang hanya akan digunakan untuk percobaan pada penelitian ini dan bukan merupakan data sebenarnya. Hasil yang didapatkan adalah hasil simulasi. Adapun hasil pengolahan data simulasi diuraikan sebagai berikut.

(1) Simulasi Pembersihan Data.

Harga Batubara Newcastle Berjangka Sebelum Pembersihan Data						
	Open	High	Low	Vol.	High (d+1)	Low (d+1)
Date						
2021-01-04	82.25	82.25	81.25	0.13	81.50	80.50
2021-01-05	81.50	81.50	80.50	0.14	80.10	79.50
2021-01-06	80.10	80.10	79.50	0.02	81.00	79.30
2021-01-07	79.30	81.00	79.30	0.05	84.00	82.75
2021-01-08	82.75	84.00	82.75	0.03	84.50	84.50
...
2021-06-24	127.50	127.50	127.50	0.02	127.70	127.70
2021-06-25	127.70	127.70	127.70	0.03	130.50	128.50
2021-06-28	130.50	130.50	128.50	0.23	130.45	130.45
2021-06-29	130.45	130.45	130.45	0.03	132.50	132.50
2021-06-30	132.50	132.50	132.50	0.07	137.75	137.75

[127 rows x 6 columns]

Gambar 5.1 Data Simulasi Harga Sebelum Pembersihan Data

Harga Batubara Newcastle Berjangka Setelah Pembersihan Data						
Date	Open	High	Low	Vol.	High (d+1)	Low (d+1)
2021-01-04	82.25	82.25	81.25	0.13	81.50	80.50
2021-01-05	81.50	81.50	80.50	0.14	80.10	79.50
2021-01-06	80.10	80.10	79.50	0.02	81.00	79.30
2021-01-07	79.30	81.00	79.30	0.05	84.00	82.75
2021-01-08	82.75	84.00	82.75	0.03	84.50	84.50
...
2021-06-24	127.50	127.50	127.50	0.02	127.70	127.70
2021-06-25	127.70	127.70	127.70	0.03	130.50	128.50
2021-06-28	130.50	130.50	128.50	0.23	130.45	130.45
2021-06-29	130.45	130.45	130.45	0.03	132.50	132.50
2021-06-30	132.50	132.50	132.50	0.07	137.75	137.75

[127 rows x 6 columns]

Gambar 5.2 Data Simulasi Harga Setelah Pembersihan Data

Berdasarkan gambar 5.1 dan 5.2 di atas, seluruh baris data diketahui telah terisi penuh. Hal tersebut dapat terlihat pada jumlah baris yang dihasilkan sama yaitu sebanyak 127 baris. Dengan demikian, data simulasi tersebut dapat dilanjutkan ke tahap pengolahan selanjutnya.

(2) Simulasi Normalisasi Data.

Data Normalisasi Historis Harga Batubara Newcastle Berjangka						
	Open	High	Low	Vol.	High (d+1)	Low (d+1)
0	0.055451	0.041031	0.036654	0.565217	0.024284	0.020530
1	0.041353	0.026718	0.022556	0.608696	0.000000	0.003422
2	0.015038	0.000000	0.003759	0.086957	0.015611	0.000000
3	0.000000	0.017176	0.000000	0.217391	0.067650	0.059025
4	0.064850	0.074427	0.064850	0.130435	0.076323	0.088965
..
122	0.906015	0.904580	0.906015	0.086957	0.825672	0.828058
123	0.909774	0.908397	0.909774	0.130435	0.874241	0.841745
124	0.962406	0.961832	0.924812	1.000000	0.873374	0.875107
125	0.961466	0.960878	0.961466	0.130435	0.908933	0.910180
126	1.000000	1.000000	1.000000	0.304348	1.000000	1.000000

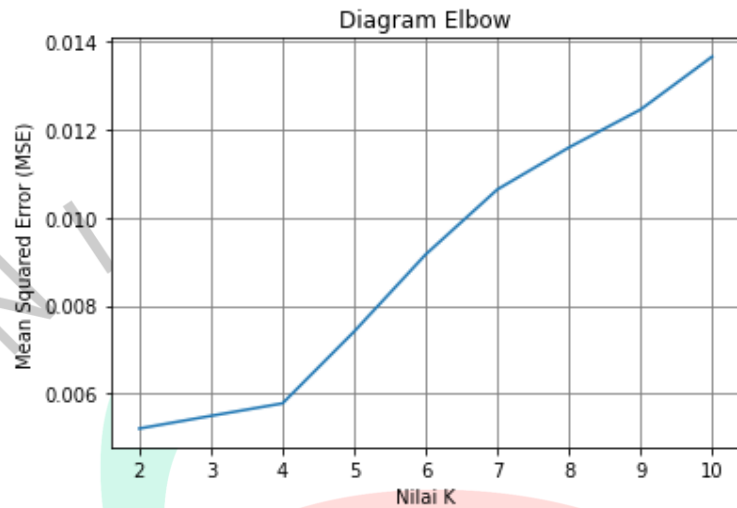
[127 rows x 6 columns]

Gambar 5.3 Data Simulasi Historis Harga Ternormalisasi

Berdasarkan gambar 5.3 di atas, data simulasi telah ternormalisasi menggunakan metode *MinMax scaler* dimana hasil yang dikeluarkan dalam jangkauan nol sampai dengan satu. Data simulasi yang telah dinormalisasi ini akan diproses untuk melakukan simulasi pemodelan menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression*.

(3) Simulasi Pemodelan Algoritma *K-Nearest Neighbors*.

Pemodelan algoritma *K-Nearest Neighbors* dihasilkan berdasarkan jumlah tetangga yang ditentukan. Berdasarkan data simulasi yang digunakan, maka hasil nilai K yang diperoleh tersaji pada gambar berikut.



Gambar 5.4 Hasil Simulasi Diagram Elbow

Berdasarkan gambar 5.4 di atas, maka 2 akan menjadi K pada pemodelan prediksi menggunakan algoritma *K-Nearest Neighbors*. Angka tersebut dipilih karena memiliki skor MSE terendah diantara nilai K yang lainnya pada diagram tersebut.

(4) Simulasi Pemodelan Algoritma *Vector Autoregression*.

Pemodelan menggunakan algoritma *Vector Autoregression* dimulai dengan memeriksa seluruh kolom apakah sudah stasioner atau belum. Adapun berikut cuplikan gambar hasil dari pemeriksaan stasioneritas menggunakan *Augmented Dickey-Fuller*.

```
Augmented Dickey-Fuller Test pada kolom Open
```

```
-----  
Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari  
nilai signifikan (tidak stasioner)
```

```
Signifikan Level ADF = 0.05
```

```
=> Nilai ADF = 0.999 || Tidak dapat menolak hipotesis nol.
```

```
=> Data tidak stasioner
```

Gambar 5.5 Luaran Hasil Simulasi AD-Fuller Tidak Stasioner

Berdasarkan gambar 5.5 di atas, dapat disimpulkan bahwa kolom *Open* tidak stasioner. Oleh karena itu, dilakukannya transformasi data pada setiap kolom yang digunakan.

Augmented Dickey-Fuller Test pada kolom Open

 Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05

=> Nilai ADF = 0.0 || Menolak hipotesis nol.

=> Data sudah stasioner

Gambar 5.6. Luaran Hasil Simulasi AD-Fuller Sudah Stasioner

Berdasarkan gambar 5.6 di atas, dapat disimpulkan bahwa kolom tersebut telah stasioner dan dapat dilanjutkan ke pemilihan urutan model menggunakan *Akaike Information Criterion* (AIC).

Tabel 5.1 Luaran Simulasi Urutan Model

Urutan Model	Nilai AIC
1	-175.74536410438430
2	-176.62037348461055
3	-175.38589888551925
4	-174.33756998424050

Berdasarkan tabel 5.1 di atas, dapat dilihat bahwa urutan model ke-2 memiliki nilai yang terendah pertama kali dikeluarkan, sehingga urutan model tersebut digunakan untuk proses selanjutnya yaitu membuat pemodelan pergerakan harga batubara Newcastle berjangka.

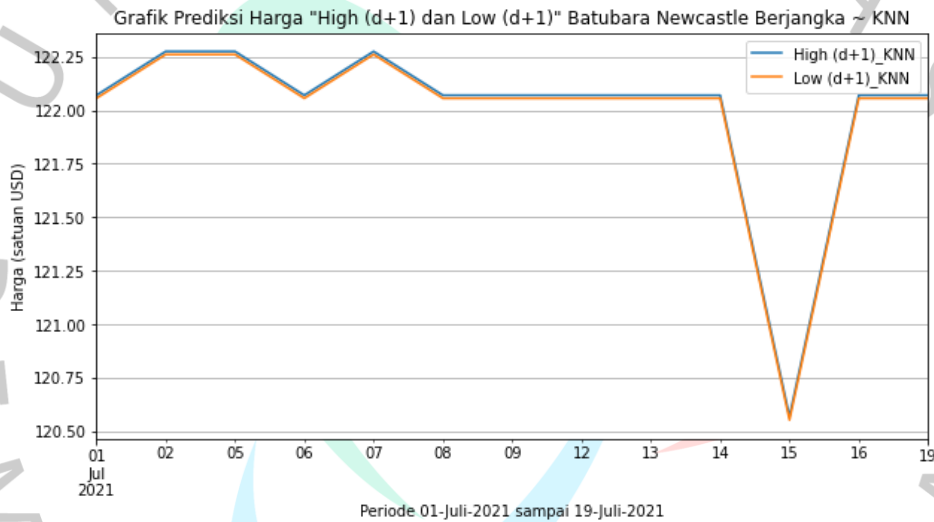
(5) Denormalisasi Hasil Simulasi.

Denormalisasi hasil simulasi dilakukan setelah hasil pemodelan pergerakan harga batubara Newcastle berjangka telah didapatkan. Hasil simulasi disajikan dalam bentuk baris data hasil pemodelan dan ditampilkan dalam bentuk grafik garis supaya dapat memudahkan pembaca dalam melihat pergerakan harga. Adapun hasil denormalisasi pada algoritma *K-Nearest Neighbors* adalah sebagai berikut.

Data Prediksi Harga Batubara Newcastle Berjangka menggunakan Algoritma KNN

Date	High (d+1)_KNN	Low (d+1)_KNN
2021-07-01	122.069991	122.055689
2021-07-02	122.274501	122.260479
2021-07-05	122.274501	122.260479
2021-07-06	122.069991	122.055689
2021-07-07	122.274501	122.260479
2021-07-08	122.069991	122.055689
2021-07-09	122.069991	122.055689
2021-07-12	122.069991	122.055689
2021-07-13	122.069991	122.055689
2021-07-14	122.069991	122.055689
2021-07-15	120.570252	120.553892
2021-07-16	122.069991	122.055689
2021-07-19	122.069991	122.055689

Gambar 5.7 Hasil Simulasi Pemodelan Prediksi Harga (KNN)



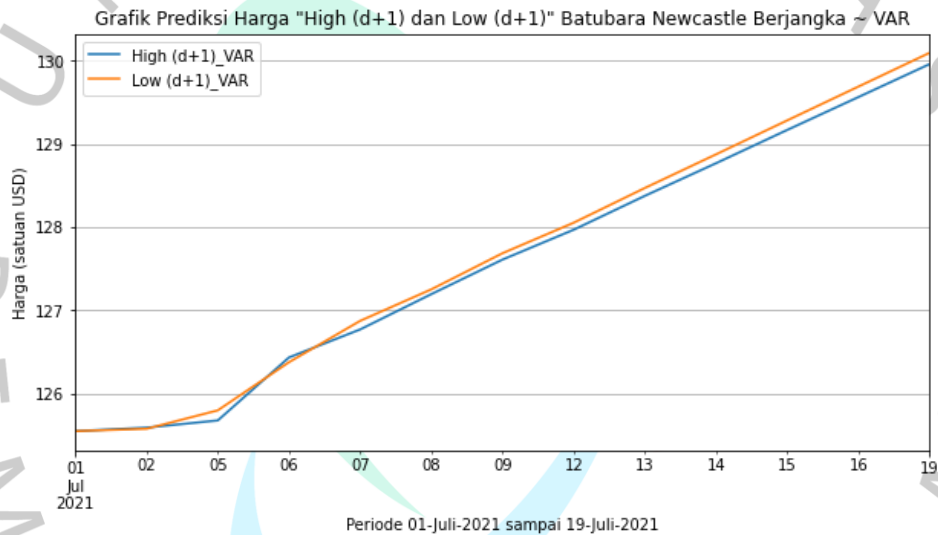
Gambar 5.8 Grafik Hasil Simulasi Pemodelan Prediksi Harga (KNN)

Berdasarkan gambar 5.7 dan 5.8 berupa hasil simulasi pemodelan menggunakan algoritma *K-Nearest Neighbors* yang telah dinormalisasi, dapat diuraikan bahwa harga *High (d+1)* dan *Low (d+1)* selama 13 hari berada di sekitar \$120 sampai \$122. Hal tersebut dapat memberikan arti bahwa dalam 13 hari ke depan, harga batubara masih konstan jika dilihat berdasarkan historis harga yang digunakan pada simulasi ini.

Data Prediksi Harga Batubara Newcastle Berjangka menggunakan Algoritma VAR

Date	High (d+1)_VAR	Low (d+1)_VAR
2021-07-01	125.550000	125.550000
2021-07-02	125.591507	125.577756
2021-07-05	125.677888	125.799104
2021-07-06	126.434820	126.377468
2021-07-07	126.771448	126.874547
2021-07-08	127.195705	127.253288
2021-07-09	127.610810	127.687274
2021-07-12	127.969611	128.053958
2021-07-13	128.376778	128.471294
2021-07-14	128.768787	128.875597
2021-07-15	129.168730	129.283510
2021-07-16	129.563776	129.689674
2021-07-19	129.959872	130.095526

Gambar 5.9 Hasil Simulasi Pemodelan Prediksi Harga (VAR)



Gambar 5.10 Grafik Hasil Simulasi Pemodelan Prediksi Harga (VAR)

Berdasarkan gambar 5.9 dan 5.10 berupa hasil simulasi pemodelan menggunakan algoritma *Vector Autoregression* yang telah dinormalisasi, dapat diuraikan bahwa harga *High (d+1)* dan *Low (d+1)* selama 13 hari mengalami kenaikan mulai dari \$125 hingga hampir menembus \$130. Dengan demikian berdasarkan model yang dihasilkan menggunakan algoritma tersebut memberikan hasil prediksi yang meningkat berdasarkan historis harga yang digunakan pada simulasi ini.

5.1.2. Persiapan Data

Pada proses pemodelan menggunakan algoritma *machine learning* menggunakan data yang sebenarnya, dilakukan persiapan data terlebih dahulu. Persiapan pertama adalah melakukan pembersihan data yang bernilai kosong. Pada pembersihan data ini, sistem akan menghapus baris data yang bernilai kosong. Adapun data sebelum dan sesudah dilakukannya pembersihan data tersaji pada tangkapan layar di bawah ini.

Harga Batubara Newcastle Berjangka Sebelum Pembersihan Data

Date	Open	High	Low	Vol	High (d+1)	Low (d+1)
2017-07-03	79.45	79.45	79.45	0.07	81.15	81.15
2017-07-04	81.15	81.15	81.15	0.14	83.05	83.05
2017-07-05	83.05	83.05	83.05	0.18	83.20	83.20
2017-07-06	83.20	83.20	83.20	0.04	82.55	82.55
2017-07-07	82.55	82.55	82.55	0.01	82.35	82.35
...
2021-06-24	127.50	127.50	127.50	0.02	127.70	127.70
2021-06-25	127.70	127.70	127.70	0.03	130.50	128.50
2021-06-28	130.50	130.50	128.50	0.23	130.45	130.45
2021-06-29	130.45	130.45	130.45	0.03	132.50	132.50
2021-06-30	132.50	132.50	132.50	0.07	137.75	137.75

[1031 rows x 6 columns]

Gambar 5.11 Harga Batubara Newcastle Berjangka Sebelum Pembersihan Data

Harga Batubara Newcastle Berjangka Setelah Pembersihan Data

Date	Open	High	Low	Vol	High (d+1)	Low (d+1)
2017-07-03	79.45	79.45	79.45	0.07	81.15	81.15
2017-07-04	81.15	81.15	81.15	0.14	83.05	83.05
2017-07-05	83.05	83.05	83.05	0.18	83.20	83.20
2017-07-06	83.20	83.20	83.20	0.04	82.55	82.55
2017-07-07	82.55	82.55	82.55	0.01	82.35	82.35
...
2021-06-24	127.50	127.50	127.50	0.02	127.70	127.70
2021-06-25	127.70	127.70	127.70	0.03	130.50	128.50
2021-06-28	130.50	130.50	128.50	0.23	130.45	130.45
2021-06-29	130.45	130.45	130.45	0.03	132.50	132.50
2021-06-30	132.50	132.50	132.50	0.07	137.75	137.75

[1031 rows x 6 columns]

Gambar 5.12 Harga Batubara Newcastle Berjangka Setelah Pembersihan Data

Berdasarkan gambar 5.11 dan 5.12 di atas, dapat diketahui bahwa seluruh baris pada setiap kolom data historis yang digunakan terisi penuh. Hal tersebut dapat dilihat dari jumlah baris yang dikeluarkan memiliki nilai yang sama. Persiapan selanjutnya adalah melakukan normalisasi data pada seluruh baris data. Data historis batubara Newcastle berjangka dibagi menjadi dua bagian yaitu data latih dan uji. Data dibagi dengan rasio 90:10 dimana 90% merupakan data latih dan 10% adalah data uji. Rasio tersebut didapatkan berdasarkan pada batasan penelitian yang telah ditentukan pada penelitian ini.

