

LAPORAN
DOKUMENTASI PROJECT AKHIR



DISUSUN OLEH :

NAMA : CITRA PRATAMA

NIM : 200102002

INSTITUT TEKNOLOGI & BISNIS STIKOM AMBON

PROGRAM STUDI S-1 SISTEM INFORMASI

TAHUN AKADEMIK 2024/2025



INSTITUT TEKNOLOGI DAN BISNIS STIKOM AMBON

Prodi :Teknik Informatika, Sistem Informasi, Pariwisata, Bisnis Digital, Desain Komunikasi Visual,

Perencanaan Wilayah dan Kota Manajemen Informatika, Komputerisasi Akuntansi

Jl. A. Y. Patty No. 108, Ambon, Maluku. Telp. (0911) 315735, 342332 Fax. (0911) 316117,

Web :www.stikomambon.ac.id, Email : info@stikomambon.ac.id, akademik.stikomambon@gmail.com

SURAT PERNYATAAN

Ditulis menggunakan huruf capital :

NAMA (SESUAI IJAZAH SMA):

CITRA PRATAMA

TEMPAT DAN TANGGAL LAHIR(SESUAI IJAZAH SMA):

KULATI, 23 SEPTEMBER 2001

JENIS KELAMIN :

LAKI-LAKI

PROGRAM STUDI :

SISTEM INFORMASI

JUDUL PROJECT AKHIR atau TUGAS AKHIR(tidak boleh memakai singkatan):

PENGELOMOKKAN HARGA SAHAM BANK RAKYAT
INDONESIA (BRI) MENGGUNAKAN METODE AGGLOMERATIVE
HIERARCHICAL CLUSTERING

NO HP : 082398177106

Biodata diisi dengan sebenar-benarnya, jika terjadi kesalahan saya akan bertanggung jawab

Ambon, 19 Agustus 2024



KATA PENGANTAR

Puji serta syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas campur tangan-Nya sehingga penyusunan Laporan Project Akhir ini dapat terselesaikan dengan baik. Laporan ini sebagai bukti bahwa penulis telah menyelesaikan Project Akhir dengan baik.

Selain itu penyusunan laporan Project Akhir ini juga adalah salah satu syarat untuk mengikuti ujian tugas akhir.

Tidak lupa saya mengucapkan banyak terima kasih kepada kedua orang tua saya dan teman - teman yang selama ini telah memberikan dukungan dan support sehingga bisa sampai pada tahap ini.

saya juga mengharapkan saran dan kritik demi perbaikan dan penyempurnaan laporan ini. Semoga laporan ini dapat bermanfaat bagi pembaca atau siapa saja yang membutuhkannya.

Ambon, 19 Agustus 2024

Penyusun



CITRA PRATAMA

DAFTAR ISI

Cover/Halaman Judul

Surat Pernyataan	I
Kata Pengantar	II
Daftar Isi	iii
Daftar Tabel.....	iv
Daftar Gambar.....	v
Bab I Pendahuluan	1
1. Latar Belakang	1
2. Rumusan Masalah.....	2
3. Tujuan	2
4. Manfaat	3
Bab II Perancangan	4
1. Pengumoulan Data	4
2. Preprocessing	6
3. Agglomerative Hierarchical Clustering	9
4. Perhitungan Jarak Euclidean Distnce.....	9
5. Pembentukan Dendrogram.....	12
6. Evaluai Shilhouette Score	13
7. Perangkat Lunak atau Software	13
Bab III Implementasi.....	15
1. Dashboard	15
2. Pemilihann Fitur Berdasarkan Karakteristik Dataset.....	15
3. Memunculkan Dataset	16
4. Fitur Preprocessing dan Pilihan Parameter Clustering	17
5. Visualisasi Dendrogram	18
6. Visualisasi Hasil Clustering	19
Bab IV Hasil dan Kesimpulan	21
1. Hasil Clustering (Complate Linkage)	21
2. Hasil Visualisasi Dendrogram dan Plot Clustering.....	21
3. Kesimpulan	23
4. Saran	23
Lampiran	24
Daftar Pustaka	30

DAFTAR TABEL

Tabel 2.1 Data Set	6
Tabel 2.2 Preprocessing	8
Tabel 2.3 Matrix Perhitungan Jarak	11
Tabel 2.4 Hasil Matrix Jarak	11
Tabel 2.5 Hasil Matrix Jarak Lanjutan.....	12
Tabel 2.6 Hasil Matrix Jarak Akhir.....	12
Tabel 2.7 Interpretasi Nilai Silhouette Score	13
Tabel 2.8 Hasil Clustering.....	21

DAFTAR GAMBAR

Gambar 1.1 Skema Artificial Intelligence dan Machine Learning	1
Gambar 2.1 Proses Alogaritma AHC	9
Gambar 2.2 Ilustrasi Perhitungan Jarak dengan Metode Linkage	10
Gambar 2.3 Hasil Dendrogram Matrix	13
Gambar 3.1 Dashboard.....	15
Gambar 3.2 Pemilihan Fitur.....	15
Gambar 3.3 Data Set.....	16
Gambar 3.4 Proses Preprocessing dan Pilihan Parameter Clustering.....	17
Gambar 3.5 Visualisasi Dendrogram	18
Gambar 3.6 Visualisasi Hasil Clustering	19
Gambar 3.7 Hasil Visualisasi Dendrogram.....	21
Gambar 3.8 Hasil Visualisasi Plot Clustering.....	22

BAB I

PENDAHULUAN

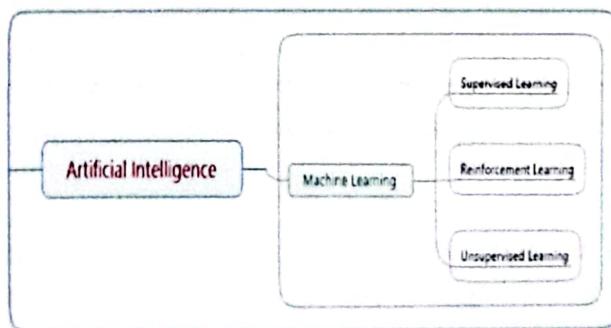
1. Latar Belakang

Analisis terhadap harga saham menjadi sangat penting bagi investor untuk membuat keputusan investasi yang tepat. Mengingat volatilitas dan kompleksitas pergerakan harga saham diperlukan pendekatan analitis yang efektif untuk memahami dan mengelompokkan data harga saham berdasarkan karakteristik pergerakannya.

Kecerdasan Buatan merupakan salah satu bidang dalam ilmu komputer yang ditujukan pada pembuatan software dan hardware yang dapat berfungsi sebagai sesuatu yang dapat berpikir seperti manusia (Sunarya, Santoso, & Sentanu, 2015). Kecerdasan buatan banyak digunakan untuk memecahkan berbagai masalah seperti bisnis (Rahardja, Roihan, & others, 2017), robotika, bahasa alami, matematika, game, persepsi, diagnosis medis, teknik, analisis keuangan, analisis sains, dan penalaran (Russell & Norvig, 2016).

Machine learning dapat didefinisikan sebagai aplikasi komputer dan algoritma matematika yang diadopsi dengan cara pembelajaran yang berasal dari data dan menghasilkan prediksi di masa yang akan datang (Goldberg & Holland, 1988). Adapun proses pembelajaran yang dimaksud adalah suatu usaha dalam memperoleh kecerdasan yang melalui dua tahap antara lain latihan (training) dan pengujian (testing) (Huang, Zhu, & Siew, 2006). Bidang machine learning berkaitan dengan pertanyaan tentang bagaimana membangun program komputer agar meningkat secara otomatis dengan berdasar dari pengalaman (Mitchell, 1997).

Penelitian terkini mengungkapkan bahwa machine learning terbagi menjadi tiga kategori: Supervised Learning, Unsupervised Learning, Reinforcement Learning (Somvanshi & Chavan, 2016). Skema keterkaitan artificial intelligence dan machine learning dapat dijelaskan dalam Gambar 1.



Gambar 1.1 Skema Artificial Intelligence dan Machine Learning

Teknik yang digunakan oleh Supervised Learning adalah metode klasifikasi di mana kumpulan data sepenuhnya diberikan label untuk mengklasifikasikan kelas yang tidak dikenal. Sedangkan teknik Unsupervised Learning sering disebut cluster dikarenakan tidak ada kebutuhan untuk pemberian label dalam kumpulan data dan hasilnya tidak mengidentifikasi contoh di kelas yang telah ditentukan (Thupae, Isong,

Gasela, & AbuMahfouz, 2018). Sedangkan Reinforcement Learning biasanya berada antara Supervised Learning dan Unsupervised Learning (Board, 2017), teknik ini bekerja dalam lingkungan yang dinamis di mana konsepnya harus menyelesaikan tujuan tanpa adanya pemberitahuan dari komputer secara eksplisit jika tujuan tersebut telah tercapai (Das & Nene, 2017).

Dalam jenis pembelajaran Unsupervised Learning, sistem disediakan dengan beberapa input sampel tetapi tidak ada output yang hadir. Karena tidak ada output yang diinginkan di sini kategorisasi dilakukan sehingga algoritma membedakan dengan benar antara kumpulan data. Ini adalah tugas mendefinisikan fungsi untuk menggambarkan struktur yang tersembunyi dari data yang tidak berlabel (Somvanshi & Chavan, 2016). Unsupervised learning dikelompokkan lebih lanjut dalam masalah clustering dan asosiasi. Masalah pengelompokan (clustering) adalah tempat untuk menemukan pengelompokan yang melekat dalam data, seperti mengelompokkan pelanggan berdasarkan pada perilaku pembelian. Sedangkan masalah asosiasi adalah aturan yang menggambarkan sebagian besar data yang ada, seperti orang yang membeli A juga cenderung membeli B (Brownlee, 2016).

Unsupervised learning memiliki beberapa algoritma populer seperti k-means, Apriori (Brownlee, 2016), Independent Subspace Analysis (ISA) (G. Wu et al., 2013), DBSCAN (Ester, Kriegel, Sander, Xu, & others, 1996). Beberapa masalah misalnya dalam bidang finansial untuk meninjau sejumlah besar data maka unsupervised learning biasanya dapat digunakan (Board, 2017). Dalam bidang industri misalnya dalam (Aji & Wibisono, 2018), kemudian dalam bidang kedokteran unsupervised learning digunakan dalam proses segmentasi pembuluh darah (Dharmawan, Ng, & Rahardja, 2018), dan bidang teknologi seperti jaringan komputer maupun pencegahan serangan keamanannya (Das & Nene, 2017) pun menggunakan kategori ini.

2. Rumusan Masalah

1. Bagaimana pola pengelompokan saham Bank Mandiri dapat membantu dalam prediksi pergerakan harga saham?
2. Seberapa efektif algoritma Agglomerative Hierarchical Clustering dalam mengelompokkan harga saham berdasarkan karakteristik pergerakannya?
3. Bagaimana mengevaluasi hasil pengelompokan harga saham Bank Mandiri yang dihasilkan oleh algoritma Agglomerative Hierarchical Clustering?

3. Tujuan

Tujuan dari project ini adalah untuk mengelompokkan harga saham Bank BRI berdasarkan histori stock price menggunakan algoritma Agglomerative Hierarchical Clustering dengan dikelompokkannya harga saham, diharapkan dapat memperoleh wawasan mengenai pola pergerakan harga saham yang serupa dalam periode tertentu dan membantu investor dalam membuat keputusan investasi yang lebih baik.

4. Manfaat

1. Memberikan wawasan dan pengetahuan mengenai pengelompokan harga saham sehingga memperoleh gambaran yang jelas.
2. Pengambilan keputusan: Membantu investor dalam pengambilan keputusan berdasarkan kelompok harga saham dengan karakteristik yang mirip.
3. Strategi Investasi: Mengembangkan strategi investasi yang lebih baik berdasarkan pengelompokan harga saham.

BAB II

PERANCANGAN

1. Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan history stock price bank BRI pada periode tertentu sejumlah 105 baris data yang diperoleh dari website BANK RAKYAT INDONESIA dengan menggunakan smartphone pada tahun 2024. Terdapat 7 variabel yang ada pada dataset, namun pada penelitian ini hanya menggunakan 2 variabel yaitu data Close dan Volume. Kedua variabel ini digunakan sebagai variable penentu karena pengelompokan harga saham berdasarkan pada karakteristik pergerakannya.

Date	Open	High	Low	Close	Volume	AdjustedClose
20240102	5650	5675	5625	5675	91143100	5675
20240103	5625	5650	5600	5600	83659700	5600
20240104	5600	5700	5575	5700	118467900	5700
20240105	5725	5850	5700	5750	134929600	5750
20240108	5750	5800	5625	5625	152015000	5625
20240109	5625	5700	5625	5700	105666200	5700
20240110	5625	5700	5625	5700	82626400	5700
20240111	5750	5800	5700	5750	132206400	5750
20240112	5800	5850	5775	5850	109715700	5850
20240115	5850	5900	5775	5825	115284200	5825
20240116	5825	5875	5775	5825	97074300	5825
20240117	5825	5825	5750	5775	107941600	5775
20240118	5800	5800	5700	5750	87259100	5750
20240119	5750	5800	5700	5800	75370700	5800
20240122	5775	5800	5700	5775	97545600	5775
20240123	5700	5750	5650	5700	122999700	5700
20240124	5675	5700	5575	5650	104225000	5650
20240125	5575	5650	5525	5525	120478800	5525
20240126	5450	5475	5400	5425	185244400	5425
20240129	5400	5575	5400	5575	145629600	5575
20240130	5525	5650	5475	5625	150287800	5625
20240131	5725	5775	5675	5700	245117200	5700
20240201	5775	5800	5725	5750	156634200	5750
20240202	5800	5850	5750	5850	177209700	5850
20240205	5800	5825	5750	5775	113760000	5775
20240206	5900	5900	5800	5825	126704300	5825
20240207	5850	5900	5825	5850	147571500	5850
20240212	5875	6025	5850	6025	200856200	6025
20240213	6050	6050	5950	6000	145852200	6000
20240215	6275	6275	6100	6125	308685100	6125
20240216	6150	6200	6125	6150	130814400	6150
20240219	6125	6175	6075	6100	111382900	6100
20240220	6175	6300	6175	6300	160872600	6300
20240221	6325	6375	6225	6300	169293100	6300