Tabel 5.2 Pembagian Data Penelitian

Nama Data	Total Baris Data	Rasio	Jumlah Baris Data
Data Latih	1031	90	927
Data Uji		10	104

Berdasarkan tabel 5.2 di atas, maka data uji memiliki 104 baris data. Jumlah baris data yang digunakan akan menjadi batasan total prediksi yang dikeluarkan. Rasio tersebut akan digunakan pada algoritma *K-Nearest Neighbors* dan *Vector Autoregression* untuk melaksanakan keseluruhan dari penelitian ini. Total baris data akan tercantum dalam satuan tanggal setiap hari kerja (Senin sampai Jumat). Adapun data sebelum dinormalisasi yang digunakan pada pemodelan pergerakan harga batubara Newcastle berjangka sebagai data latih disajikan dalam bentuk tangkapan layar berupa gambar dan data uji disajikan melalui tabel berikut ini.

	Open	High	Low	Vol	High (d+1)	Low (d+1)
0	79.45	79.45	79.45	0.07	81.15	81.15
1	81.15	81.15	81.15	0.14	83.05	83.05
2	83.05	83.05	83.05	0.18	83.20	83.20
3	83.20	83.20	83.20	0.04	82.55	82.55
4	82.55	82.55	82.55	0.01	82.35	82.35
..
922	86.30	86.30	86.30	0.00	86.20	86.20
923	86.20	86.20	86.20	0.00	89.25	88.00
924	89.25	89.25	88.00	0.08	85.30	85.00
925	85.30	85.30	85.00	0.16	85.10	83.50
926	85.10	85.10	83.50	0.09	84.25	84.25

[927 rows x 6 columns]

Gambar 5.13 Data Latih Penelitian

Berdasarkan gambar 5.13 di atas, disajikan data latih yang akan digunakan untuk membuat suatu pemodelan pergerakan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression*. Data tersebut disajikan dalam satuan USD dan pada laporan ini hanya disajikan beberapa baris data dikarenakan terdapat 927 baris data latih yang digunakan.

Tabel 5.3 Data Uji Penelitian

No	Open	High	Low	Vol	High (d+1)	Low (d+1)
1	84.25	84.25	84.25	0.03	86.65	86.65
2	86.65	86.65	86.65	0.04	87.5	87.5
3	87.5	87.5	87.5	0	87	87
4	87	87	87	0	87.6	87.6
5	87.6	87.6	87.6	0.01	87.4	87.4
6	87.4	87.4	87.4	0.13	87.45	87.45
7	87.45	87.45	87.45	0	87	87
8	87	87	87	0	86.5	86.5
9	86.5	86.5	86.5	0	86	86
10	86	86	86	0.02	85.75	85.75
11	85.75	85.75	85.75	0.01	86	86
12	86	86	86	0.01	85.75	85.75
13	85.75	85.75	85.75	0	85.5	85.5
14	85.5	85.5	85.5	0.01	85.5	85.5
15	85.5	85.5	85.5	0.01	85.5	85.5
16	85.5	85.5	85.5	0.02	85.1	85.1
17	85.1	85.1	85.1	0.03	86	84.5
18	84.5	86	84.5	0.05	87.5	87.3
19	87.5	87.5	87.3	0.08	88	88
20	88	88	88	0	88	87.5
21	88	88	87.5	0.1	88	87.5
22	88	88	87.5	0	88	87.5
23	87.5	88	87.5	0	89	87.3
24	87.3	89	87.3	0.03	89	89
25	89	89	89	0.01	89	89
26	89	89	89	0	90	90
27	90	90	90	0.03	90.5	90.5
28	90.5	90.5	90.5	0.01	90.5	90.5
29	90.5	90.5	90.5	0.03	90.75	90.75
30	90.75	90.75	90.75	0.02	91	91
31	91	91	91	0	93	91.5
32	91.5	93	91.5	0.02	94	94
33	94	94	94	0.01	95	95
34	95	95	95	0	94.5	94.5
35	94.5	94.5	94.5	0.01	94.5	94.5
36	94.5	94.5	94.5	0.01	93.95	93.95
37	93.95	93.95	93.95	0	95.25	93.25
38	95.25	95.25	93.25	0.09	94	93.8
39	94	94	93.8	0.07	96.25	95.75
40	95.75	96.25	95.75	0.05	97.25	96.95
41	97	97.25	96.95	0.07	96.5	96.5
42	96.5	96.5	96.5	0	94.85	94.85
43	94.85	94.85	94.85	0.08	94.6	94.6
44	94.6	94.6	94.6	0.08	94	94
45	94	94	94	0.09	93.5	93

46	93.5	93.5	93	0.15	93.3	93.3
47	93.3	93.3	93.3	0.04	93.5	93.5
48	93.5	93.5	93.5	0.09	93.4	93.4
49	93.4	93.4	93.4	0	93.5	93.5
50	93.5	93.5	93.5	0.01	94.35	93.5
51	93.5	94.35	93.5	0.06	94.35	94.35
52	94.35	94.35	94.35	0.02	94	94
53	94	94	94	0.01	93.75	93.75
54	93.75	93.75	93.75	0.03	93.95	93.95
55	93.95	93.95	93.95	0	93	93
56	93	93	93	0	93.1	93.1
57	93.1	93.1	93.1	0.01	93.1	93.1
58	93.1	93.1	93.1	0	92.9	92.9
59	92.9	92.9	92.9	0	92.9	92.9
60	92.9	92.9	92.9	0.01	93.3	93.3
61	93.3	93.3	93.3	0.01	92.8	92.8
62	92.8	92.8	92.8	0	93	93
63	93	93	93	0.02	93.8	93.8
64	93.8	93.8	93.8	0.03	95.25	95.15
65	95.15	95.25	95.15	0	96.15	96.15
66	96.15	96.15	96.15	0.01	97.65	97.4
67	97.4	97.65	97.4	0.01	98.7	98.7
68	98.7	98.7	98.7	0.02	100	100
69	100	100	100	0.03	101	101
70	101	101	101	0.02	99	99
71	99	99	99	0	98.35	98.35
72	98.35	98.35	98.35	0.02	99.5	99.2
73	99.5	99.5	99.2	0.02	99.1	99.1
74	99.1	99.1	99.1	0.01	99.75	99.75
75	99.75	99.75	99.75	0.02	101	101
76	101	101	101	0.01	103	101
77	101	103	101	0.01	105	105
78	105	105	105	0.01	105	105
79	105	105	105	0	105.5	105.5
80	105.5	105.5	105.5	0	106	106
81	106	106	106	0.01	118.9	118.9
82	118.9	118.9	118.9	0	119	119
83	119	119	119	0	121.25	120
84	121.25	121.25	120	0.01	119	119
85	119	119	119	0.01	119	119
86	119	119	119	0.01	121.6	121.6
87	121.6	121.6	121.6	0	122.75	122.75
88	122.75	122.75	122.75	0.02	123.7	123.7
89	123.7	123.7	123.7	0.01	126	126
90	126	126	126	0.01	127	127
91	127	127	127	0.01	125.55	125.55
92	125.55	125.55	125.55	0.01	125	125
93	125	125	125	0	125.25	125.25
94	125.25	125.25	125.25	0.01	126	126
95	126	126	126	0.01	125	125
96	125	125	125	0	125.5	125.5
97	125.5	125.5	125.5	0.01	127.25	127.25
98	127.25	127.25	127.25	0	127.5	127.5
99	127.5	127.5	127.5	0.01	127.5	127.5
100	127.5	127.5	127.5	0.02	127.7	127.7
101	127.7	127.7	127.7	0.03	130.5	128.5
102	130.5	130.5	128.5	0.23	130.45	130.45
103	130.45	130.45	130.45	0.03	132.5	132.5

104	132.5	132.5	132.5	0.07	137.75	137.75
-----	-------	-------	-------	------	--------	--------

Berdasarkan tabel 5.3 di atas, disajikan data uji sebanyak 104 baris data yang akan digunakan untuk membuat suatu pemodelan pergerakan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression*. Data tersebut disajikan dalam satuan USD.

Pada penelitian ini, metode *MinMax scaler* digunakan sebagai fungsi yang dapat melakukan normalisasi data. Hasil dari normalisasi data adalah berupa angka pada jangkauan nol sampai satu dimana nol berarti memiliki nilai terendah dan satu merupakan nilai tertinggi dari seluruh baris pada suatu kolom. Data yang dinormalisasi pada penelitian ini adalah data latih dan uji yang akan digunakan untuk proses pemodelan prediksi harga batubara Newcastle berjangka. Adapun hasil normalisasi yang dilakukan berdasarkan data latih tersaji pada tangkapan layar berupa gambar dan data uji diuraikan pada tabel berikut ini.

Data Latih Normalisasi

	Open	High	Low	Vol	High (d+1)	Low (d+1)
0	0.369203	0.368452	0.369203	0.259259	0.388690	0.389417
1	0.389417	0.388690	0.389417	0.518519	0.411310	0.412010
2	0.412010	0.411310	0.412010	0.666667	0.413095	0.413793
3	0.413793	0.413095	0.413793	0.148148	0.405357	0.406064
4	0.406064	0.405357	0.406064	0.037037	0.402976	0.403686
...
922	0.450654	0.450000	0.450654	0.000000	0.448810	0.449465
923	0.449465	0.448810	0.449465	0.000000	0.485119	0.470868
924	0.485731	0.485119	0.470868	0.296296	0.438095	0.435196
925	0.438763	0.438095	0.435196	0.592593	0.435714	0.417360
926	0.436385	0.435714	0.417360	0.333333	0.425595	0.426278

[927 rows x 6 columns]

Gambar 5.14 Hasil Normalisasi Data Latih

Berdasarkan gambar 5.14 di atas, disajikan data latih yang telah dinormalisasi menggunakan metode *MinMax scaler*. Data latih tersebut dinormalisasi sesuai dengan standar normalisasi yang ditetapkan pada rumus *MinMax scaler* dimana akan dihasilkan angka pada jangkauan 0 sampai 1. Hasil normalisasi data latih tersebut hanya disajikan sebagian dikarenakan terdapat 927 baris data yang dihasilkan.

Tabel 5.4 Hasil Normalisasi Data Uji

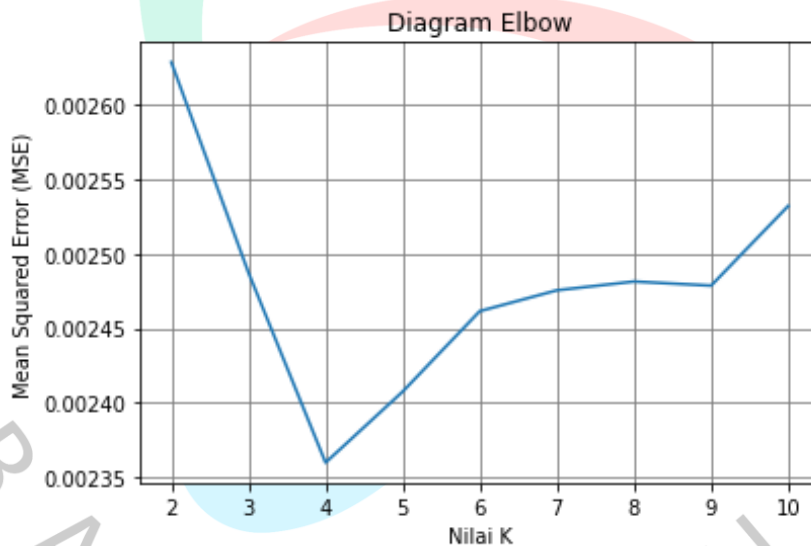
No	Open	High	Low	Vol	High (d+1)	Low (d+1)
1	0.426278	0.425595	0.426278	0.111111	0.454167	0.454816
2	0.454816	0.454167	0.454816	0.148148	0.464286	0.464923
3	0.464923	0.464286	0.464923	0.000000	0.458333	0.458977
4	0.458977	0.458333	0.458977	0.000000	0.465476	0.466112
5	0.466112	0.465476	0.466112	0.037037	0.463095	0.463734
6	0.463734	0.463095	0.463734	0.481481	0.463690	0.464328
7	0.464328	0.463690	0.464328	0.000000	0.458333	0.458977
8	0.458977	0.458333	0.458977	0.000000	0.452381	0.453032
9	0.453032	0.452381	0.453032	0.000000	0.446429	0.447087
10	0.447087	0.446429	0.447087	0.074074	0.443452	0.444114
11	0.444114	0.443452	0.444114	0.037037	0.446429	0.447087
12	0.447087	0.446429	0.447087	0.037037	0.443452	0.444114
13	0.444114	0.443452	0.444114	0.000000	0.440476	0.441141
14	0.441141	0.440476	0.441141	0.037037	0.440476	0.441141
15	0.441141	0.440476	0.441141	0.037037	0.440476	0.441141
16	0.441141	0.440476	0.441141	0.074074	0.435714	0.436385
17	0.436385	0.435714	0.436385	0.111111	0.446429	0.429251
18	0.429251	0.446429	0.429251	0.185185	0.464286	0.462545
19	0.464923	0.464286	0.462545	0.296296	0.470238	0.470868
20	0.470868	0.470238	0.470868	0.000000	0.470238	0.464923
21	0.470868	0.470238	0.464923	0.370370	0.470238	0.464923
22	0.470868	0.470238	0.464923	0.000000	0.470238	0.464923
23	0.464923	0.470238	0.464923	0.000000	0.482143	0.462545
24	0.462545	0.482143	0.462545	0.111111	0.482143	0.482759
25	0.482759	0.482143	0.482759	0.037037	0.482143	0.482759
26	0.482759	0.482143	0.482759	0.000000	0.494048	0.494649
27	0.494649	0.494048	0.494649	0.111111	0.500000	0.500595
28	0.500595	0.500000	0.500595	0.037037	0.500000	0.500595
29	0.500595	0.500000	0.500595	0.111111	0.502976	0.503567
30	0.503567	0.502976	0.503567	0.074074	0.505952	0.506540
31	0.506540	0.505952	0.506540	0.000000	0.529762	0.512485
32	0.512485	0.529762	0.512485	0.074074	0.541667	0.542212
33	0.542212	0.541667	0.542212	0.037037	0.553571	0.554102
34	0.554102	0.553571	0.554102	0.000000	0.547619	0.548157
35	0.548157	0.547619	0.548157	0.037037	0.547619	0.548157
36	0.548157	0.547619	0.548157	0.037037	0.541071	0.541617
37	0.541617	0.541071	0.541617	0.000000	0.556548	0.533294
38	0.557075	0.556548	0.533294	0.333333	0.541667	0.539834
39	0.542212	0.541667	0.539834	0.259259	0.568452	0.563020
40	0.563020	0.568452	0.563020	0.185185	0.580357	0.577289
41	0.577883	0.580357	0.577289	0.259259	0.571429	0.571938
42	0.571938	0.571429	0.571938	0.000000	0.551786	0.552319
43	0.552319	0.551786	0.552319	0.296296	0.548810	0.549346
44	0.549346	0.548810	0.549346	0.296296	0.541667	0.542212
45	0.542212	0.541667	0.542212	0.333333	0.535714	0.530321
46	0.536266	0.535714	0.530321	0.555556	0.533333	0.533888
47	0.533888	0.533333	0.533888	0.148148	0.535714	0.536266
48	0.536266	0.535714	0.536266	0.333333	0.534524	0.535077
49	0.535077	0.534524	0.535077	0.000000	0.535714	0.536266
50	0.536266	0.535714	0.536266	0.037037	0.545833	0.536266
51	0.536266	0.545833	0.536266	0.222222	0.545833	0.546373
52	0.546373	0.545833	0.546373	0.074074	0.541667	0.542212
53	0.542212	0.541667	0.542212	0.037037	0.538690	0.539239
54	0.539239	0.538690	0.539239	0.111111	0.541071	0.541617
55	0.541617	0.541071	0.541617	0.000000	0.529762	0.530321
56	0.530321	0.529762	0.530321	0.000000	0.530952	0.531510