20240222	6250	6300	6225	6250	100256000	6250
20240223	6125	6175	6050	6125	148643000	6125
20240226	6100	6200	6075	6175	110509000	6175
20240227	6125	6200	6125	6125	99187500	6125
20240228	6100	6225	6100	6225	118745300	6225
20240229	6150	6200	6125	6125	173845800	6125
20240301	6175	6175	6075	6125	76286400	6125
20240304	6125	6125	6050	6050	70354900	6050
20240305	6100	6175	6075	6125	115084600	6125
20240306	6100	6200	6100	6200	84108800	6200
20240307	6200	6275	6200	6225	117724700	6225
20240308	6275	6400	6250	6350	163060000	6350
20240313	6350	6450	6350	6400	195173100	6400
20240314	6175	6200	6050	6150	271254000	6150
20240315	6000	6075	5950	5975	305995800	5975
20240318	6000	6100	5975	6000	110564700	6000
20240319	6000	6050	5975	6000	114338600	6000
20240320	6000	6100	6000	6100	120885400	6100
20240321	6175	6200	6100	6100	136020000	6100
20240322	6050	6125	6050	6125	74936500	6125
20240325	6150	6250	6125	6250	115729600	6250
20240326	6200	6300	6200	6300	120748400	6300
20240327	6300	6325	6225	6250	73308900	6250
20240328	6100	6200	6025	6050	288926900	6050
20240401	6000	6025	5825	5925	217200400	5925
20240402	5850	5950	5625	5675	412180900	5675
20240403	5575	5675	5525	5625	333518700	5625
20240404	5650	5750	5650	5700	133406600	5700
20240405	5625	5700	5625	5650	178817800	5650
20240416	5375	5575	5350	5350	600717500	5350
20240417	5475	5500	5300	5300	244002000	5300
20240418	5300	5500	5300	5475	247410700	5475
20240419	5350	5375	5250	5275	258211500	5275
20240422	5200	5325	5150	5300	342282800	5300
20240423	5350	5400	5225	5300	283676400	5300
20240424	5250	5350	5150	5225	471241400	5225
20240425	5125	5200	5050	5150	479170900	5150
20240426	5050	5100	4830	4830	782589900	4830
20240429	4700	4890	4680	4770	700644800	4770
20240430	4850	5050	4820	4940	565609200	4940
20240502	4910	4920	4700	4760	636922100	4760
20240503	4760	4830	4710	4750	322179100	4750
20240506	4800	4840	4730	4790	262707400	4790
20240507	4770	4780	4670	4670	304013600	4670
20240508	4650	4740	4570	4680	572275800	4680
20240513	4650	4700	4630	4680	320067100	4680

20240514	4720	4810	4700	4730	341458900	4730
20240515	4780	4870	4760	4820	298990300	4820
20240516	4930	4960	4840	4840	371295000	4840
20240517	4840	4960	4820	4920	208991800	4920
20240520	4990	5050	4820	4820	335150700	4820
20240521	4810	4870	4680	4680	272599400	4680
20240522	4650	4770	4650	4720	215066000	4720
20240527	4700	4720	4550	4550	363719300	4550
20240528	4580	4630	4520	4530	258070200	4530
20240529	4480	4500	4400	4410	465008500	4410
20240530	4400	4480	4310	4380	502927800	4380
20240531	4430	4470	4330	4340	655144100	4340
20240603	4430	4560	4400	4530	345770100	4530
20240604	4560	4590	4450	4450	332902200	4450
20240605	4500	4500	4400	4400	246362500	4400
20240606	4450	4500	4390	4500	281534200	4500
20240607	4490	4500	4350	4350	375439600	4350
20240610	4350	4430	4320	4400	181387000	4400
20240611	4380	4440	4340	4340	197854400	4340
20240612	4340	4360	4310	4350	187332500	4350
20240613	4400	4420	4300	4310	238349600	4310
20240614	4300	4320	4170	4180	358893200	4180
20240619	4100	4240	4090	4100	410338400	4100
20240620	4130	4270	4130	4270	387133400	4270
20240621	4280	4490	4260	4440	542704300	4440
20240624	4440	4510	4400	4400	303830800	4400

Tabel 2.1 Data Set

2. Preprocessing

Tahapan preprocessing data dilakukan untuk membersihkan data sehingga data yang digunakan tidak mengalami duplicate data. Dalam perhitungan menggunakan algoritma AHC akan menggunakan variabel Close dan Volume. Dataset yang sudah melalui proses preprocessing tidak mengalami duplikasi data sehingga data yang digunakan. Pada tahap preprocessing menggunakan dua metode yaitu:

2.1 Normalization

Teknik prapemrosesan data lain yang berguna adalah Normalisasi. Digunakan untuk mengubah skala setiap baris data memiliki panjang 1 berguna dalam dataset Sparse di mana kita memiliki banyak nol. Kita dapat mengubah skala data dengan bantuan kelas Normalizer dari pustaka scikit-learn Perpustakaan Python.

2.2 Standard Scaler

Standard Scaler merupakan metode preprocessing di mana metode tersebut akan melakukan standarisasi fitur dengan menghapus rata-rata dan menskalakan unit varian. Proses ini akan dilakukan pada setiap fitur pada sampel. Preprocessing ini dilakukan untuk mencegah adanya data yang memiliki nilai terlalu besar dibanding dengan nilai yang lain yang akan dapat mengakibatkan proses training tidak sesuai keinginan.