57	0.531510	0.530952	0.531510	0.037037	0.530952	0.531510
58	0.531510	0.530952	0.531510	0.000000	0.528571	0.529132
59	0.529132	0.528571	0.529132	0.000000	0.528571	0.529132
60	0.529132	0.528571	0.529132	0.037037	0.533333	0.533888
61	0.533888	0.533333	0.533888	0.037037	0.527381	0.527943
62	0.527943	0.527381	0.527943	0.000000	0.529762	0.530321
63	0.530321	0.529762	0.530321	0.074074	0.539286	0.539834
64	0.539834	0.539286	0.539834	0.111111	0.556548	0.555886
65	0.555886	0.556548	0.555886	0.000000	0.567262	0.567776
66	0.567776	0.567262	0.567776	0.037037	0.585119	0.582640
67	0.582640	0.585119	0.582640	0.037037	0.597619	0.598098
68	0.598098	0.597619	0.598098	0.074074	0.613095	0.613555
69	0.613555	0.613095	0.613555	0.111111	0.625000	0.625446
70	0.625446	0.625000	0.625446	0.074074	0.601190	0.601665
71	0.601665	0.601190	0.601665	0.000000	0.593452	0.593936
72	0.593936	0.593452	0.593936	0.074074	0.607143	0.604043
73	0.607143	0.607143	0.604043	0.074074	0.602381	0.602854
74	0.602854	0.602381	0.602854	0.037037	0.610119	0.610583
75	0.610583	0.610119	0.610583	0.074074	0.625000	0.625446
76	0.625446	0.625000	0.625446	0.037037	0.648810	0.625446
77	0.625446	0.648810	0.625446	0.037037	0.672619	0.673008
78	0.673008	0.672619	0.673008	0.037037	0.672619	0.673008
79	0.673008	0.672619	0.673008	0.000000	0.678571	0.678954
80	0.678954	0.678571	0.678954	0.000000	0.684524	0.684899
81	0.684899	0.684524	0.684899	0.037037	0.838095	0.838288
82	0.838288	0.838095	0.838288	0.000000	0.839286	0.839477
83	0.839477	0.839286	0.839477	0.000000	0.866071	0.851367
84	0.866231	0.866071	0.851367	0.037037	0.839286	0.839477
85	0.839477	0.839286	0.839477	0.037037	0.839286	0.839477
86	0.839477	0.839286	0.839477	0.037037	0.870238	0.870392
87	0.870392	0.870238	0.870392	0.000000	0.883929	0.884067
88	0.884067	0.883929	0.884067	0.074074	0.895238	0.895363
89	0.895363	0.895238	0.895363	0.037037	0.922619	0.922711
90	0.922711	0.922619	0.922711	0.037037	0.934524	0.934602
91	0.934602	0.934524	0.934602	0.037037	0.917262	0.917360
92	0.917360	0.917262	0.917360	0.037037	0.910714	0.910820
93	0.910820	0.910714	0.910820	0.000000	0.913690	0.913793
94	0.913793	0.913690	0.913793	0.037037	0.922619	0.922711
95	0.922711	0.922619	0.922711	0.037037	0.910714	0.910820
96	0.910820	0.910714	0.910820	0.000000	0.916667	0.916766
97	0.916766	0.916667	0.916766	0.037037	0.937500	0.937574
98	0.937574	0.937500	0.937574	0.000000	0.940476	0.940547
99	0.940547	0.940476	0.940547	0.037037	0.940476	0.940547
100	0.940547	0.940476	0.940547	0.074074	0.942857	0.942925
101	0.942925	0.942857	0.942925	0.111111	0.976190	0.952438
102	0.976219	0.976190	0.952438	0.851852	0.975595	0.975624
103	0.975624	0.975595	0.975624	0.111111	1.000000	1.000000
104	1.000000	1.000000	1.000000	0.259259	0.454167	0.454816

Berdasarkan tabel 5.4 di atas, disajikan data uji yang telah dinormalisasi menggunakan metode *MinMax scaler*. Hasil normalisasi data uji tersebut sesuai dengan standar normalisasi yang ditetapkan pada rumus *MinMax scaler* dimana akan menghasilkan angka pada jangkauan 0 sampai 1.

5.1.3. Pemodelan Algoritma *K-Nearest Neighbors*

Pemodelan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* dilakukan dengan menggunakan *library* dan *tools* yang sebelumnya telah dijabarkan. Adapun langkah komputasi pemodelan algoritma tersebut yaitu dimulai dengan menginisialisasikan fitur yang akan digunakan. Fitur pada variabel x merupakan *Open*, *High*, *Low*, dan *Vol* pada hari itu (d). Sedangkan, fitur pada variabel y adalah kolom *High* ($d+1$) dan *Low* ($d+1$). Setelah itu, dilakukan pembagian data dengan rasio 90:10 berdasarkan data latih dan data uji. Tahap selanjutnya pada pemodelan prediksi harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* adalah menentukan nilai K menggunakan metode Elbow. Hasil perhitungan metode Elbow disajikan dengan grafik untuk mempermudah penglihatan secara statistik. Adapun hasil perhitungan untuk mencari nilai K optimal tersaji pada gambar di bawah ini.



Gambar 5.15 Hasil Pemodelan Diagram Elbow Penelitian

Berdasarkan diagram Elbow pada gambar 5.15 di atas, diketahui bahwa nilai K optimal yang dapat digunakan pada tahap selanjutnya yaitu 4. Nilai terbaik yang dihasilkan pada operasi penghitungan melalui metode Elbow adalah K yang memiliki nilai *Mean Squared Error* (MSE) terendah. Oleh karena itu, nilai 4 akan digunakan sebagai K pada pemodelan untuk prediksi menggunakan algoritma *K-Nearest Neighbors*. Selanjutnya, dilakukan pemodelan berdasarkan data historis yang telah diinisialisasikan dan ternormalisasi. Data uji dimasukkan ke dalam suatu

array untuk mempermudah proses prediksi. *Array* tersebut digunakan untuk menyimpan setiap hasil yang diperoleh. *Array* dapat digunakan di berbagai kondisi seperti yang dibutuhkan pada pemodelan algoritma *K-Nearest Neighbors*.

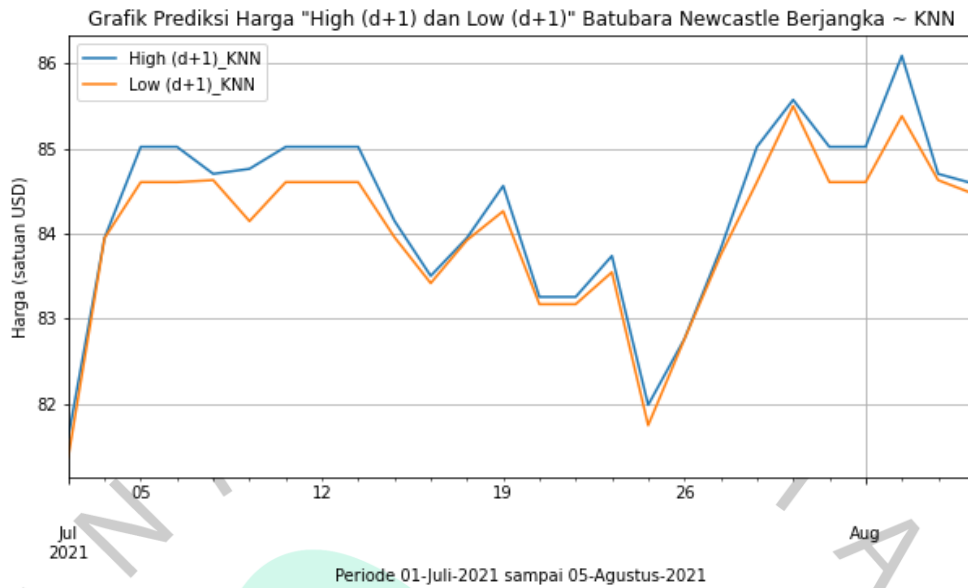
Algoritma *K-Nearest Neighbors* melakukan prediksi berdasarkan jumlah tetangga yang diinisialisasikan yaitu 4. Setelah proses pemodelan telah selesai, maka terlebih dahulu dilakukan denormalisasi data. Logika yang dijalankan pada proses denormalisasi data adalah hasil prediksi data dikalikan dengan hasil dari pengurangan antara nilai tertinggi dengan nilai terendah dari kolom yang dipilih serta dijumlah dengan nilai minimum dari kolom tersebut. Hasil denormalisasi harga (*High (d+1)* dan *Low (d+1)*) dimasukkan ke dalam *array* untuk mempermudah proses pembuatan pemodelan berupa luaran grafik. Grafik dirancang untuk membantu pengguna memahami isi luaran. Grafik akan dihasilkan dengan menggunakan *library* Pandas yang telah menyimpan seluruh baris model prediksi ke dalam *array*. Adapun hasil pemodelan prediksi menggunakan algoritma *K-Nearest Neighbors* disajikan pada tabel dan grafik berikut.

Tabel 5.5 Hasil Pemodelan Prediksi Algoritma *K-Nearest Neighbors*

<i>Date</i>	<i>High (d+1)_KNN</i>	<i>Low (d+1)_KNN</i>
2021-07-01	81.60588	81.36701
2021-07-02	83.94706	83.94366
2021-07-05	85.01765	84.60253
2021-07-06	85.01765	84.60253
2021-07-07	84.70000	84.62606
2021-07-08	84.75882	84.14368
2021-07-09	85.01765	84.60253
2021-07-12	85.01765	84.60253
2021-07-13	85.01765	84.60253
2021-07-14	84.14706	83.95543
2021-07-15	83.50000	83.41421
2021-07-16	83.94706	83.92013
2021-07-19	84.55882	84.26133
2021-07-20	83.25294	83.16714
2021-07-21	83.25294	83.16714
2021-07-22	83.73529	83.54363
2021-07-23	81.98235	81.74351
2021-07-26	82.75882	82.75534
2021-07-27	83.81765	83.74365
2021-07-28	85.01765	84.60253

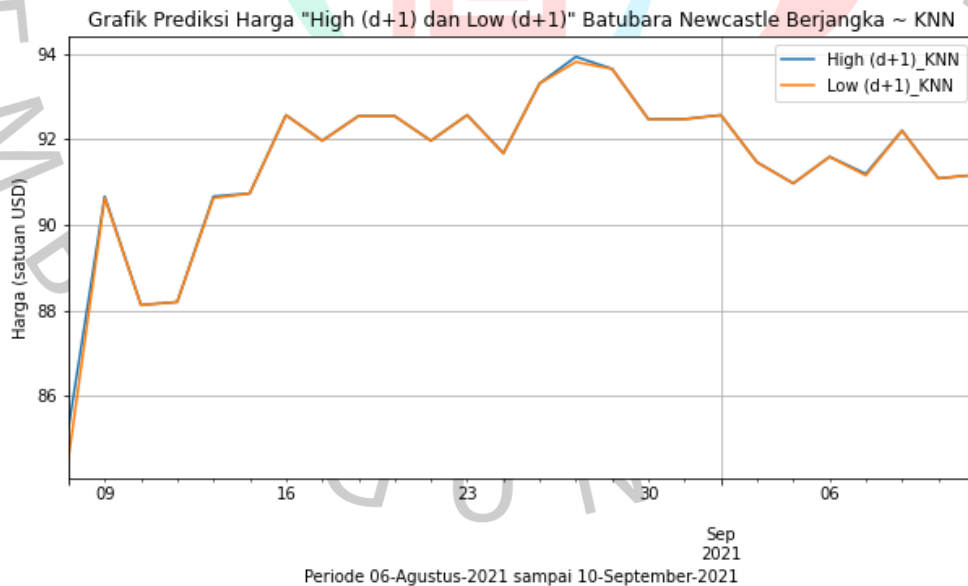
2021-07-29	85.57059	85.49671
2021-07-30	85.01765	84.60253
2021-08-02	85.01765	84.60253
2021-08-03	86.08824	85.37906
2021-08-04	84.70000	84.62606
2021-08-05	84.58235	84.46135
2021-08-06	85.24118	84.53194
2021-08-09	90.66471	90.62648
2021-08-10	88.12353	88.12043
2021-08-11	88.19412	88.19102
2021-08-12	90.66471	90.62648
2021-08-13	90.73529	90.73237
2021-08-16	92.57059	92.56780
2021-08-17	91.97059	91.96775
2021-08-18	92.54706	92.54426
2021-08-19	92.54706	92.54426
2021-08-20	91.97059	91.96775
2021-08-23	92.57059	92.56780
2021-08-24	91.67647	91.67361
2021-08-25	93.31176	93.30902
2021-08-26	93.93529	93.81494
2021-08-27	93.65294	93.65022
2021-08-30	92.47647	92.47367
2021-08-31	92.47647	92.47367
2021-09-01	92.57059	92.56780
2021-09-02	91.46471	91.46184
2021-09-03	90.97059	90.96768
2021-09-06	91.59412	91.59126
2021-09-07	91.19412	91.15593
2021-09-08	92.20588	92.20306
2021-09-09	91.08824	91.08534
2021-09-10	91.17059	91.16770
2021-09-13	92.57059	92.56780
2021-09-14	91.25294	91.25006
2021-09-15	91.97059	91.96775
2021-09-16	91.01765	90.97945
2021-09-17	90.64118	90.63825
2021-09-20	91.01765	90.97945
2021-09-21	90.66471	90.62648
2021-09-22	90.64118	90.63825
2021-09-23	90.64118	90.63825
2021-09-24	90.66471	90.62648
2021-09-27	90.73529	90.73237
2021-09-28	91.25294	91.25006

2021-09-29	91.97059	91.96775
2021-09-30	93.97059	93.96789
2021-10-01	95.05294	94.87384
2021-10-04	95.42941	95.32093
2021-10-05	96.55882	96.55631
2021-10-06	97.95882	97.95641
2021-10-07	96.08824	96.08569
2021-10-08	95.50000	95.42682
2021-10-11	96.67647	96.63867
2021-10-12	95.95882	95.77978
2021-10-13	97.13529	97.13282
2021-10-14	97.75882	97.75639
2021-10-15	99.04118	99.03884
2021-10-18	102.2647	101.7567
2021-10-19	102.2412	102.2391
2021-10-20	102.2882	102.2861
2021-10-21	103.0529	103.0509
2021-10-22	114.1941	114.1928
2021-10-25	114.1941	114.1928
2021-10-26	114.4529	114.4517
2021-10-27	115.4176	115.4165
2021-10-28	115.4176	115.4165
2021-10-29	114.0765	114.0752
2021-11-01	114.4529	114.4517
2021-11-02	114.4529	114.4517
2021-11-03	114.4529	114.4517
2021-11-04	114.4529	114.4517
2021-11-05	114.4529	114.4517
2021-11-08	114.4529	114.4517
2021-11-09	114.4529	114.4517
2021-11-10	114.4529	114.4517
2021-11-11	114.4529	114.4517
2021-11-12	114.4529	114.4517
2021-11-15	114.4529	114.4517
2021-11-16	114.4529	114.4517
2021-11-17	114.4529	114.4517
2021-11-18	114.4529	114.4517
2021-11-19	111.9941	111.9927
2021-11-22	114.4529	114.4517
2021-11-23	114.3353	114.3340



Gambar 5.16 Grafik Prediksi Periode 1 Juli - 5 Agustus 2021 (KNN)

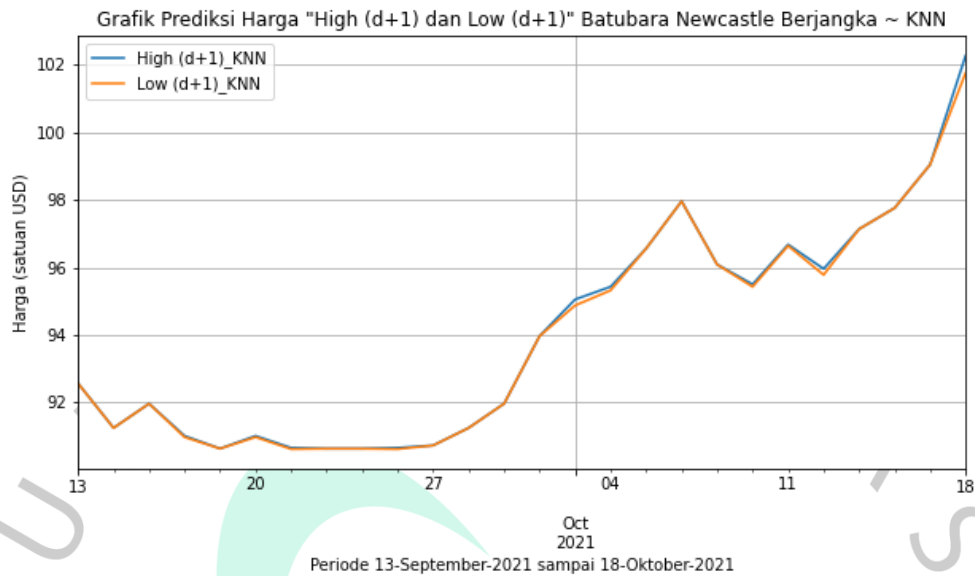
Berdasarkan gambar 5.16 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 1 Juli – 5 Agustus 2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan setiap harinya, namun masih dalam harga \$80.



Gambar 5.17 Grafik Prediksi Periode 6 Agustus - 10 September 2021 (KNN)

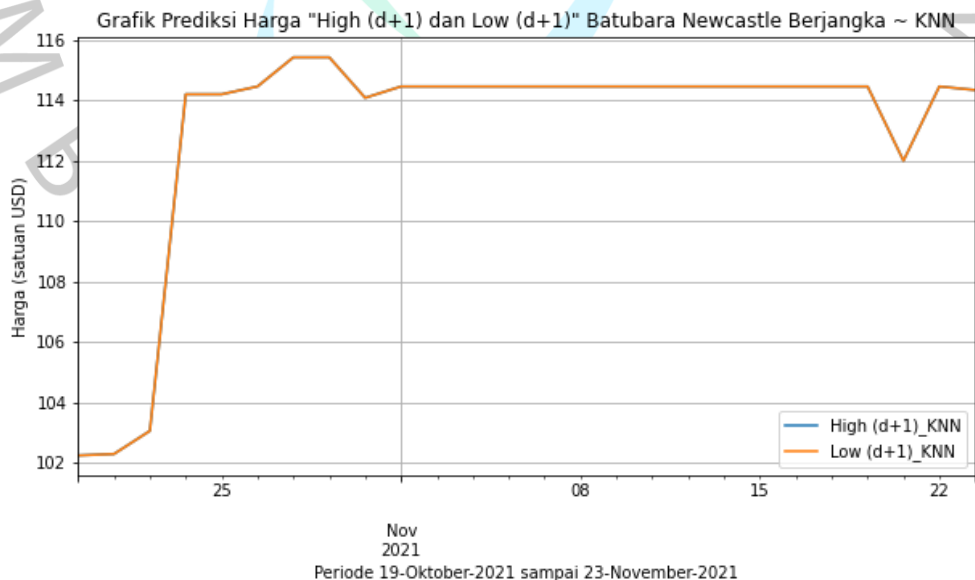
Berdasarkan gambar 5.17 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 6 Agustus – 10 September

2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan setiap harinya dimana telah mencapai di atas \$90.