No	Close	Volume
0	-0.6489261737362939	0.68265583733106
1	-0.5873568673054407	0.6512101596694264
2	-0.7535526119272262	0.7316705778087891
3	-0.8845367657423063	0.7851851767604642
4	-0.5771510004206716	0.6458114833105565
5	-0.7713424474519712	0.739450845930351
6	0.9871980947797834	-0.8080898409961369
7	-0.9854087752769466	0.8204495804723376
8	-0.7321083375950467	0.7220780589924208
9	1.0401725305311014	-0.8791115816344799
10	-0.5143087081808808	0.6114012835021561
11	2.272282319681192	-2.9336259390034143
12	-1.2979513305393036	0.8968556389467847
13	0.8972162268548594	-0.690724045114516
14	-0.691933049758367	0.703477490156913
15	-1.096788174275834	0.853375852593105
16	-1.1704578412838205	0.871687584359046
17	-0.5420263581691759	0.6268259265833888
18	0.9182780993177627	-0.7178265260416669
19	-1.0262745350648126	0.8332632800299128
20	-0.19304030341544504	0.4041047263094102
21	0.14080457377108324	0.13307645560830592
22	0.5145352464223232	-0.23758034832020913
23	-1.070265439130984	0.8461076196494616
24	2.475110524865898	-3.345849768931442
25	-0.962297283044372	0.812826999699008
26	0.8637190604343914	-0.6480846169459279
27	-1.0940271191992952	0.8526358758662783
28	0.6177822660586902	-0.352501349298749
29	0.9136128723936554	-0.7118037667468662
30	-0.8904031698767415	0.7873778001074697
31	0.15084129873064417	0.12405058563680495
32	-0.8187169375816025	0.7593847061655468
33	-0.5679130897679918	0.6408789461371929
34	0.7965418333644015	-0.5642926369215957
35	1.198618565142754	-1.1000559147161824
36	-1.146249528074678	0.8659746871135766
37	0.23323368038512066	0.04801951536852005
38	-0.6768847600425922	0.69629886977946
39	0.9509449401762637	-0.7603085232642149
40	0.9269276845394818	-0.7290220713639672
41	-0.9980880945544411	0.8245162474424811
42	-0.9663045350457112	0.8141682395666553
43	-1.110227970471495	0.8569224077162741
44	-0.5766064160873121	0.6455218770344129

45	1.8900508694170213	-2.2136547368246924
46	0.5627723928928522	-0.29059662961125093
47	-1.2326789147135828	0.8850031912079459
48	-0.22691627691653404	0.4284387045657674
49	-0.937078402809631	0.8041993223320676
50	1.2368569881468607	-1.1552898185679739
51	0.2436546447971357	0.03815700504371508
52	-0.6245983770604979	0.6704612208666666
53	0.4135278411555599	-0.13039846291137558
54	-0.8542009539705768	0.7735675934297241
55	-1.2662444392791565	0.8913688884348777
56	0.40820515098637444	-0.12489427705039576
57	0.5682278517304805	-0.2966670797138096
58	-0.322532488951619	0.49397382289329955
59	-0.4665003889327587	0.5838776619742384
60	2.044382666135598	-2.4954099778417644
61	-1.3066957587840449	0.8982790557170474
62	1.6647270277285589	0.45003063825946266
63	0.45003063825946266	-0.16853380196984571
64	-0.7859467359516752	0.7457177001121714
65	-0.11681044284467854	0.34721202847487265
66	0.9606864953165504	-0.7730822240797994
67	-0.5006138558073664	0.6036357585066483
68	0.2665441221414935	0.016300769372142875
69	-0.7607252638884743	0.7348268887953574
70	-0.8043065370465637	0.7534419443547273
71	2.2033662502125173	-2.79832455337188
72	1.3481089580735748	-1.320218221115296
73	-1.3282655770094614	0.9016234377116293
74	0.12683427429431787	0.1455543964951328
75	-0.9198408114905152	0.7981160521198964
76	0.16106778744618264	0.11480126115476018
77	0.7852270007165205	-0.5504052531839836
78	-1.2325180638833333	0.8849712691350111
79	0.33781429482853037	-0.053459216078149574
80	-0.8607324732018037	0.7761085391548982
81	0.5857814016242658	-0.3163023595961329
82	-0.5159020041055361	0.8798247643322167
83	-0.5159020041055361	0.6122983065717343

Tabel 2.2 Preprocessing

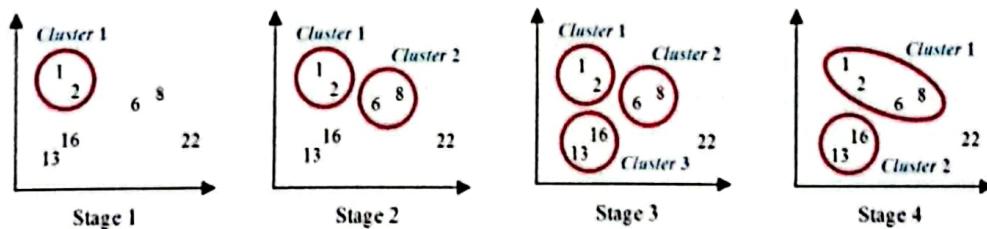
3. Agglomerative Hierarchical Clustering

Strategi untuk pengelompokan hirarki bottom up, dimulai dari masing-masing data sebagai kluster kemudian memungkinkan kluster yang semakin besar. Algoritma AHC Complate Linkage langkah-langkah algoritma AHC Complate Linkage secara umum :

1. Mencari jarak minimum dari dua objek.
2. Gabungkan dua objek dengan jarak minimum menjadi satu cluster.
3. Cari jarak antar cluster dengan mencari jarak terjauh.
4. Ulangi langkah 1-2 sampai semua objek bergabung menjadi satu cluster.

Tahapan selanjutnya adalah melakukan proses clustering. Hierarchical Clustering (HC) telah banyak digunakan untuk melakukan analisis cluster karena dapat memvisualisasikan struktur hierarki cluster dengan menggunakan dendrogram. Pada umumnya HC terbagi menjadi dua jenis, yaitu Agglomerative dan Devisve Namun, pada penelitian ini menggunakan metode Agglomerative Hierarchical Clustering (AHC), dikarenakan AHC memiliki karakteristik seperti hasil clustering yang dapat dilihat dengan dendrogram, tidak perlu menentukan jumlah cluster awal, dan menggunakan pendekatan (bottom-up) dimana clustering data dilakukan dari kecil ke besar.