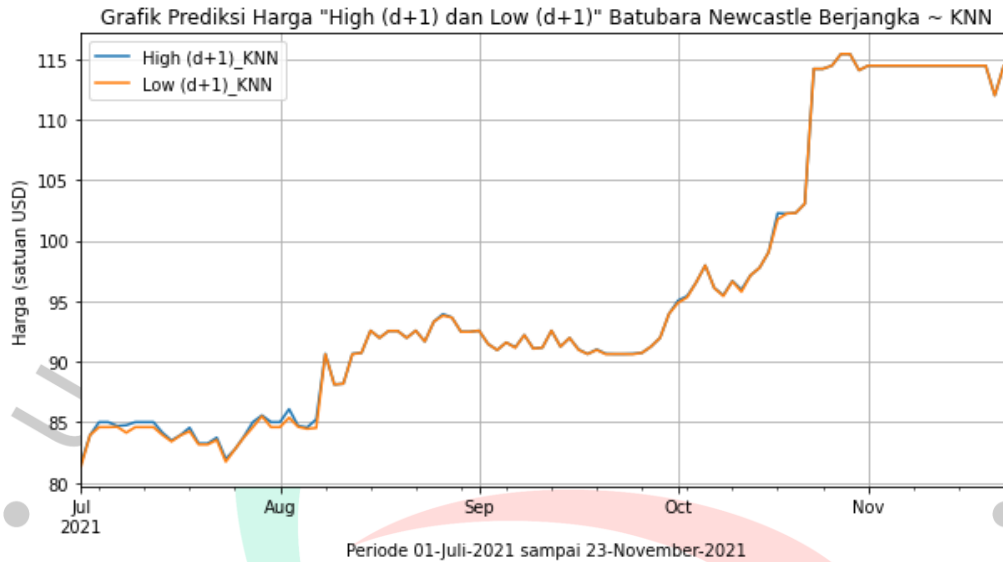


Gambar 5.18 Grafik Prediksi Periode 13 September - 18 Oktober 2021 (KNN)

Berdasarkan gambar 5.18 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 13 September – 18 Oktober 2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan setiap harinya dimana telah mencapai di atas \$100 pada akhir periode grafik tersebut.



Berdasarkan gambar 5.19 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 19 Oktober – 23 November 2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan tinggi pada 22 Oktober 2021 dan mulai kongsan sampai 23 November 2021.



Gambar 5.20 Grafik Prediksi Periode 1 Juli - 23 November 2021 (KNN)

Prediksi yang dihasilkan dengan menggunakan algoritma *K-Nearest Neighbors* dimulai dari 1 Juli 2021 sampai dengan 23 November 2021 atau setara dengan 104 hari terhitung hari kerja (Senin sampai Jumat). Sebagai wujud untuk mempermudah pembacaan dan penglihatan grafik, maka grafik dibagi menjadi empat bagian periode seperti pada gambar 5.16 sampai 5.19 dengan interval 5 hari. Namun, pada gambar 5.20 memiliki interval bulanan (setiap awal bulan) secara keseluruhan pemodelan prediksi. Berdasarkan luaran berupa harga prediksi (*High (d+1)* dan *Low (d+1)*) batubara Newcastle berjangka pada tabel 5.5 di atas, dapat terlihat bahwa harga batubara mengalami naik dan turun yang tidak menentu. Mulai dari Juli sampai awal Oktober, pemodelan prediksi yang dihasilkan dapat dikatakan normal. Namun, pada pertengahan Oktober dihasilkan prediksi yang mengalami kenaikan secara signifikan walaupun setelah itu titik garis memberikan hasil yang datar. Grafik di atas hanya berdasarkan data historis mulai dari Juli 2017 sampai dengan Juni 2021. Dengan demikian, jika terdapat fluktuasi atau pergerakan pasar yang tidak stabil baik diakibatkan secara sengaja seperti adanya korupsi maupun

secara tidak sengaja seperti terjadinya bencana alam, maka hal tersebut tidak terhitung dalam grafik hasil pemodelan prediksi tersebut.

5.1.4. Pemodelan Algoritma *Vector Autoregression*

Pemodelan pergerakan harga batubara Newcastle berjangka menggunakan algoritma *Vector Autoregression* diprogram menggunakan bahasa pemrograman Python dengan berbagai *library* yang diperlukan. Adapun langkah komputasi algoritma tersebut dimulai dengan melakukan pembagian data antara data latih dengan data uji. Rasio yang digunakan pada algoritma *Vector Autoregression* sama seperti *K-Nearest Neighbors* yaitu 90:10 terhadap data latih dan data uji. Data uji yang digunakan sebanyak 104 baris dari total 1031 data atau setara dengan 10% dari keseluruhan data.

Proses selanjutnya dalam membuat pemodelan prediksi harga batubara Newcastle berjangka adalah dengan melakukan pengujian stasioneritas setiap kolom pada data historis menggunakan *Augmented Dickey-Fuller (AD-Fuller)*. Pengujian tersebut digunakan untuk memeriksa apakah kolom pada data latih yang akan di program menggunakan algoritma *Vector Autoregression* dapat menolak hipotesis bernilai nol atau tidak. Nilai signifikan pada algoritma *Vector Autoregression* memiliki ketetapan yaitu 0.05. Pengujian *AD-Fuller* menggunakan fungsi *autolag AIC* yang merupakan *lag order default* dari penggunaan *AD-Fuller*. Adapun hasil pengujian setiap kolom menggunakan *Augmented Dickey-Fuller* sebagai berikut.

Augmented Dickey-Fuller Test pada kolom Open

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05

=> Nilai ADF = 0.799 || Tidak dapat menolak hipotesis nol.

=> Data tidak stasioner

Augmented Dickey-Fuller Test pada kolom High

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05

=> Nilai ADF = 0.8 || Tidak dapat menolak hipotesis nol.

=> Data tidak stasioner

Gambar 5.21 Hasil Pengujian AD-Fuller (1)

Augmented Dickey-Fuller Test pada kolom Low

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05

=> Nilai ADF = 0.797 || Tidak dapat menolak hipotesis nol.

=> Data tidak stasioner

Augmented Dickey-Fuller Test pada kolom Vol

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05

=> Nilai ADF = 0.0 || Menolak hipotesis nol.

=> Data sudah stasioner

Gambar 5.22 Hasil Pengujian AD-Fuller (2)

```

Augmented Dickey-Fuller Test pada kolom High (d+1)
-----
Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari
nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.802 || Tidak dapat menolak hipotesis nol.
=> Data tidak stasioner

Augmented Dickey-Fuller Test pada kolom Low (d+1)
-----
Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari
nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.8 || Tidak dapat menolak hipotesis nol.
=> Data tidak stasioner

```

Gambar 5.23 Hasil Pengujian AD-Fuller (3)

Berdasarkan gambar 5.21 sampai 5.23 di atas, beberapa kolom data harga batubara Newcastle berjangka tidak stasioner. Dengan demikian, setiap kolom tidak dapat menolak hipotesis nol. Oleh karena itu, pengujian dilakukan kembali dengan melakukan transformasi data menggunakan fungsi *diff* dan *dropna*. Setelah transformasi data dilakukan, pengujian AD-Fuller diproses kembali. Adapun hasil pengujian AD-Fuller setelah melakukan transformasi data sebagai berikut.

```

Augmented Dickey-Fuller Test pada kolom Open
-----
Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari
nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

Augmented Dickey-Fuller Test pada kolom High
-----
Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari
nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

```

Gambar 5.24 Hasil Pengujian AD-Fuller Setelah Transformasi Data (1)

Augmented Dickey-Fuller Test pada kolom Low

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

Augmented Dickey-Fuller Test pada kolom Vol

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

Gambar 5.25 Hasil Pengujian AD-Fuller Setelah Transformasi Data (2)

Augmented Dickey-Fuller Test pada kolom High (d+1)

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

Augmented Dickey-Fuller Test pada kolom Low (d+1)

Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari nilai signifikan (tidak stasioner)

Signifikan Level ADF = 0.05
=> Nilai ADF = 0.0 || Menolak hipotesis nol.
=> Data sudah stasioner

Gambar 5.26 Hasil Pengujian AD-Fuller Setelah Transformasi Data (3)

Berdasarkan gambar 5.24 sampai 5.26 di atas, seluruh kolom telah memiliki nilai *Augmented Dickey-Fuller* kurang dari skor signifikan. Seluruh kolom telah menolak argumen hipotesis nol. Dengan demikian, seluruh kolom pada data historis harga batubara Newcastle berjangka telah stasioner. Data tersebut akan digunakan untuk melakukan prediksi dalam membuat pemodelan pergerakan harga batubara Newcastle berjangka. Tahap selanjutnya yaitu mencetak hasil *lag order* untuk

mengeluarkan model yang optimal. *Lag order* yang digunakan berupa perhitungan dari *Akaike Information Criterion* (AIC). Perhitungan *Akaike Information Criterion* (AIC) dikerjakan menggunakan program dikarenakan hal tersebut sulit dihitung secara manual serta mengurangi kesalahan yang dibuat oleh manusia. Adapun hasil *Akaike Information Criterion* (AIC) yang dikeluarkan sebanyak 6 *lag order* tersaji pada tabel berikut ini.

Tabel 5.6 Luaran *Lag Order*

<i>Lag Order</i>	Nilai AIC
1	-184.16200156692520
2	-183.31267285518368
3	-184.39391485192820
4	-182.56752102532855
5	-184.27494868249790
6	-184.08866234450870

Berdasarkan nilai AIC yang dihasilkan dari tabel 5.6 di atas, dapat disimpulkan bahwa nilai terendah yang diperoleh berada pada saat menggunakan *lag order* ke-3. Dengan demikian, *lag order* ke-3 digunakan untuk melatih model pada algoritma *Vector Autoregression*. Pemodelan dilakukan dengan menyesuaikan bobot berdasarkan nilai data sehingga prediksi akan dihasilkan dengan akurasi terbaik. Setelah melatih model algoritma *Vector Autoregression*, data berupa *lag order*, hasil transformasi data pertama, hasil pelatihan model, dan rentang batasan waktu yang ingin diprediksi diprogram untuk melakukan pemodelan data prediksi harga batubara Newcastle berjangka. Rentang batasan waktu pada program dituliskan dengan menggunakan frekuensi *B* yang berarti bahwa data akan dikeluarkan setiap hari kerja (Senin sampai Jumat) dalam satu minggu. Data pemodelan prediksi yang dikeluarkan sebanyak 104 hari terhitung sejak 1 Juli 2021 sampai dengan 23 November 2021.

Hasil pemodelan prediksi batubara Newcastle berjangka yang telah selesai diproses masih berupa angka transformasi. Oleh karena itu, dilakukan pengembalian data yang telah ditransformasi sebelumnya. Proses mengembalikan transformasi data yaitu dilakukan menggunakan data latih yang telah dinormalisasi dengan hasil dari prediksi yang diperoleh. Setelah itu, data prediksi harga batubara Newcastle berjangka di denormalisasi dengan menggunakan rumus denormalisasi. Hasil pengembalian transformasi data dikalikan dengan nilai tertinggi dan

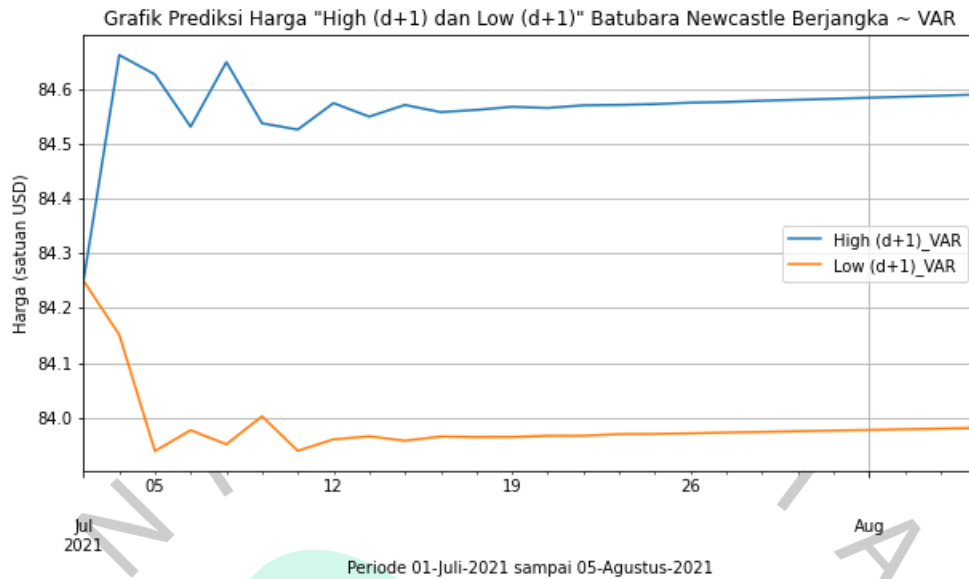
dikurangi dengan nilai terendah dari kolom data yang akan dinormalisasi serta ditambahkan dengan nilai terendah dari kolom terpilih tersebut. Dengan demikian, hasil pemodelan pada prediksi harga batubara Newcastle berjangka dapat dilihat dengan lebih mudah dan sesuai dengan nilai aslinya setelah proses denormalisasi selesai dilakukan. Adapun hasil pemodelan pada prediksi harga batubara Newcastle berjangka menggunakan algoritma *Vector Autoregression* tersaji pada tabel dan grafik di berikut ini.

Tabel 5.7 Hasil Pemodelan Prediksi Algoritma *Vector Autoregression*

<i>Date</i>	<i>High (d+1)_VAR</i>	<i>Low (d+1)_VAR</i>
2021-07-01	84.25000000	84.25000000
2021-07-02	84.66166368	84.15158878
2021-07-05	84.62618808	83.93918338
2021-07-06	84.53079321	83.97740572
2021-07-07	84.64879726	83.95167055
2021-07-08	84.53709692	84.00268711
2021-07-09	84.52565699	83.93983221
2021-07-12	84.57399425	83.96066755
2021-07-13	84.54921781	83.96630294
2021-07-14	84.57063879	83.95812137
2021-07-15	84.55749445	83.96597641
2021-07-16	84.56151098	83.96486801
2021-07-19	84.56722566	83.96508026
2021-07-20	84.56503348	83.96728397
2021-07-21	84.56993348	83.96739303
2021-07-22	84.57066597	83.97012145
2021-07-23	84.57209287	83.97051545
2021-07-26	84.57471576	83.97163724
2021-07-27	84.57597339	83.97310358
2021-07-28	84.57830862	83.97417765
2021-07-29	84.58001093	83.97552350
2021-07-30	84.58181340	83.97666871
2021-08-02	84.58380655	83.97786382
2021-08-03	84.58559157	83.97912628
2021-08-04	84.58751816	83.98031079
2021-08-05	84.58937923	83.98154828
2021-08-06	84.59123680	83.98275604
2021-08-09	84.59313017	83.98396897
2021-08-10	84.59498993	83.98519191
2021-08-11	84.59687041	83.98640403
2021-08-12	84.59874291	83.98762272

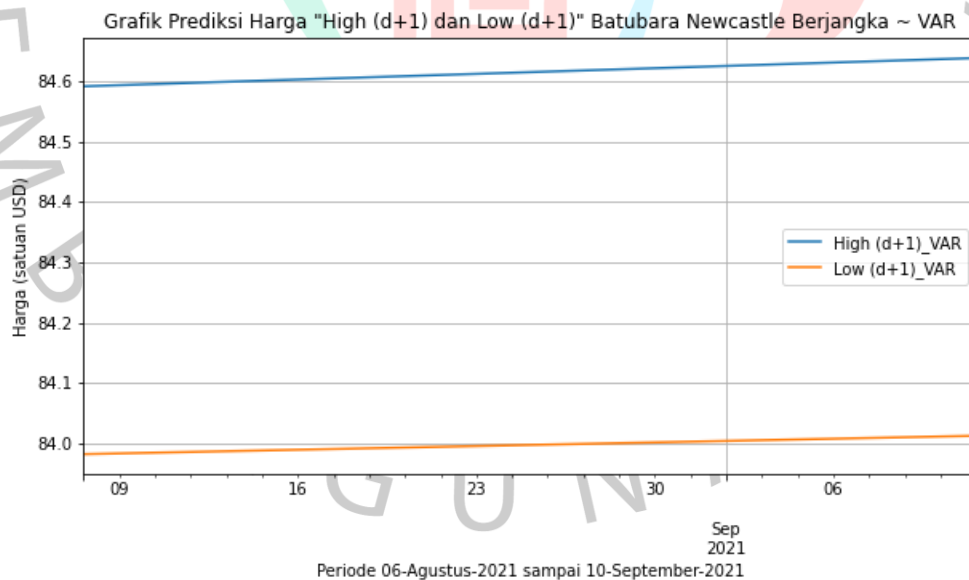
2021-08-13	84.60061368	83.98883855
2021-08-16	84.60249012	83.99005419
2021-08-17	84.60436170	83.99127169
2021-08-18	84.60623590	83.99248739
2021-08-19	84.60810942	83.99370412
2021-08-20	84.60998239	83.99492050
2021-08-23	84.61185630	83.99613679
2021-08-24	84.61372948	83.99735335
2021-08-25	84.61560301	83.99856967
2021-08-26	84.61747649	83.99978612
2021-08-27	84.61934986	84.00100254
2021-08-30	84.62122338	84.00221893
2021-08-31	84.62309679	84.00343537
2021-09-01	84.62497025	84.00465177
2021-09-02	84.62684370	84.00586819
2021-09-03	84.62871714	84.00708461
2021-09-06	84.63059059	84.00830102
2021-09-07	84.63246404	84.00951743
2021-09-08	84.63433749	84.01073385
2021-09-09	84.63621093	84.01195026
2021-09-10	84.63808438	84.01316668
2021-09-13	84.63995783	84.01438309
2021-09-14	84.64183128	84.01559951
2021-09-15	84.64370472	84.01681592
2021-09-16	84.64557817	84.01803234
2021-09-17	84.64745162	84.01924875
2021-09-20	84.64932507	84.02046517
2021-09-21	84.65119852	84.02168158
2021-09-22	84.65307196	84.02289799
2021-09-23	84.65494541	84.02411441
2021-09-24	84.65681886	84.02533082
2021-09-27	84.65869231	84.02654724
2021-09-28	84.66056575	84.02776365
2021-09-29	84.66243920	84.02898007
2021-09-30	84.66431265	84.03019648
2021-10-01	84.66618610	84.03141290
2021-10-04	84.66805954	84.03262931
2021-10-05	84.66993299	84.03384573
2021-10-06	84.67180644	84.03506214
2021-10-07	84.67367989	84.03627856
2021-10-08	84.67555333	84.03749497
2021-10-11	84.67742678	84.03871139
2021-10-12	84.67930023	84.03992780
2021-10-13	84.68117368	84.04114421