Secara garis besar AHC awalnya menetapkan cluster untuk setiap objek, kemudian mencari pasangan cluster yang paling mirip dan menggabungkannya menjadi satu cluster. Proses ini dilakukan terus menerus hingga semua objek ditempatkan pada masing-masing cluster yang paling cocok . Seperti yang dapat dilihat pada Gambar 2 berikut:

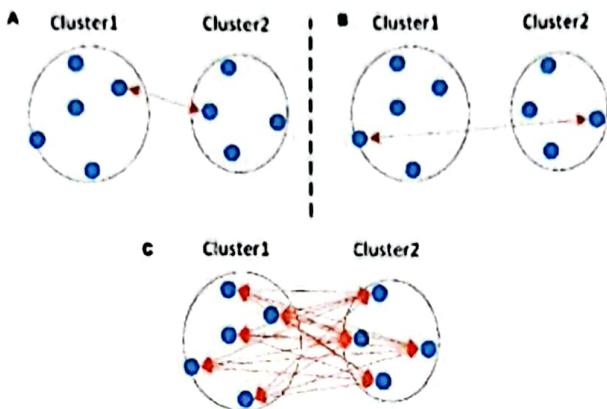


Gambar 2.1 Proses Alogaritma AHC

4. Perhitungan Jarak Euclidean Distace

Dalam penerapannya untuk menghitung jarak clustering dari algoritma agglomerative hierarchy clustering terbagi menjadi beberapa jenis. Single-linkage adalah pengelompokan hirarki dengan sistem kerja bottom-up, sistem kerja ini dimulai dari cluster. Ilustrasi dalam penerapannya, jika pendekatan single linkage digunakan maka kita menggabungkan kedua cluster A dan B tersebut jika jarak antara setiap item di A dan item di B adalah minimum, dalam ilustrasi ini pada awalnya semua cluster dianggap terpisah dan tidak dikelompokan, lalu pada setiap langkah iteratif dengan jarak minimum dikelompokkan bersama sehingga menjadi satu kelompok dari hasil jarak kedekatanya.

Complate-linkage juga dikenal sebagai tetangga terjauh. kedua cluster dapat digabungkan. Oleh karena itu, antara cluster pertama dan cluster kedua harus memiliki jarak yang jauh. Average-linkage yaitu metode clustering yang mengelompokkan suatu dokumen atau objek dengan jarak ratarata.



Gambar 2.2 Ilustrasi Perhitungan Jarak dengan Metode Linkage

Untuk menghitung jarak dengan Euclidean yaitu dengan rumus seperti berikut:

$$\text{dist}(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Dimana:

x dan y = dua objek yang akan dihitung jaraknya

x_i = komponen ke i dari x secara berurutan

Simulasi Perhitungan Manual Euclidean Distance :

Selanjutnya dilakukan perhitungan dengan menggunakan seluruh data dijadikan sebagai variable pendukung. Sehingga diperoleh hasil sebagai berikut:

$$D(1,1), D(2,2), D(3,3), D(4,4), D(5,5)=0$$

$$D(1,2) = \sqrt{(1-4)^2 + (1-1)^2} = \sqrt{9} = 3$$

$$D(1,3) = \sqrt{(1-1)^2 + (1-2)^2} = \sqrt{1} = 1$$

$$D(1,4) = \sqrt{(1-3)^2 + (1-4)^2} = \sqrt{13} = 3,61$$

$$D(1,5) = \sqrt{(1-5)^2 + (1-4)^2} = \sqrt{25} = 5$$

$$D(2,3) = \sqrt{(4-1)^2 + (1-2)^2} = \sqrt{10} = 3,16$$

$$D(2,4) = \sqrt{(4-3)^2 + (1-4)^2} = \sqrt{10} = 3,16$$

$$D(2,5) = \sqrt{(4-5)^2 + (1-4)^2} = \sqrt{10} = 3,16$$

$$D(3,4) = \sqrt{(1-3)^2 + (2-4)^2} = \sqrt{8} = 2,82$$

$$D(3,5) = \sqrt{(1-5)^2 + (2-4)^2} = \sqrt{20} = 4,47$$

$$D(4,5) = \sqrt{(3-5)^2 + (4-4)^2} = \sqrt{4} = 2$$

Data	Fitur x	Fitur y
1	1	1
2	4	1
3	1	2
4	3	4
5	5	4

Setelah mendapatkan hasil perhitungan jarak dari seluruh data yang dimiliki, langkah selanjutnya ialah memindahkan hasil perhitungan kedalam bentuk tabel matrix seperti yang ditunjukkan pada Tabel 2.3.

Dman	1	2	3	4	5
1	0	3	1	3,61	5
2	3	0	3,16	3,16	3,16
3	1	3,16	0	2,82	4,47
4	3,61	3,16	2,82	0	2
5	5	3,16	4,47	2	0

Tabel 2.3 Matrix Perhitungan Jarak

Dapat dilihat pada Tabel 2.3 merupakan hasil perhitungan jarak yang telah dimasukkan kedalam tabel matrix. Dimana pada table 2.3 Kemudian, langkah selanjutnya ialah menentukan jarak terdekat atau nilai terkecil yang dimiliki oleh seluruh data. Tabel 2.4, nilai terkecil dimiliki oleh data 1 ke 3 dengan masing-masing bernilai 1. Oleh karena itu data 1 dan 3 akan digabungkan menjadi satu cluster. Tahapan tersebut terus dilakukan hingga perhitungan antara jarak data secara keseluruhan selesai dilakukan.

4.1 Hasil Perhitungan Jarak Complete Linkage

Langkah berikutnya ialah kembali melakukan perhitungan jarak terdekat, namun kali ini menggunakan Algoritma AHC dengan metode complete linkage. Seperti yang sudah dijelaskan sebelumnya, bahwa data 1 dan 3 telah digabung menjadi 1 cluster Pilih jarak kelompok terkecil 13 kemudian setelah dicari kelompok yang tersisa adalah jarak kelompok terbesar, maka perhitungan dengan menggunakan metode Complete linkage dilakukan sebagai berikut:

$$\text{Min } (D_{\text{man}}) = \min(d_{13}) = 1$$

$$D_{(13)2} = \max\{d_{12}, d_{32}\} = \max\{(3), (3,16)\} = 3,16$$

$$D_{(13)4} = \max\{d_{14}, d_{34}\} = \max\{(3,61), (2,28)\} = 3,61$$

$$D_{(13)5} = \max\{d_{15}, d_{35}\} = \max\{(5), (4,47)\} = 5$$

Dman	(13)	2	4	5
(13)	0	3,16	3,61	5
2	3,16	0	3,16	3,16
4	3,61	3,16	0	2
5	5	3,16	2	0

Tabel 2.4 Hasil Matrix Jarak

Lakukan Hal yang sama seperti yang di atas sehingga membentuk satu cluster yang besar. Pilih jarak kelompok terkecil $\text{Min } (D_{\text{man}}) = \min(d_{45}) = 2$ Hitung jarak kelompok (4,5) terhadap yang tersisa (13) dan 2.

$$D_{(45)(13)} = \max\{d_{41}, d_{43}, d_{51}, d_{53}\} = \max\{(3,61), (2,82), (5), (4,47)\} = 5$$

$$D_{(45)2} = \max\{d_{42}, d_{52}\} = \max\{(3,16), (3,16)\} = 3,16$$

Dman	(45)	(13)	2
(45)	0	5	3,16
(13)	5	0	3,16
2	3,16	3,16	0

Tabel 2.5 Hasil Matrix Jarak Lanjutan

$$\text{Min } (D_{\text{man}}) = \min(d_{45}) = 3,16$$

Hitung jarak kelompok (45) dan 2 terhadap yang tersisa (13).