2021-10-14	84.68304713	84.04236063
2021-10-15	84.68492057	84.04357704
2021-10-18	84.68679402	84.04479346
2021-10-19	84.68866747	84.04600987
2021-10-20	84.69054092	84.04722629
2021-10-21	84.69241436	84.04844270
2021-10-22	84.69428781	84.04965912
2021-10-25	84.69616126	84.05087553
2021-10-26	84.69803471	84.05209195
2021-10-27	84.69990815	84.05330836
2021-10-28	84.70178160	84.05452478
2021-10-29	84.70365505	84.05574119
2021-11-01	84.70552850	84.05695760
2021-11-02	84.70740195	84.05817402
2021-11-03	84.70927539	84.05939043
2021-11-04	84.71114884	84.06060685
2021-11-05	84.71302229	84.06182326
2021-11-08	84.71489574	84.06303968
2021-11-09	84.71676918	84.06425609
2021-11-10	84.71864263	84.06547251
2021-11-11	84.72051608	84.06668892
2021-11-12	84.72238953	84.06790534
2021-11-15	84.72426297	84.06912175
2021-11-16	84.72613642	84.07033817
2021-11-17	84.72800987	84.07155458
2021-11-18	84.72988332	84.07277100
2021-11-19	84.73175676	84.07398741
2021-11-22	84.73363021	84.07520382
2021-11-23	84.73550366	84.07642024



Gambar 5.27 Grafik Prediksi Periode 1 Juli - 5 Agustus 2021 (VAR)

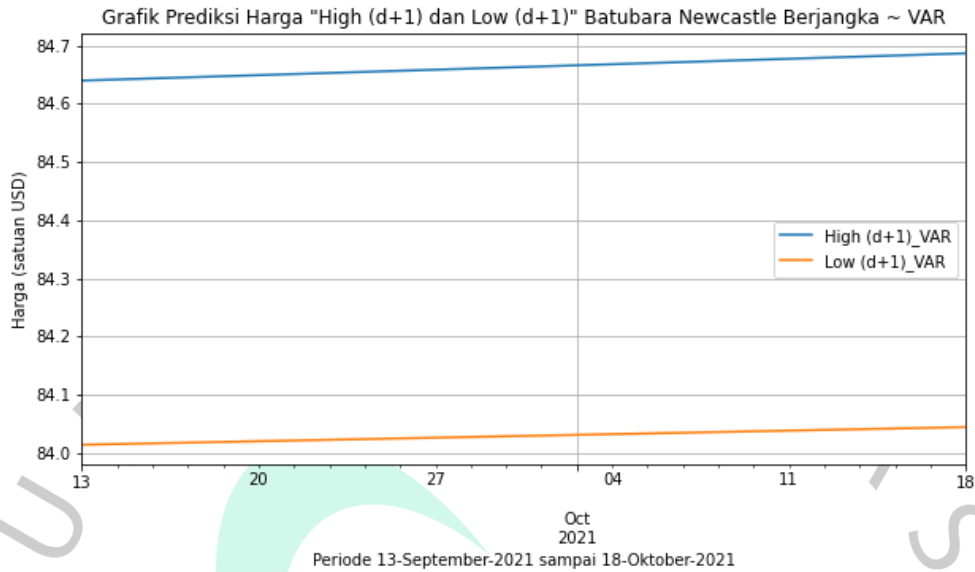
Berdasarkan gambar 5.27 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 1 Juli – 5 Agustus 2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan di awal periode, namun setelah 12 Juli 2021 harga naik konstan.



Gambar 5.28 Grafik Prediksi Periode 6 Agustus - 10 September 2021 (VAR)

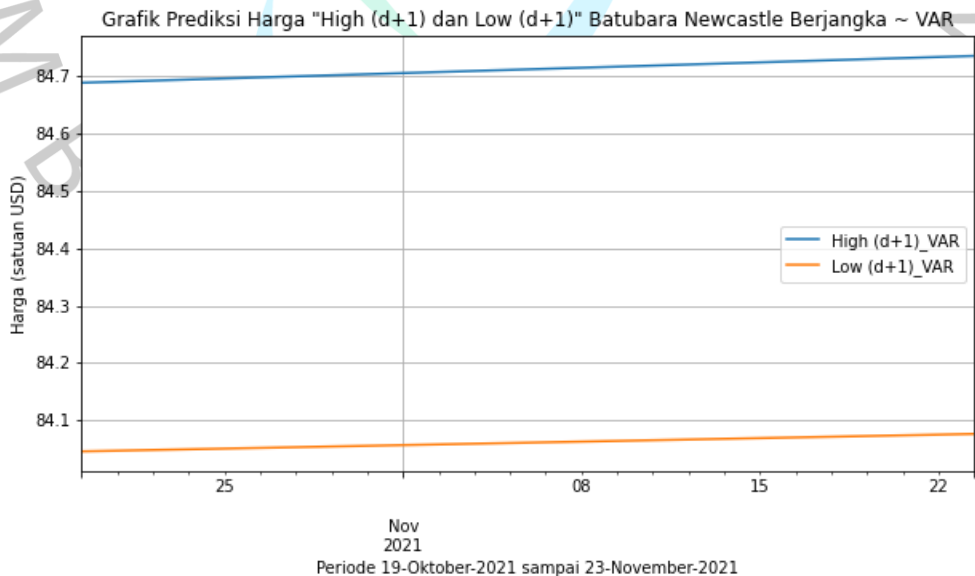
Berdasarkan gambar 5.28 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 6 Agustus – 10 September

2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan harga yang konstan di \$84.



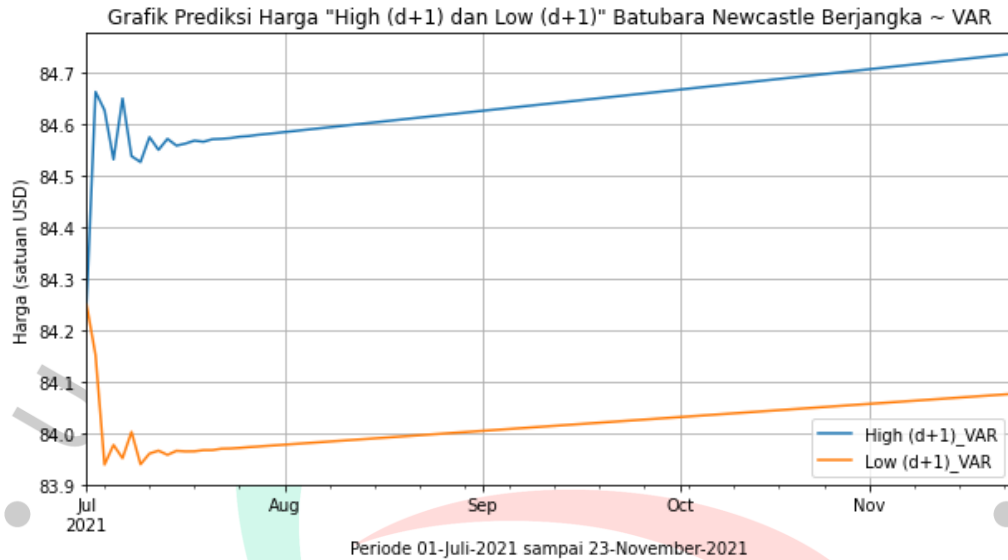
Gambar 5.29 Grafik Prediksi Periode 13 September - 18 Oktober 2021 (VAR)

Berdasarkan gambar 5.29 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 13 September – 18 Oktober 2021, harga *High (d+1)* dan *Low (d+1)* mengalami perubahan harga yang konstan di \$84.



Gambar 5.30 Grafik Prediksi Periode 19 Oktober - 23 November 2021 (VAR)

Berdasarkan gambar 5.30 dapat diuraikan bahwa grafik ditampilkan dalam interval 5 hari kerja dengan satuan USD. Pada periode 19 Oktober – 23 November 2021, harga $High(d+1)$ dan $Low(d+1)$ mengalami perubahan harga yang konstan di \$84.

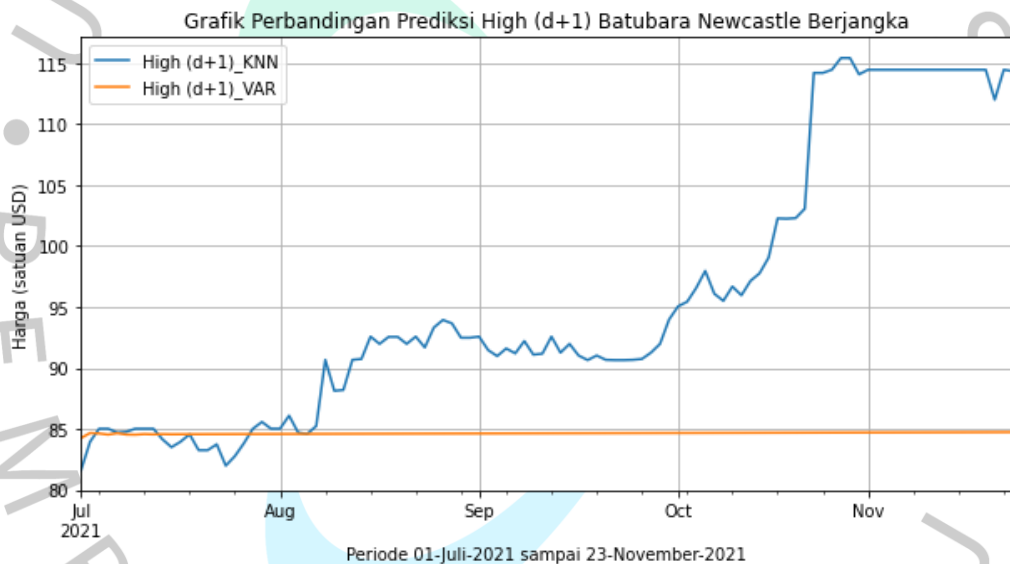


Gambar 5.31 Grafik Prediksi Periode 1 Juli - 23 November 2021 (VAR)

Prediksi yang dihasilkan dengan menggunakan algoritma *Vector Autoregression* dimulai sejak 1 Juli 2021 sampai dengan 23 November 2021 atau sebanyak 104 hari terhitung selama hari kerja yaitu Senin sampai Jumat berdasarkan banyaknya data uji. Grafik yang diberikan pada laporan ini dibagi menjadi empat periode untuk memudahkan dalam membaca pemodelan prediksi harga batubara Newcastle berjangka. Grafik tersebut tersaji pada gambar 5.27 sampai 5.30 dengan interval 5 hari kerja. Adapun pada gambar 5.31 berisikan data prediksi dengan interval bulanan. Berdasarkan tabel 5.7 dan gambar 5.31 di atas, dapat terlihat bahwa prediksi harga ($High(d+1)$ dan $Low(d+1)$) batubara Newcastle berjangka naik dan turun pada awal hingga pertengahan Juli, namun setelah itu berjalan konstan atau tidak banyak peningkatan maupun penurunan. Data pemodelan prediksi tersebut diperoleh dengan berdasarkan historis harga sejak Juli 2017 sampai dengan Juni 2021. Dengan demikian, faktor-faktor lain diluar harga historis tersebut tidak diperhitungkan pada luaran data tabel dan grafik.

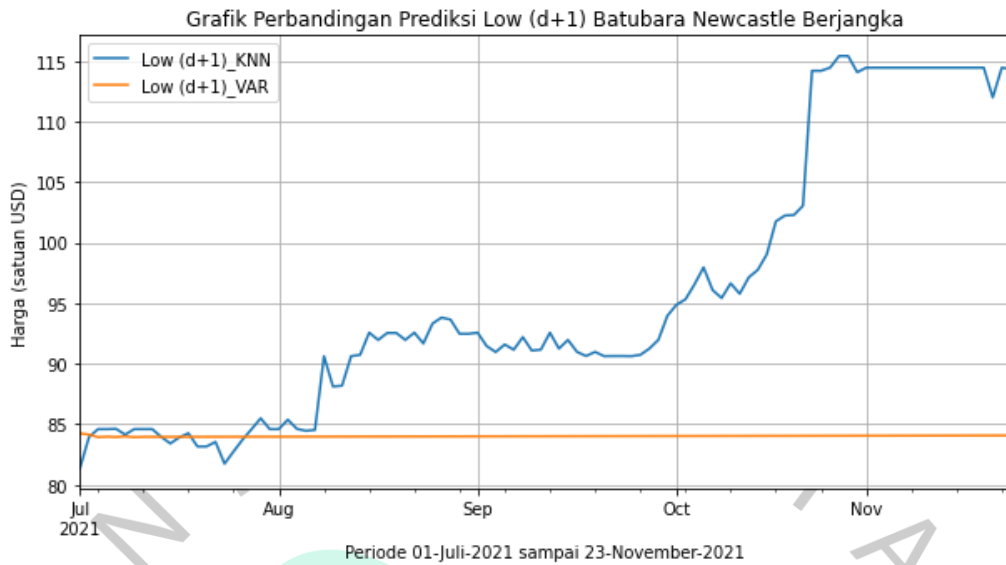
5.1.5. Perbandingan Hasil Pemodelan Algoritma

Pemodelan prediksi harga batubara Newcastle berjangka yang diperoleh menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression* dibandingkan pada suatu grafik. Perbandingan hasil ini dapat memperlihatkan perbedaan model yang dihasilkan pada setiap algoritma. Perbedaan hasil di setiap algoritma merupakan suatu kewajaran dalam pemrosesan data dengan *machine learning*. Grafik yang ditampilkan terbagi menjadi perbandingan antar *High (d+1)*, perbandingan antar *Low (d+1)*, perbandingan antara *High (d+1)* dengan *Low (d+1)* yang tersaji dengan interval bulanan, serta dipotong menjadi empat periode untuk mempermudah dalam melihat perbandingan dengan interval 5 hari kerja. Adapun perbandingan tersebut disajikan pada grafik di bawah ini.



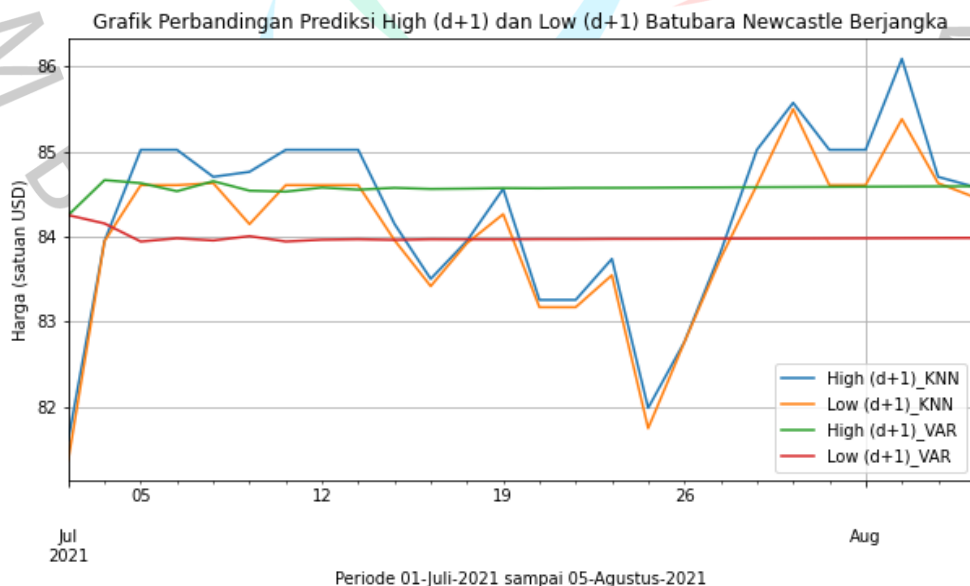
Gambar 5.32 Grafik *High (d+1)* Periode 1 Juli – 23 November 2021

Berdasarkan gambar 5.32 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 1 Juli 2021 sampai dengan 23 November 2021 atau sebanyak 104 hari dihitung selama hari kerja dengan satuan USD. Pada periode tersebut harga *High (d+1)_KNN* dan *High (d+1)_VAR* memiliki perbedaan hasil yang tajam. Pada *High (d+1)_KNN* terlihat model yang dihasilkan mengalami kenaikan harga mencapai di atas \$110, sedangkan pada *High (d+1)_VAR* konstan di harga \$84.



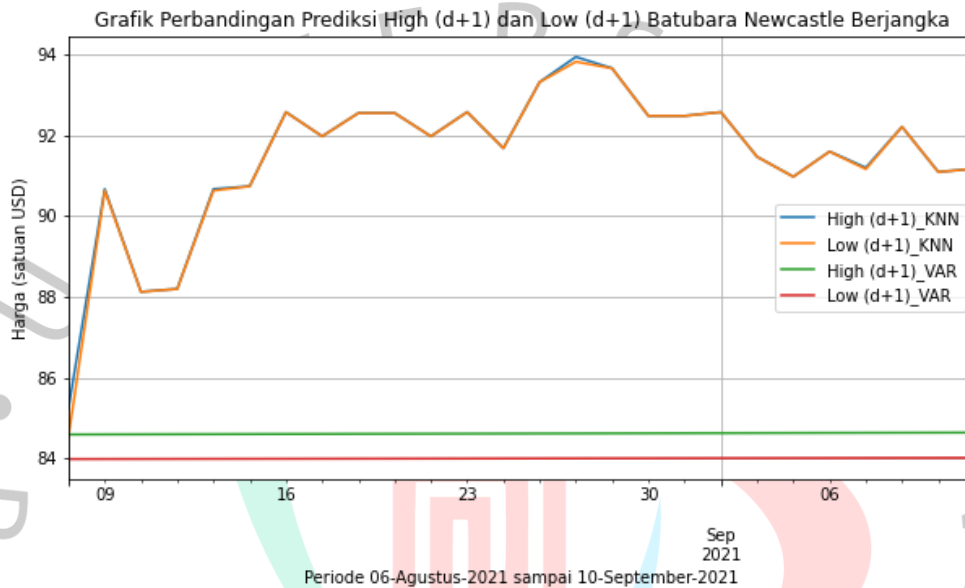
Gambar 5.33 Grafik $Low(d+1)$ Periode 1 Juli – 23 November 2021

Berdasarkan gambar 5.33 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 1 Juli 2021 sampai dengan 23 November 2021 atau sebanyak 104 hari terhitung selama hari kerja dengan satuan USD. Pada periode tersebut harga $Low(d+1)_KNN$ dan $Low(d+1)_VAR$ memiliki perbedaan hasil pemodelan. Pada $Low(d+1)_KNN$ terlihat model yang dihasilkan mencapai harga di atas \$110, sedangkan pada $Low(d+1)_VAR$ konstan di harga \$84.



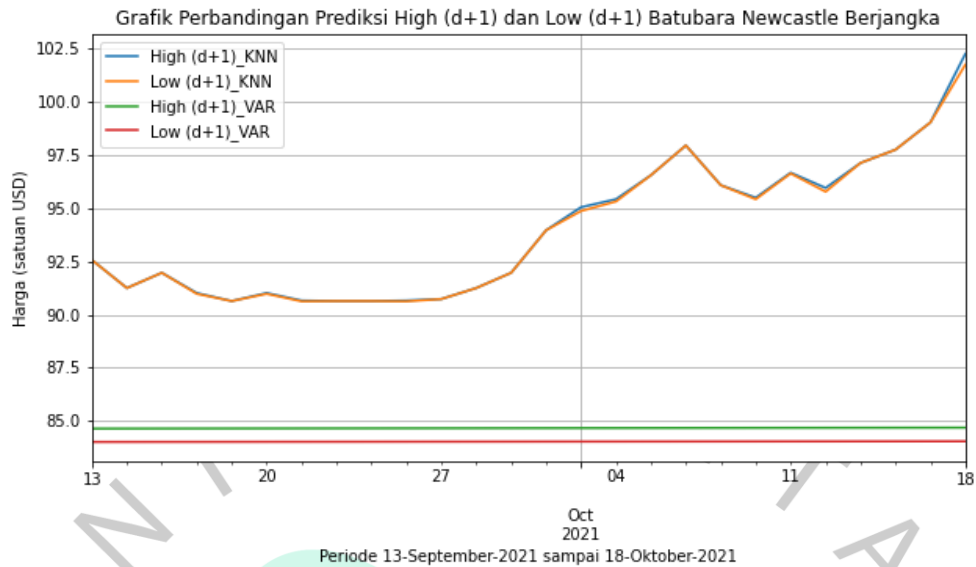
Gambar 5.34 Grafik Perbandingan Periode 1 Juli - 5 Agustus 2021

Berdasarkan gambar 5.34 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 1 Juli 2021 sampai 5 Agustus 2021 dengan interval 5 hari terhitung selama hari kerja dengan satuan USD. *High (d+1)* dan *Low (d+1)* algoritma *K-Nearest Neighbors* mengalami perubahan harga yang tidak menentu dengan hasil akhir yang lebih tinggi dari hasil awalnya, sedangkan pada *High (d+1)* dan *Low (d+1)* algoritma *Vector Autoregression* terlihat konstan di angka \$84.



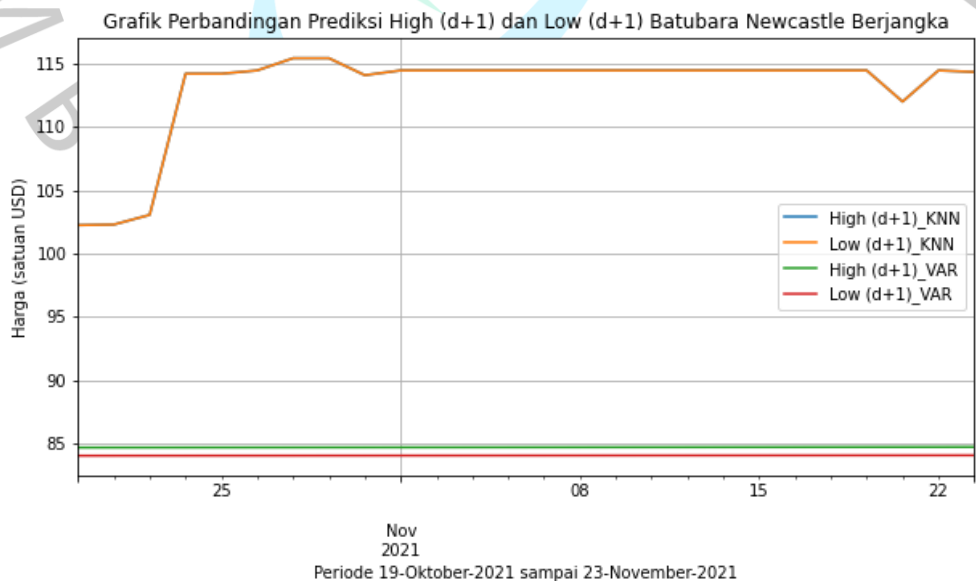
Gambar 5.35 Grafik Perbandingan Periode 6 Agustus - 10 September 2021

Berdasarkan gambar 5.35 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 6 Agustus 2021 sampai 10 September 2021 dengan interval 5 hari terhitung selama hari kerja dengan satuan USD. *High (d+1)* dan *Low (d+1)* algoritma *K-Nearest Neighbors* mengalami perubahan harga yang tinggi dengan hasil akhir di atas \$90 sedangkan pada *High (d+1)* dan *Low (d+1)* algoritma *Vector Autoregression* terlihat konstan di angka \$84.



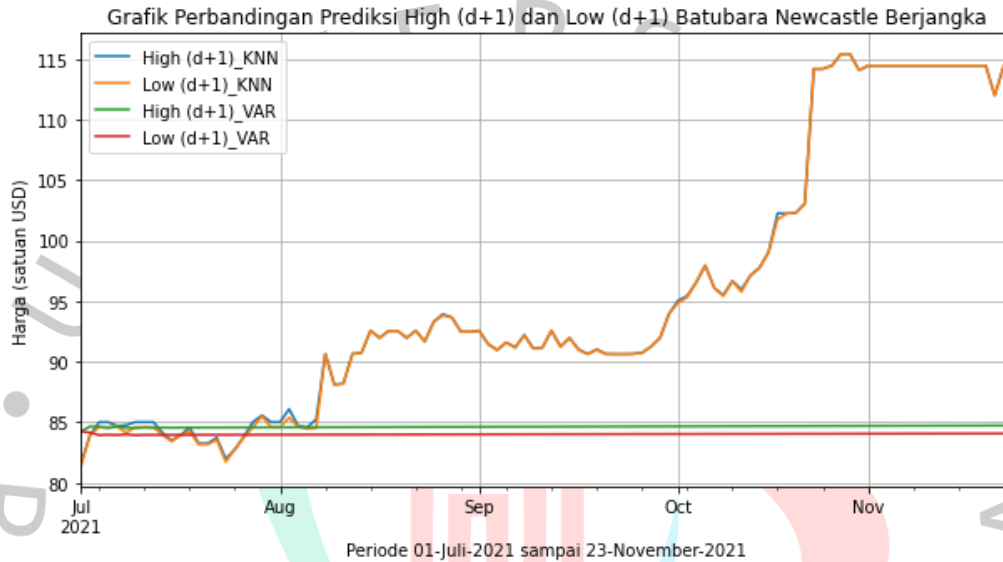
Gambar 5.36 Grafik Perbandingan Periode 13 September - 18 Oktober 2021

Berdasarkan gambar 5.36 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 13 September 2021 sampai 18 Oktober 2021 dengan interval 5 hari terhitung selama hari kerja dengan satuan USD. *High (d+1)* dan *Low (d+1)* algoritma *K-Nearest Neighbors* mengalami perubahan harga yang melambung signifikan dimana hasil akhir pada periode ini berada di angka di atas \$100 sedangkan pada *High (d+1)* dan *Low (d+1)* algoritma *Vector Autoregression* terlihat konstan di angka \$84.



Gambar 5.37 Grafik Perbandingan Periode 19 Oktober - 23 November 2021

Berdasarkan gambar 5.37 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 19 Oktober 2021 sampai 23 November 2021 dengan interval 5 hari terhitung selama hari kerja dengan satuan USD. $High(d+1)$ dan $Low(d+1)$ algoritma *K-Nearest Neighbors* mengalami perubahan tinggi pada 22 Oktober 2021 dan mulai konstan sampai 23 November 2021 sedangkan pada $High(d+1)$ dan $Low(d+1)$ algoritma *Vector Autoregression* tetap terlihat konstan di angka \$84.



Gambar 5.38 Grafik Perbandingan Periode 1 Juli - 23 November 2021

Berdasarkan gambar 5.38 dapat diuraikan bahwa grafik ditampilkan dimulai sejak 1 Juli 2021 sampai dengan 23 November 2021 atau sebanyak 104 hari terhitung selama hari kerja dengan satuan USD. $High(d+1)$ dan $Low(d+1)$ algoritma *K-Nearest Neighbors* mengalami perubahan harga naik dan turun setiap harinya, sedangkan pada $High(d+1)$ dan $Low(d+1)$ algoritma *Vector Autoregression* menghasilkan harga yang konstan dari awal hingga akhir periode pemodelan.

5.2. Pembahasan

Pemodelan harga batubara Newcastle berjangka menggunakan *machine learning* merupakan suatu cara untuk memperoleh prediksi harga tersebut dengan mengaplikasikan algoritma *K-Nearest Neighbors* dan *Vector Autoregression*. Kedua algoritma tersebut akan dibandingkan tingkat akurasinya menggunakan *Root Mean Squared Error (RMSE)* dan *Mean Absolute Percentage Error (MAPE)*.

Tingkat akurasi algoritma *machine learning* akan diketahui dengan melakukan pengujian setiap baris kode untuk menyesuaikan dengan hasil yang diharapkan.

5.2.1. Pengujian Algoritma *K-Nearest Neighbors*

Pengujian pada algoritma *K-Nearest Neighbors* diperlukan untuk mengetahui apakah seluruh baris kode program telah dapat berjalan sesuai dengan hasil yang diharapkan. Adapun pengujian algoritma *K-Nearest Neighbors* pada pemodelan harga batubara Newcastle berjangka sebagai berikut.

Tabel 5.8 Pengujian Algoritma *K-Nearest Neighbors*

No	Kode Program	Hasil yang Diharapkan	Kesimpulan
1	<pre>x = data_nor[['Open', 'High', 'Low', 'Vol.']] y = data_nor[['High (d+1)', 'Low (d+1)']] x.shape, y.shape</pre>	Variabel terbuat untuk menyimpan fitur berupa kolom yang digunakan pada proses prediksi.	Valid
2	<pre>train_x, test_x, train_y, test_y = train_test_split(x, y, random_state = 0, test_size = 0.1, shuffle=False)</pre>	Data terbagi menjadi data latih dan data uji dengan rasio 90:10.	Valid
3	<pre>def Elbow(K): # Menginisialisasikan variabel sebagai array test_mse = [] # Melatih model untuk setiap nilai K for i in K: model_mse = knn_reg(n_neighbors=i) model_mse.fit(train_x, train_y) mse_predict = model_mse.predict(test_x) mse_predict = mse(mse_predict, test_y) test_mse.append(mse_predict) return test_mse # Menentukan batas nilai K ElbowRange = range(2, 11) MElbow = Elbow(ElbowRange) # Mengeluarkan diagram Elbow method plt.plot(ElbowRange, MElbow) plt.grid(color='grey') plt.title('Diagram Elbow') plt.xlabel('Nilai K') plt.ylabel('Mean Squared Error (MSE)')</pre>	Nilai <i>K</i> optimal dihasilkan dengan menggunakan metode Elbow. Nilai <i>K</i> dicetak dalam bentuk grafik berdasarkan hasil yang diperoleh dari perhitungan pada metode Elbow.	Valid
4	<pre># Memasukkan nilai K optimum knn_model = knn_reg(n_neighbors = 4) knn_model.fit(train_x, train_y) # Menginisialisasi isi kolom pada variabel baru untuk pemodelan data dp_knn = (x[['Open', 'High', 'Low', 'Vol']])</pre>	Variabel baru terinisialisasikan untuk memasukkan data yang akan dimodelkan dengan	Valid

```

5 # Menginisialisasikan variabel sebagai array
array_storage_high = []
array_storage_low = []

print("Data Prediksi Harga Batubara Newcastle Berjangka menggunakan Algoritma KNN")

# Proses Algoritma KNN
for predict_price in range(927, 1031):
    # Memasukkan nilai dp_knn ke dalam array
    data_knn = dp_knn.iloc[predict_price]
    pemodelan_prediksi_knn = np.array([data_knn])
    # Proses Prediksi harga Batubara Newcastle Berjangka
    prediksi_knn = knn_model.predict(pemodelan_prediksi_knn)
    # Denormalisasi hasil prediksi_knn HIGH
    min_knn_h = clean_kolom_dataset_original['High'].min()
    max_knn_h = clean_kolom_dataset_original['High'].max()
    denormalisasi_prediksi_knn_high = prediksi_knn[0][0] * (max_knn_h - min_knn_h) + min_knn_h
    # Denormalisasi hasil prediksi_knn LOW
    min_knn_l = clean_kolom_dataset_original['Low'].min()
    max_knn_l = clean_kolom_dataset_original['Low'].max()
    denormalisasi_prediksi_knn_low = prediksi_knn[0][1] * (max_knn_l - min_knn_l) + min_knn_l
    # Memasukkan denormalisasi_prediksi_knn ke dalam array
    array_storage_high.append(denormalisasi_prediksi_knn_high)
    array_storage_low.append(denormalisasi_prediksi_knn_low)

```

menggunakan nilai K optimal.