$$D_{(452)(13)} = \max\{d_{41}, d_{43}, d_{51}, d_{53}, d_{21}, d_{23}\} = \max\{(3,61), (2,82), (5), (4,47), (3), (3,16)\} = 5$$

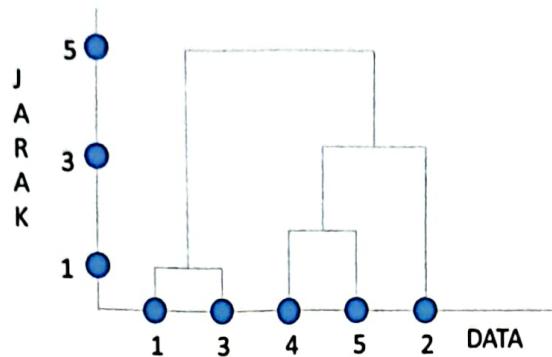
Dman	(452)	(13)
(452)	0	5
(13)	5	0

Tabel 2.6 Hasil Matrix Jarak Akhir

Pada perhitungan diatas merupakan perhitungan dengan menggunakan metode Complete linkage. Perhitungan pertama dilakukan kepada data ke-1. Dimana minimal jarak/nilai dari data 1 ke 3 ialah 1. Selanjutnya, dari kedua data yang sudah didapatkan maka hanya data terkecil yang digunakan. Sehingga pada perhitungan data ke-1 dan ke-3, nilai terkecil yang akan digunakan ialah 1 kemudia menentukan jarak terjauhnya setelah menentukan jarak minimum. Begitu seterusnya perhitungan dilakukan hingga ke seluruh data. Langkah selanjutnya ialah kembali memindahkan hasil perhitungan jarak kedalam bentuk tabel matrix seperti yang dapat dilihat pada Tabel di atas. Kemudian dari tabel matrix tersebut kembali akan dicari jarak minimum setelahnya jarak terjauh untuk masing masing data, sehingga akan memunculkan cluster baru atau bahkan adanya data yang bergabung dengan cluster sebelumnya. Perhitungan akan terus dilakukan hingga seluruh data menemukan cluster yang cocok dengannya.

5. Pembentukan Dendrogram

Dendrogram adalah bentuk dari pohon biner yang biasanya digunakan untuk memvisualisasikan hubungan dari hierarki dalam data seperti pada clustering. Sebuah dendrogram sering diintegrasikan dengan matriks yang dapat diurutkan ulang, yang disebut sebagai tampilan dendrogram-matriks, tetapi skala untuk menentukan lokasi jarak antar cluster sangat sulit dilakukan. Algoritma ini didasarkan pada penilaian kesamaan nilai atribut mereka, pohon dendrogram sebagai diagram analisis cluster digunakan dalam pembelajaran mesin untuk kasus di mana kelas tidak ditentukan sebelumnya dan harus dikenali oleh algoritme analisis statistic dengan mencari kesamaan terdekat.



Gambar 2.3 Hasil Dendrogram Matrix

Hasil dari perhitungan jarak menggunakan perhitungan Complete Euclidean dan Complete Linkage untuk mencari jarak terjauh dari seluruh data yang dihitung, diperoleh pembentukan 2 cluster dalam bentuk dendrogram pada gambar 2.3 diatas.

5. Evaluasi Shilhouette Score

Satu cara yang dapat digunakan untuk mengetahui kekuatan klaster dan melihat kualitasnya adalah menggunakan evaluasi silhouette coefficient. Metode ini banyak digunakan untuk memvalidasi klaster yang menggabungkan nilai kohesi dan separasi. Rentang nilai SI adalah -1 hingga +1. Nilai SI mendekati 1 menunjukkan bahwa data tersebut tidak tepat berada pada klaster tersebut. SI bernilai 0 atau mendekati 0 maka posisi data berada pada perbatasan dua klaster. Penafsiran nilai Silhouette coefficient ditunjukkan dalam Tabel 3.

Silhouette Score	Interpretasi
0.71 - 1.00	Struktur yang dihasilkan kuat
0.51 - 0.70	Struktur yang dihasilkan baik
0.26 - 0.50	Struktur yang dihasilkan lemah
≤0.25	Tidak Terstruktur

Tabel 2.7 Interpretasi Nilai Silhouete Score

7. Perangkat Lunak atau Software

Python adalah bahasa pemrograman yang banyak digunakan dalam aplikasi web, pengembangan perangkat lunak, ilmu data, dan machine learning (ML). Developer menggunakan Python karena efisien dan mudah dipelajari serta dapat dijalankan di berbagai platform.

7.1 library python

Library adalah kumpulan kode yang sering digunakan yang dapat disertakan oleh developer dalam program Phyton untuk menghindari menulis kode dari awal. Secara default, Phyton dilengkapi dengan Pustaka Standar, yang memuat banyak fungsi yang dapat digunakan kembali. Selain itu, lebih dari 137.000 pustaka Phyton tersedia

7.1.1 Pandas

Pandas menyediakan struktur data fleksibel yang dioptimalkan yang dapat Anda gunakan untuk memanipulasi data deret waktu dan data terstruktur, seperti tabel dan deret. Contohnya, Anda dapat menggunakan Pandas untuk membaca, menulis, menggabungkan, memfilter, dan mengelompokkan data. Banyak orang menggunakannya untuk tugas ilmu data, analisis data, dan ML.

7.1.2 Numpy

NumPy adalah pustaka populer yang digunakan developer untuk dengan mudah membuat dan mengelola deret, memanipulasi bentuk logis, dan melakukan operasi aljabar linier. NumPy mendukung integrasi dengan banyak bahasa seperti C dan C++.

7.1.3 Matplotlib

Developer menggunakan Matplotlib untuk memplot data dalam grafik dua dan tiga dimensi (2D dan 3D) berkualitas tinggi. Matplotlib sering digunakan dalam aplikasi ilmiah. Dengan Matplotlib, Anda dapat memvisualisasikan data dengan menampilkannya dalam diagram yang berbeda seperti diagram batang dan diagram garis. Anda juga dapat memplot beberapa diagram sekaligus, dan grafiknya portabel di seluruh platform.

7.1.4 Scipy

SciPy (Scientific Python) adalah perpustakaan *open-source* yang digunakan untuk perhitungan ilmiah tingkat tinggi. Jenis *library* ini dibangun di atas ekstensi NumPy dan bekerja bersama untuk menangani komputasi yang kompleks. NumPy memungkinkan pengurutan dan pengindeksan data *array*, sementara kode data numerik disimpan di SciPy.

7.1.5 Scikit-Learn

Scikit-learn adalah *library* python terkenal yang digunakan untuk data kompleks. Perpustakaan *open source* ini mendukung *machine learning* dengan mendukung berbagai algoritma yang diawasi dan tidak diawasi seperti regresi linier, klasifikasi, pengelompokan, dan lain sebagainya.

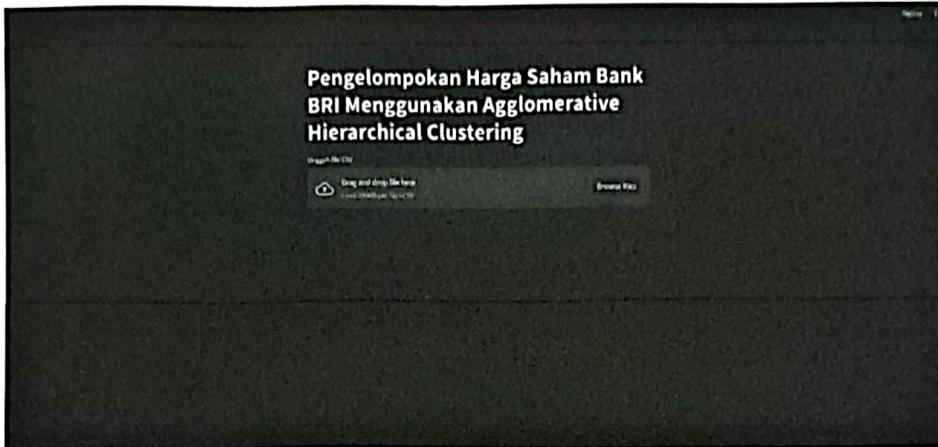
7.1.6 Streamlit

Streamlit adalah sebuah framework berbasis Python dan bersifat *open-source* yang dibuat untuk memudahkan dalam membangun aplikasi web di bidang sains data dan machine learning yang interaktif. Salah satu hal menarik dari framework ini adalah kita tidak perlu mengetahui banyak hal tentang teknologi web development. Kita tidak perlu dipusingkan tentang bagaimana mengatur tampilan website dengan CSS, HTML, atau Javascript. Untuk menggunakan Streamlit, kita cukup memiliki modal dasar mengetahui bahasa Python saja.

BAB III

IMPLEMENTASI

1. Dashboard



Gambar 3.1 Dashboard

Dashboard digunakan untuk mengungah file data set dalam bentuk csv.

Kode Program:

```
# Memuat Fungsi data dari file csv
@st.cache_data
def load_data(file):
    df = pd.read_csv(file)
    return df

# Unggah file csv
upload_file = st.file_uploader("Unggah file CSV", type=["csv"])
if upload_file is not None:
    df = load_data(upload_file)
```

2. Pemilihan Fitur Berdasarkan Karakteristik Dataset



Gambar 3.2 Pemilihan Fitur

Kode Program :

```
# Pilih fitur
fitur = st.multiselect("Pilih Fitur", df.columns.tolist())
```

```

if fitur:
    x = df[fitur].values

```

3. Memunculkan Data set

	Date	Open	High	Low	Close	Volume	AdjustedClose
0	20,240,102	5,650	5,675	5,625	5,675	91,143,100	5,675
1	20,240,103	5,625	5,650	5,600	5,600	83,659,700	5,600
2	20,240,104	5,600	5,700	5,575	5,700	118,467,900	5,700
3	20,240,105	5,725	5,850	5,700	5,750	134,929,600	5,750
4	20,240,108	5,750	5,800	5,625	5,625	152,015,000	5,625
5	20,240,109	5,625	5,700	5,625	5,700	105,666,200	5,700
6	20,240,110	5,625	5,700	5,625	5,700	82,626,400	5,700
7	20,240,111	5,750	5,800	5,700	5,750	132,206,400	5,750
8	20,240,112	5,800	5,850	5,775	5,850	109,715,700	5,850
9	20,240,115	5,850	5,900	5,775	5,825	115,284,200	5,825
10	20,240,116	5,825	5,875	5,775	5,825	97,074,300	5,825
11	20,240,117	5,825	5,825	5,750	5,775	107,941,600	5,775
12	20,240,118	5,800	5,800	5,700	5,750	87,259,100	5,750
13	20,240,119	5,750	5,800	5,700	5,800	75,370,700	5,800
14	20,240,122	5,775	5,800	5,700	5,775	97,545,600	5,775
15	20,240,123	5,700	5,750	5,650	5,700	122,999,700	5,700
16	20,240,124	5,675	5,700	5,575	5,650	104,225,000	5,650
17	20,240,125	5,575	5,650	5,525	5,525	120,478,800	5,525
18	20,240,126	5,450	5,475	5,400	5,425	185,244,400	5,425
19	20,240,129	5,400	5,575	5,400	5,575	145,629,600	5,575
20	20,240,130	5,525	5,650	5,475	5,625	150,287,800	5,625
21	20,240,131	5,725	5,775	5,675	5,700	245,117,200	5,700

Gambar 3.3 Data set

Kode Program :

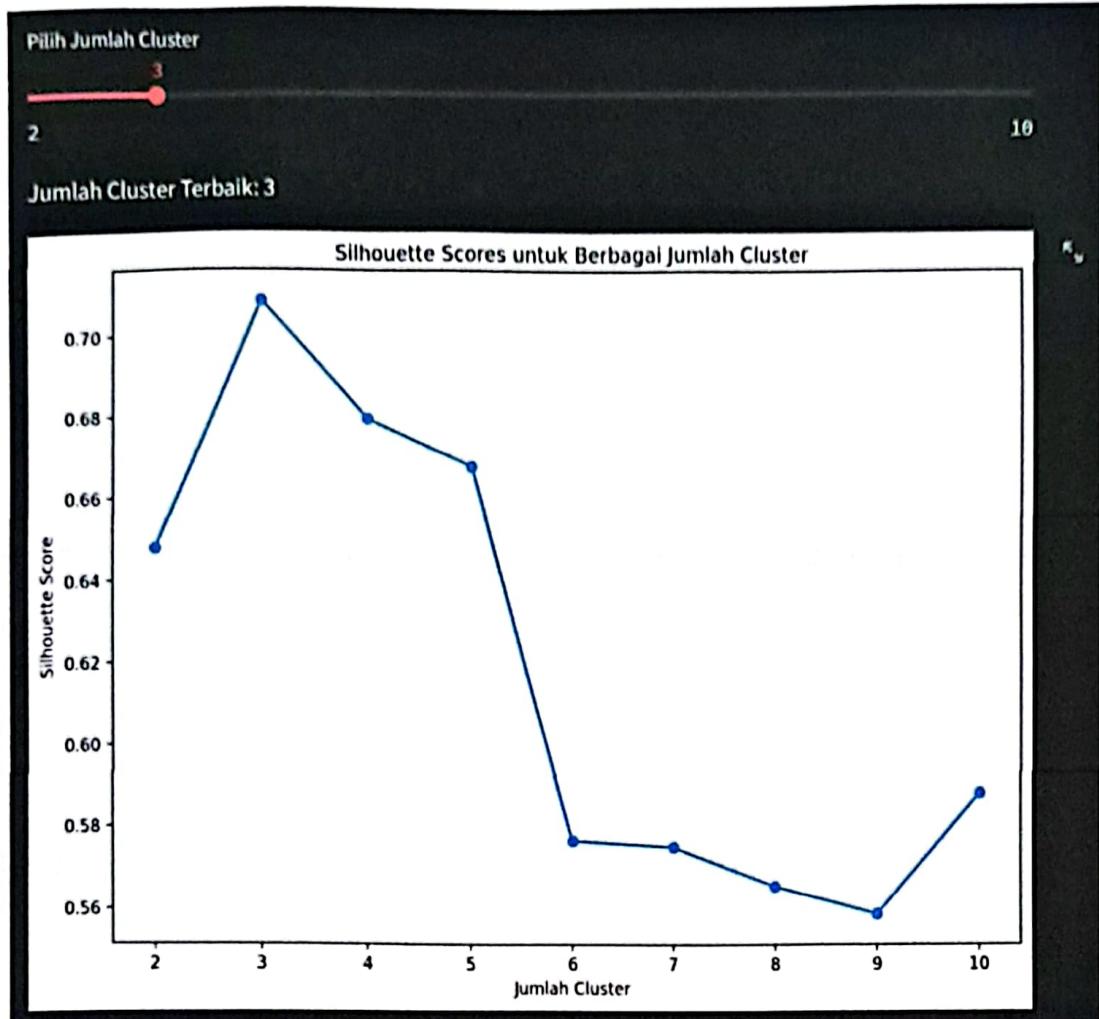
```

st.write("Data Harga Saham:")
st.write(df)