Prediksi harga batubara Newcastle berjangka dihasilkan dengan melakukan pemodelan pada algoritma *K-Nearest Neighbors*. Data prediksi tersebut di denormalisasi dan disimpan ke dalam *array*.

```

6 # Membaca hasil denormalisasi
grafik_prediksi_knn_h = pd.DataFrame(array_storage_high)
grafik_prediksi_knn_l = pd.DataFrame(array_storage_low)

# Menggabungkan kolom
merge_column_knn = pd.concat([grafik_prediksi_knn_h, grafik_prediksi_knn_l], axis=1)
merge_column_knn.columns = ['High (d+1)_KNN', 'Low (d+1)_KNN']
merge_column_knn['Date'] = pd.date_range('2021/07/01', periods=len(merge_column_knn), freq='B')
hasil_prediksi_knn = merge_column_knn.set_index('Date')
print(hasil_prediksi_knn)

```

Array hasil denormalisasi terbaca oleh variabel dan dikeluarkan menjadi tabel prediksi harga (*High (d+1)_KNN* dan *Low (d+1)_KNN*) dengan rentang waktu yang telah ditetapkan.

```

7 # Membuat grafik prediksi harga batubara Newcastle berjangka
judul_grafik_prediksi_knn = 'Grafik Prediksi Harga "High (d+1) dan Low (d+1)" Batubara Newcastle Berjangka ~ KNN'
label_grafik_prediksi_knn = 'Harga (satuannya USD)'
label_grafik_prediksi_knn = 'Periode 01-Juli-2021 sampai 23-November-2021'
# Menampilkan Grafik Prediksi Harga Batubara Newcastle Berjangka
hasil_prediksi_knn.plot(figsize=(10,5), legend=True, grid=True,
                        title=judul_grafik_prediksi_knn,
                        xlabel=label_grafik_prediksi_knn,
                        ylabel=label_grafik_prediksi_knn)

```

Hasil pemodelan prediksi pergerakan harga batubara Newcastle berjangka ditampilkan dalam bentuk grafik. Grafik yang dikeluarkan yaitu gabungan prediksi *High (d+1)_KNN - Low (d+1)_KNN*.

```

8 # Grafik Prediksi Harga Batubara Newcastle Berjangka - KNN
xlabel_grafik_prediksi_knn_1 = 'Periode 01-Juli-2021 sampai 05-Agustus-2021'
xlabel_grafik_prediksi_knn_2 = 'Periode 06-Agustus-2021 sampai 10-September-2021'
xlabel_grafik_prediksi_knn_3 = 'Periode 13-September-2021 sampai 18-Oktober-2021'
xlabel_grafik_prediksi_knn_4 = 'Periode 19-Oktober-2021 sampai 23-November-2021'

# Periode 01-Juli-2021 sampai 5-Agustus-2021
hasil_prediksi_knn.iloc[:26,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_knn,
xlabel=xlabel_grafik_prediksi_knn_1,
ylabel=ylabel_grafik_prediksi_knn)
# Periode 06-Agustus-2021 sampai 10-September-2021
hasil_prediksi_knn.iloc[26:52,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_knn,
xlabel=xlabel_grafik_prediksi_knn_2,
ylabel=ylabel_grafik_prediksi_knn)
# Periode 13-September-2021 sampai 18-Oktober-2021
hasil_prediksi_knn.iloc[52:78,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_knn,
xlabel=xlabel_grafik_prediksi_knn_3,
ylabel=ylabel_grafik_prediksi_knn)
# Periode 19-Oktober-2021 sampai 23-November-2021
hasil_prediksi_knn.iloc[78:104,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_knn,
xlabel=xlabel_grafik_prediksi_knn_4,
ylabel=ylabel_grafik_prediksi_knn)

```

Menampilkan Valid grafik prediksi harga batubara Newcastle berjangka berdasarkan periode yang telah ditentukan.

Berdasarkan tabel 5.8 di atas, seluruh pengujian telah dilakukan dan memberikan kesimpulan yang terbaik. Kesimpulan tersebut yaitu seluruh baris kode pada pemodelan pergerakan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* berhasil dijalankan dan memberikan luaran sesuai dengan yang peneliti harapkan.

5.2.2. Pengujian Algoritma *Vector Autoregression*

Kode program pada algoritma *Vector Autoregression* perlu dijalankan, diperiksa, dan diuji apakah setiap baris sudah sesuai dengan hasil yang diharapkan peneliti dalam membuat pemodelan pergerakan harga batubara Newcastle berjangka. Adapun pengujian algoritma tersebut diuraikan sebagai berikut.

Tabel 5.9 Pengujian Algoritma *Vector Autoregression*

No	Kode Program	Hasil yang Diharapkan	Kesimpulan
1	<pre> range_date = 104 data_nor_train, data_nor_test = data_nor[0:-range_date], data_nor[-range_date:] </pre>	Data terbagi menjadi data latih dan data uji berdasarkan rentang waktu prediksi yang ditentukan.	Valid

```

2 # Proses Augmented Dickey-Fuller
def adf_test(data_var, signifikan_lev=0.05, name=''):
    # Pengujian Stasioneritas
    data_adf = adfuller(data_var, autolag='AIC')
    output = {'nilaiadf':round(data_adf[1], 3)}
    nilai_adf = output['nilaiadf']

    # Cetak rangkuman hasil ADF pada setiap kolom
    print(' Augmented Dickey-Fuller Test pada kolom', name, "\n", '-'*65)
    print(' Hipotesis Nol: data memiliki akar root atau nilai ADF lebih dari',
          '\n', 'nilai signifikan (tidak stasioner)', '\n')
    print(' Signifikan Level ADF =', signifikan_lev)

    if nilai_adf <= signifikan_lev:
        print(' => Nilai ADF =', nilai_adf, '|| Menolak hipotesis nol.')
        print(' => Data sudah stasioner')
    else:
        print(' => Nilai ADF =', nilai_adf, '|| Tidak dapat menolak hipotesis nol.')
        print(' => Data tidak stasioner')

```

Fungsi *Augmented Dickey-Fuller (AD-Fuller)* dapat menguji stasioneritas pada setiap kolom data.

Valid

```

3 # Mencetak hasil ADF pada setiap kolom
for name, column in data_nor_train.items():
    adf_test(column, name = column.name)
    print('\n')

```

Setiap kolom diinformasikan (tercetak) apakah sudah stasioner atau belum.

Valid

```

4 # Transformasi pertama baris data
data_nor_diff = data_nor_train.diff().dropna()
print('Data Transformasi Pertama VAR')
print(data_nor_diff)

```

Seluruh baris data pada data latih di transformasikan dan tercetak di program.

Valid

```

5 # Mencetak hasil ADF pada setiap kolom yang telah diolah
# pada transformasi data pertama
for name, column in data_nor_diff.iteritems():
    adf_test(column, name = column.name)
    print('\n')

```

Setiap kolom diinformasikan (tercetak) kembali dengan menggunakan data yang telah di transformasikan apakah sudah stasioner atau belum.

Valid

```

6 # Mencetak nilai lag order untuk mencari hasil optimal
model_lag_order = VAR(data_nor_diff)
for i in range(1,7):
    hasil_lag_order = model_lag_order.fit(i)
    print('Lag Order ke-', i)
    print('AIC =', hasil_lag_order.aic, '\n')

```

Hasil *lag order* AIC berdasarkan data yang telah stasioner tercetak.

Valid

```

7 # Melatih model VAR berdasarkan nilai AIC yang dipilih
hasil_model_var = model_lag_order.fit(3)

```

Data pemodelan dilatih menggunakan algoritma *Vector Autoregression* berdasarkan nilai AIC yang ditentukan.

Valid


```

8 # Proses prediksi menggunakan algoritma VAR
# Mendapatkan nilai lag order
lag_order = hasil_model_var.k_ar

# Input data untuk prediksi menggunakan algoritma VAR
input_prediksi_var = data_nor_diff.values[-lag_order:]

# Prediksi
prediksi_var = hasil_model_var.forecast(y=input_prediksi_var, steps=range_date)

# Mencetak hasil prediksi menggunakan algoritma VAR
# Prediksi pada skala data latih yang digunakan oleh model
# Untuk melihat angka asli dilakukan de-difference sebanyak 1 kali
date_forecast = pd.date_range('2021/07/01', freq="B", periods=len(prediksi_var))
out_prediksi_var = pd.DataFrame(prediksi_var, columns=data_nor.columns + '_1d')

```

Pemodelan prediksi harga batubara Newcastle berjangka dihasilkan berdasarkan hasil pelatihan model dan tercetak dengan rentang tanggal yang ditetapkan setiap hari Senin – Jumat.

Valid

```

9 # Mengembalikan transformasi data untuk mendapatkan hasil prediksi yang sebenarnya
def invert_transformation(data_nor_train, out_prediksi_var):
    data_tf = out_prediksi_var.copy()
    kolom_data_tf = data_nor_train.columns
    for col in kolom_data_tf:
        # Mengembalikan transformasi data pertama
        data_tf[str(col) + ' (d+1)_VAR'] = data_nor_train[col].iloc[-1] + data_tf[str(col) + '_1d'].cumsum()
    return data_tf

# Menyimpan hasil prediksi pada sebuah variabel
hasil_tf = invert_transformation(data_nor_train, out_prediksi_var)

```

Data hasil prediksi yang sebelumnya ditransformasikan akan dikembalikan transformasinya dan disimpan ke dalam suatu variabel.

Valid

```

10 # Denormalisasi hasil prediksi
min_var_h = clean_kolom_dataset_original['High'].min()
max_var_h = clean_kolom_dataset_original['High'].max()
min_var_l = clean_kolom_dataset_original['Low'].min()
max_var_l = clean_kolom_dataset_original['Low'].max()
drn_prediksi_var_h = (hasil_tf[['High (d+1)_VAR']] * (max_var_h - min_var_h)) + min_var_h
drn_prediksi_var_l = (hasil_tf[['Low (d+1)_VAR']] * (max_var_l - min_var_l)) + min_var_l

```

Data di denormalisasi untuk memperoleh nilai nyata.

Valid

```

11 # Menggabungkan kolom prediksi High dengan Low
merge_column_var = pd.concat([drn_prediksi_var_h, drn_prediksi_var_l], axis=1)

# Menghapus Index pada merge_column_var
hasil_reset_index_var = merge_column_var.reset_index()
hasil_prediksi_var_hl = hasil_reset_index_var[['High (d+1)_VAR', 'Low (d+1)_VAR']]

# Menggabungkan kolom
hasil_prediksi_var_hl['Date'] = pd.date_range('2021/07/01', periods=len(merge_column_var), freq='B')
hasil_prediksi_var = hasil_prediksi_var_hl.set_index('Date')
print("Data Prediksi Harga Batubara Newcastle Berjangka menggunakan Algoritma VAR")
print(hasil_prediksi_var)

# Membuat grafik prediksi harga batubara Newcastle berjangka
judul_grafik_prediksi_var = "Grafik Prediksi Harga "High (d+1) dan Low (d+1)" Batubara Newcastle Berjangka - VAR"
label_grafik_prediksi_var = "Harga (satuan USD)"
periode_grafik_prediksi_var = "Periode 01-Jul-2021 sampai 23-November-2021"
hasil_prediksi_var.plot(figsize=(10,5), legend=True, title=judul_grafik_prediksi_var, grid=True,
                        xlabel=xlabel_grafik_prediksi_var, ylabel=ylabel_grafik_prediksi_var);

```

Pemodelan prediksi pergerakan harga (*High (d+1)_VAR* dan *Low (d+1)_VAR*) batubara Newcastle berjangka menggunakan algoritma *Vector Autoregression* dikeluarkan dalam bentuk tabel dan grafik.

Valid

```

12 # Grafik Prediksi Harga Batubara Newcastle Berjangka - VAR
xlabel_grafik_prediksi_var_1 = 'Periode 01-Juli-2021 sampai 05-Agustus-2021'
xlabel_grafik_prediksi_var_2 = 'Periode 06-Agustus-2021 sampai 10-September-2021'
xlabel_grafik_prediksi_var_3 = 'Periode 13-September-2021 sampai 18-Oktober-2021'
xlabel_grafik_prediksi_var_4 = 'Periode 19-Oktober-2021 sampai 23-November-2021'

# Periode 01-Juli-2021 sampai 05-Agustus-2021
hasil_prediksi_var.iloc[:26,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_var,
xlabel=xlabel_grafik_prediksi_var_1,
ylabel=ylabel_grafik_prediksi_var)

# Periode 06-Agustus-2021 sampai 10-September-2021
hasil_prediksi_var.iloc[26:52,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_var,
xlabel=xlabel_grafik_prediksi_var_2,
ylabel=ylabel_grafik_prediksi_var)

# Periode 13-September-2021 sampai 18-Oktober-2021
hasil_prediksi_var.iloc[52:78,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_var,
xlabel=xlabel_grafik_prediksi_var_3,
ylabel=ylabel_grafik_prediksi_var)

# Periode 19-Oktober-2021 sampai 23-November-2021
hasil_prediksi_var.iloc[78:104,:].plot(figsize=(10,5),legend=True, grid=True,
title=judul_grafik_prediksi_var,
xlabel=xlabel_grafik_prediksi_var_4,
ylabel=ylabel_grafik_prediksi_var)

```

Menampilkan Valid grafik prediksi harga batubara Newcastle berjangka berdasarkan periode yang telah ditentukan.

Berdasarkan tabel 5.9 di atas, seluruh baris kode pada pengujian pemodelan prediksi pergerakan harga batubara Newcastle berjangka memberikan kesimpulan terbaik seperti hasil yang diharapkan oleh peneliti. Dengan demikian, algoritma *Vector Autoregression* telah teruji dan dapat diperiksa akurasi menggunakan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE).

5.2.3. Evaluasi Hasil Algoritma *K-Nearest Neighbors* dengan *Vector Autoregression*

Evaluasi hasil algoritma merupakan tujuan dari dilakukannya penelitian ini. Hasil pemodelan prediksi pergerakan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression* akan diperiksa akurasi dengan melakukan perhitungan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Hasil RMSE dan MAPE dari kedua algoritma tersebut akan dibandingkan nilainya untuk mengetahui algoritma mana yang akurat dalam melakukan pemodelan prediksi pada topik penelitian ini. Adapun baris kode yang digunakan untuk menghitung nilai RMSE dan MAPE pada algoritma *K-Nearest Neighbors* tertulis pada gambar berikut ini.

```

# Menguji skor prediksi data pada algoritma KNN menggunakan RMSE
tes_prediksi_mse = knn_model.predict(test_x)
rmse_knn = math.sqrt(mse(tes_prediksi_mse, test_y))

# Menguji skor prediksi data pada algoritma KNN menggunakan MAPE
mape_knn = mape(test_y, tes_prediksi_mse)

# Mencetak hasil skor akurasi pada algoritma KNN
print('Nilai Root Mean Squared Error (RMSE) ~ KNN =', rmse_knn)
print('Nilai Mean Absolute Percentage Error (MAPE) ~ KNN =', mape_knn*100, '%')

```

Gambar 5.39 Baris Kode Perhitungan RMSE dan MAPE Algoritma KNN

Berdasarkan gambar 5.39 di atas, perhitungan *Root Mean Squared Error* (RMSE) pada algoritma *K-Nearest Neighbors* dilakukan dengan menggunakan rumus akar kuadrat dari *Mean Squared Error* (MSE). Logika yang dijalankan yaitu hasil pengurangan dari data asli dengan data prediksi dan dikuadratkan, lalu dibagi dengan banyaknya data dan hasilnya di akar kuadratkan. Selanjutnya pada perhitungan *Mean Absolute Percentage Error* (MAPE) dilakukan dengan menjumlahkan setiap baris data sebenarnya dengan data prediksi yang dibagi dengan data sebenarnya dan hasilnya di rata-ratakan. Penggunaan rumus yang sama, algoritma *Vector Autoregression* bekerja pada baris kode yang serupa dengan *K-Nearest Neighbors*. Adapun baris kode perhitungan RMSE dan MAPE pada algoritma tersebut tertulis pada gambar berikut ini.

```

# Menginisialisasi kolom ke suatu variabel
hasil_tf_acc = hasil_tf[['High (d+1)_VAR', 'Low (d+1)_VAR']]
data_nor_test_acc = data_nor_test[['High (d+1)', 'Low (d+1)']]

# Menguji skor prediksi data pada algoritma VAR menggunakan RMSE
rmse_var = math.sqrt(mse(hasil_tf_acc, data_nor_test_acc))

# Menguji skor prediksi data pada algoritma VAR menggunakan MAPE
mape_var = mape(data_nor_test_acc, hasil_tf_acc)

# Mencetak hasil skor akurasi pada algoritma VAR
print('Nilai Root Mean Squared Error (RMSE) ~ VAR =', rmse_var)
print('Nilai Mean Absolute Percentage Error (MAPE) ~ VAR =', mape_var*100, '%')

```

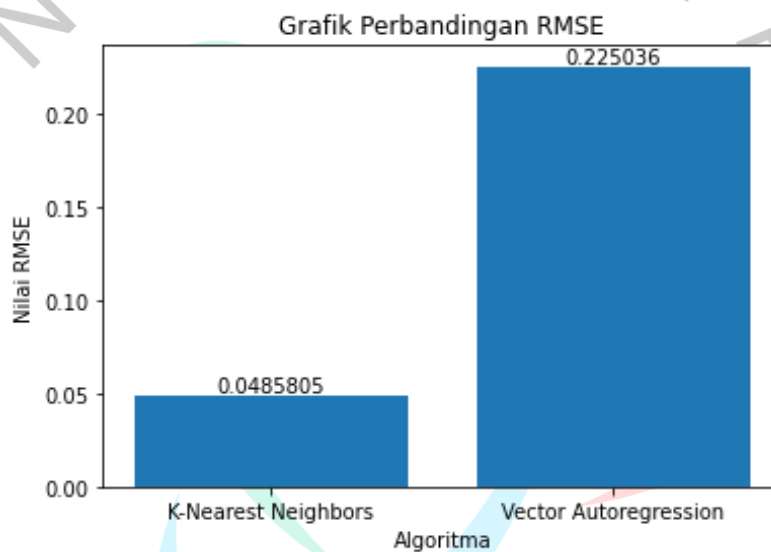
Gambar 5.40 Baris Kode Perhitungan RMSE dan MAPE Algoritma VAR

Berdasarkan gambar 5.40 di atas, RMSE dan MAPE pada algoritma *Vector Autoregression* diuji masing-masing pada setiap kolom data yang disatukan pada suatu variabel. Dengan demikian hasil RMSE dan MAPE pada prediksi pemodelan pergerakan harga batubara Newcastle berjangka dapat dihasilkan untuk

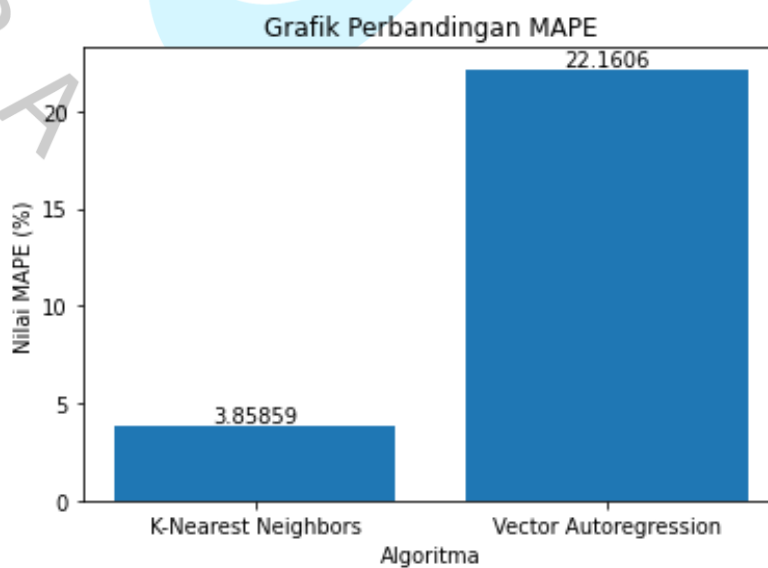
dibandingkan dengan algoritma pembandingnya. Adapun evaluasi hasil berdasarkan nilai *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE) pada algoritma *machine learning* yang digunakan tersaji pada tabel dan grafik berikut ini.

Tabel 5.10 Perbandingan Hasil RMSE dan MAPE

Algoritma	Nilai RMSE	Nilai MAPE
<i>K-Nearest Neighbors</i>	0.04858	3.85859 %
<i>Vector Autoregression</i>	0.22504	22.16059 %



Gambar 5.41 Grafik Perbandingan Hasil RMSE



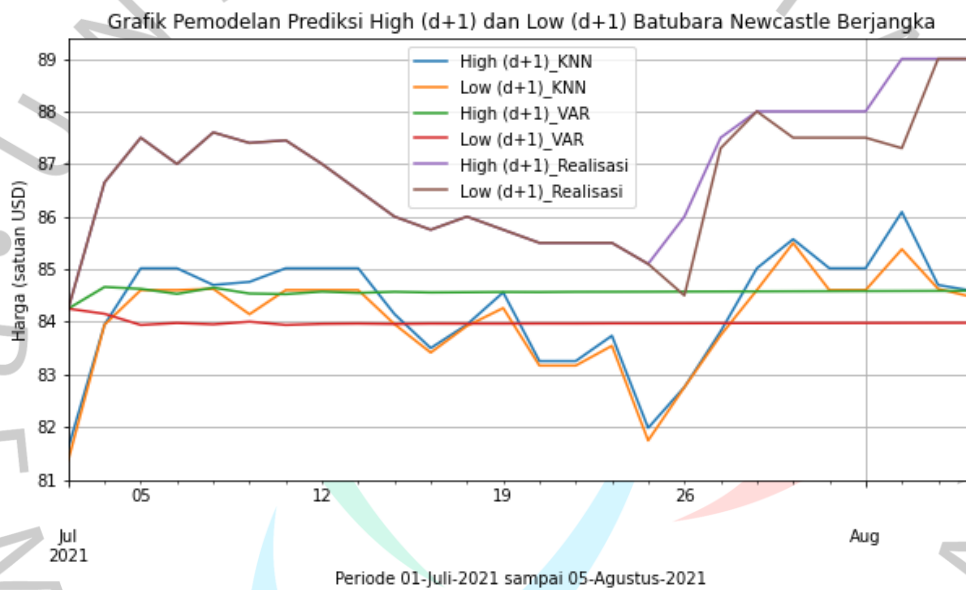
Gambar 5.42 Grafik Perbandingan Hasil MAPE

Algoritma dapat dikatakan akurat jika memiliki nilai RMSE dan MAPE terendah daripada algoritma lainnya yang digunakan pada suatu penelitian. Berdasarkan nilai RMSE yang dihasilkan pada tabel 5.10 serta grafik 5.41 dan 5.42 di atas, terlihat bahwa algoritma *K-Nearest Neighbors* memiliki nilai RMSE yang lebih rendah daripada algoritma *Vector Autoregression*. Perbandingan nilai yang dihasilkan mempunyai selisih 0.17645. Sedangkan nilai MAPE yang dihasilkan pada algoritma *K-Nearest Neighbors* memiliki hasil prediksi sangat akurat dan pada *Vector Autoregression* memberikan hasil prediksi cukup akurat. Tidak ada nilai yang salah pada pemodelan prediksi harga batubara Newcastle berjangka ini dikarenakan algoritma bekerja sesuai dengan logika masing-masing dengan data yang digunakan.

5.2.4. Perbandingan Data Realisasi dengan Hasil Pemodelan

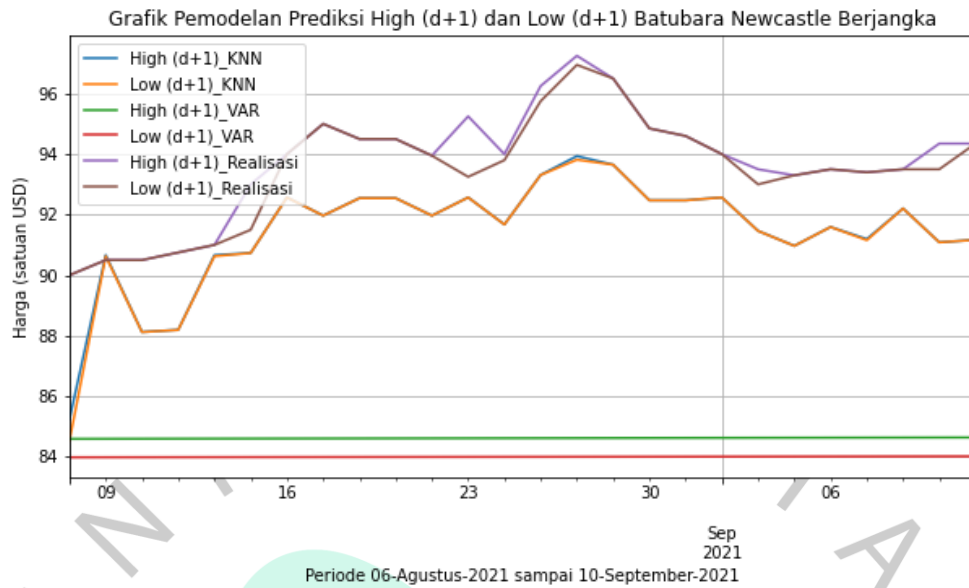
- Hasil dari pemodelan pergerakan harga batubara Newcastle berjangka menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression* dibandingkan dengan harga asli atau data realisasinya untuk dapat melihat perbedaannya. Data realisasi diperoleh setelah data harga dikeluarkan oleh situs investing.com sebagai laman referensi pada penelitian ini. Pada situs tersebut, data dikeluarkan dalam bentuk tabel harga dan grafik. Berdasarkan data tersebut, setiap investor dapat berdiskusi mengenai prediksi yang dianalisis secara individu melalui Forum. Analisis tersebut tidak dipastikan dapat terealisasi di kemudian hari karena hal tersebut hanya berdasarkan penglihatan dari setiap individu yang memiliki pandangan berbeda-beda. Oleh karena itu, data hasil analisis individu secara manual tidak terekam dengan baik dan tingkat keakuratannya tidak dapat dijelaskan oleh data. Pada situs investing.com yang dijadikan sebagai acuan pada penelitian ini hanya memberikan prediksi berupa petunjuk untuk beli atau jual saat hari itu, namun tidak menyajikan perkiraan harga yang akan dikeluarkan. Berbeda saat melaksanakan pemodelan dengan *machine learning*, data prediksi dapat terekam dengan baik dan harga batubara Newcastle berjangka dapat diperkirakan. Dengan demikian, hasil pemodelan pergerakan harga batubara Newcastle berjangka dengan menggunakan algoritma *machine learning* dibandingkan dengan data realisasinya.

Pada hasil penelitian ini disajikan perbandingan data realisasi dengan hasil pemodelan menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression*. Grafik disajikan pada perbandingan antara *High (d+1)* dengan *Low (d+1)* yang tersaji dengan interval bulanan, serta dipotong menjadi empat periode untuk mempermudah dalam melihat perbandingan dengan interval 5 hari kerja. Pada setiap grafik terdapat keterangan yang membedakan antara hasil realisasi dan dengan menggunakan algoritma *K-Nearest Neighbors* (KNN) dan *Vector Autoregression* (VAR). Adapun perbandingan data realisasi dengan hasil pemodelan tersebut disajikan pada grafik di bawah ini.



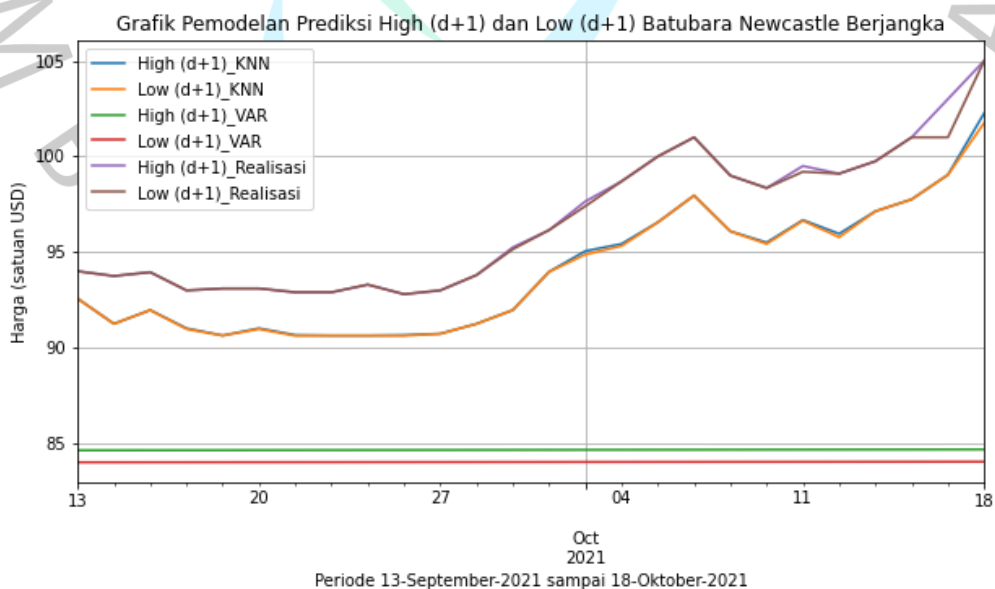
Gambar 5.43 Grafik Perbandingan Periode 1 Juli - 5 Agustus 2021

Berdasarkan gambar 5.43 di atas, pemodelan prediksi harga *High (d+1)* dan *Low (d+1)* yang dihasilkan saat menggunakan algoritma *K-Nearest Neighbors* pada periode 1 Juli – 5 Agustus 2021 menyerupai data realisasinya. Hal tersebut dapat terlihat saat data realisasi menunjukkan kenaikan harga, maka model prediksi yang dihasilkan dari algoritma tersebut juga mengalami kenaikan. Berbeda saat menggunakan algoritma *Vector Autoregression* dimana model yang dihasilkan mengalami perubahan di beberapa hari pertama dan konstan di harga \$84 pada hari setelahnya.



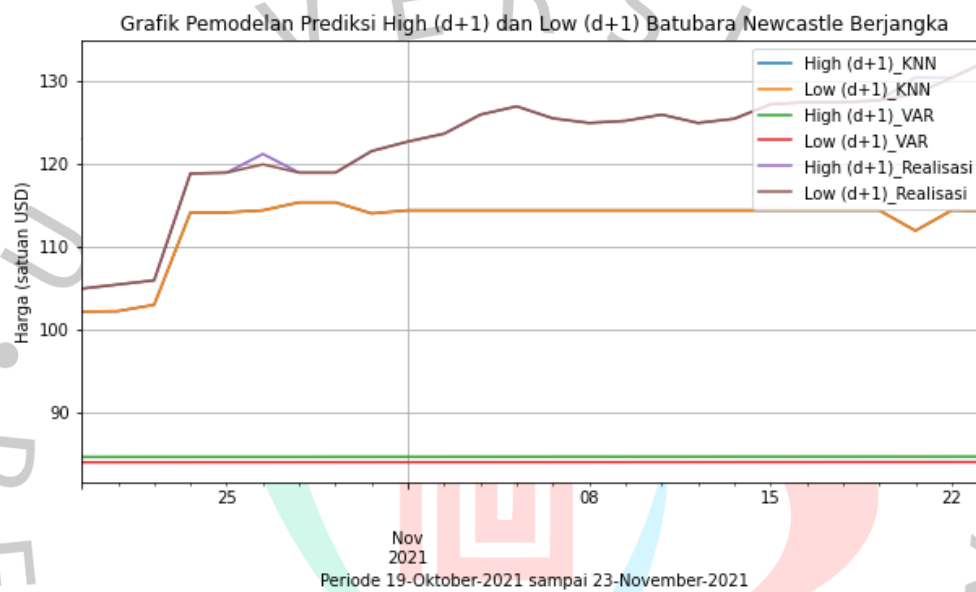
Gambar 5.44 Grafik Perbandingan Periode 6 Agustus - 10 September 2021

Berdasarkan gambar 5.44 di atas, pemodelan prediksi harga *High (d+1)* dan *Low (d+1)* yang dihasilkan saat menggunakan algoritma *K-Nearest Neighbors* pada periode 6 Agustus – 10 September 2021 melebihi harga realisasinya pada 9 Agustus 2021 dan pergerakan setelahnya menyerupai data realisasinya. Berbeda saat menggunakan algoritma *Vector Autoregression* dimana model yang dihasilkan konstan di harga \$84.



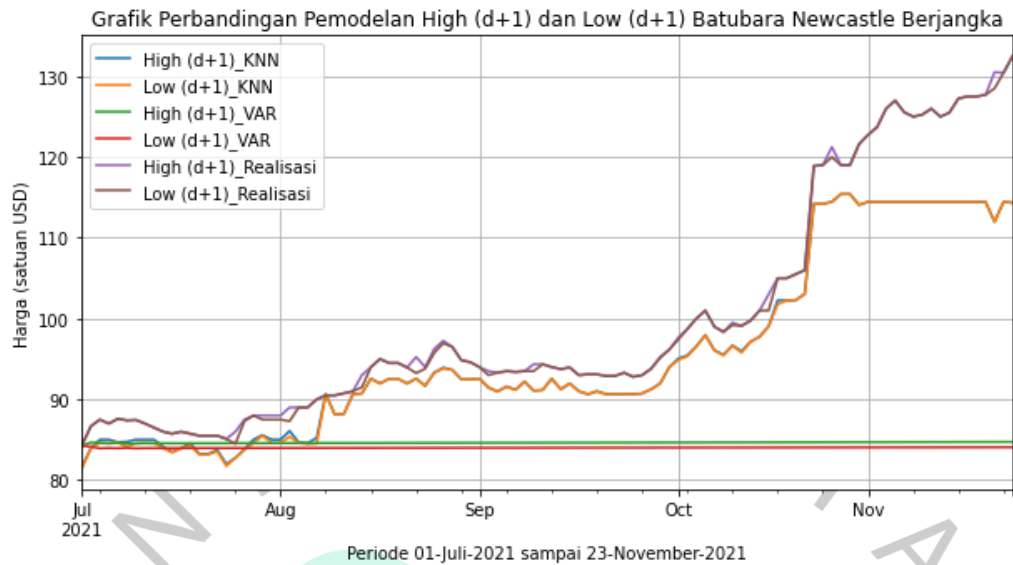
Gambar 5.45 Grafik Perbandingan Periode 13 September - 18 Oktober 2021

Berdasarkan gambar 5.45 di atas, pemodelan prediksi harga $High(d+1)$ dan $Low(d+1)$ yang dihasilkan saat menggunakan algoritma K -Nearest Neighbors pada periode 13 September – 18 Oktober 2021 menyerupai data realisasinya. Hal tersebut dapat terlihat saat harga realisasi naik, maka data prediksi memberikan kenaikan harga walaupun terdapat selisih angka. Berbeda saat menggunakan algoritma $Vector Autoregression$ dimana model yang dihasilkan tetap konstan di harga \$84.



Gambar 5.46 Grafik Perbandingan Periode 19 Oktober - 23 November 2021

Berdasarkan gambar 5.46 di atas, pemodelan prediksi harga $High(d+1)$ dan $Low(d+1)$ yang dihasilkan saat menggunakan algoritma K -Nearest Neighbors pada periode 19 Oktober – 23 November 2021 menyerupai data realisasinya saat di awal periode dan mulai konstan saat awal November 2021. Sedangkan pada model prediksi menggunakan algoritma $Vector Autoregression$ memberikan hasil yang konstan.



Gambar 5.47 Grafik Perbandingan Periode 1 Juli - 23 November 2021

Berdasarkan grafik pada gambar 5.47 di atas, dapat dilihat bahwa data realisasi dengan hasil pemodelan periode 1 Juli – 23 November 2021 menggunakan algoritma *machine learning* mengalami perbedaan harga. Pemodelan menggunakan algoritma *K-Nearest Neighbors* memberikan hasil yang tidak berbeda jauh dengan data realisasinya. Namun, pada algoritma *Vector Autoregression* memiliki hasil pemodelan yang cukup jauh dari data realisasi. Hal tersebut merupakan hal yang dapat terjadi saat melakukan pemodelan dalam memprediksi suatu harga berdasarkan data historis sebelumnya. Oleh karena itu, hasil pemodelan menggunakan algoritma *K-Nearest Neighbors* dan *Vector Autoregression* dievaluasi menggunakan *Root Mean Squared Error (RMSE)* dan *Mean Absolute Percentage Error (MAPE)* untuk dapat melihat seberapa besar tingkat *error* yang dihasilkan.