```

4. Fitur Preprocessing dan Pilihan Parameter Clustering





Gambar 3.4 Proses Preprocessing dan Pilihan Parameter Clustering

Kode Program :

```
# Checkbox untuk memilih normalisasi dan skala
use_normalizer = st.checkbox("Gunakan Normalizer")
use_scaler = st.checkbox("Gunakan Standard Scaler")

if use_normalizer:
    normalizer = Normalizer()
    x = normalizer.fit_transform(x)

if use_scaler:
    scaler = StandardScaler()
    x = scaler.fit_transform(x)

# Pilih parameter clustering
linkage_method = st.selectbox("Pilih metode linkage", ["complete",
    "average", "single"])
n_clusters = st.slider("Pilih Jumlah Cluster", 2, 10, 4)
jumlah_clusters_terbaik, silhouette_scores = find_best_cluster(x_train,
    linkage_method)
```

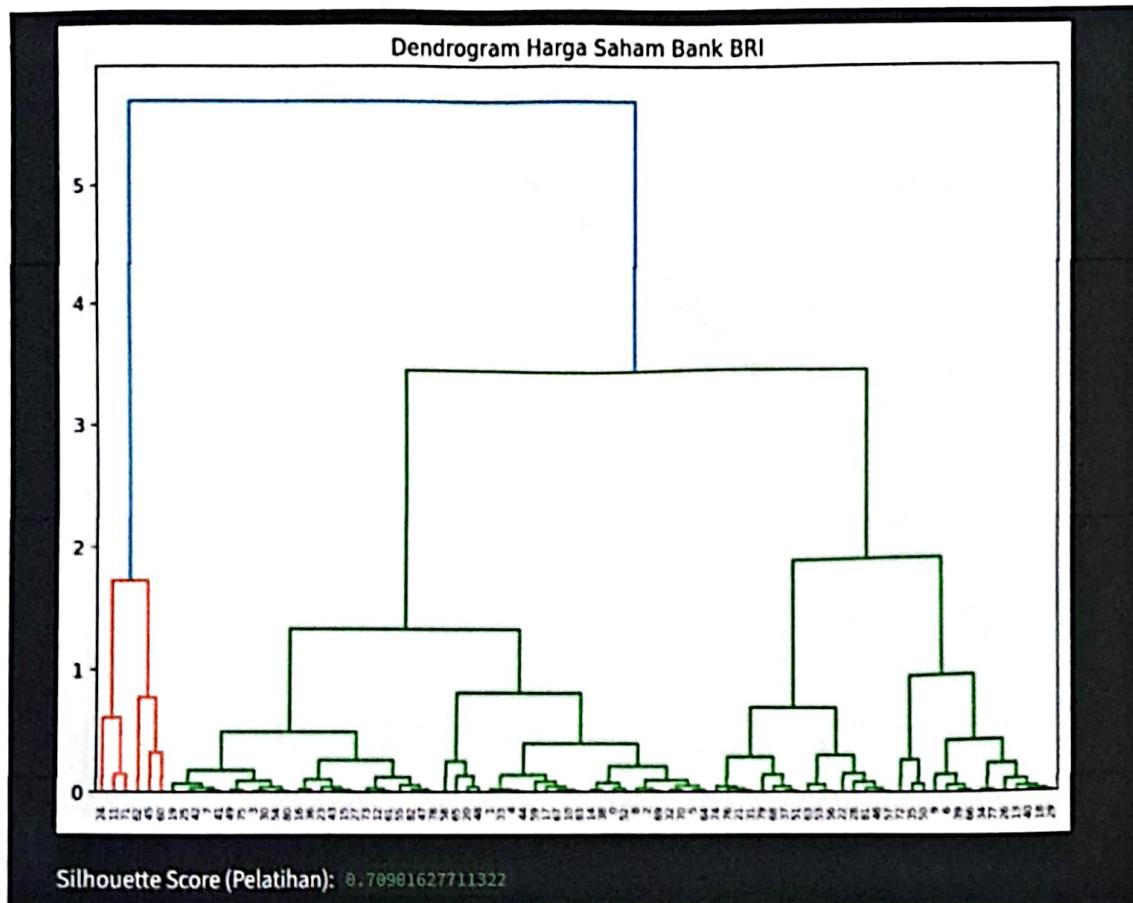
```

st.write(f'Jumlah Cluster Terbaik: {jumlah_clusters_terbaik}')

# Plot silhouette scores
fig, ax = plt.subplots(figsize=(10, 7))
ax.plot(range(2, 11), silhouette_scores, marker='o')
ax.set_title('Silhouette Scores untuk Berbagai Jumlah Cluster')
ax.set_xlabel('Jumlah Cluster')
ax.set_ylabel('Silhouette Score')
st.pyplot(fig)

```

5. Visualisasi Dendrogram



Gambar 3.5 Visualisasi Dendrogram

Kode Program:

```

def perform_clustering(x_train, n_clusters, linkage_method):
    agglomerative_cluster = AgglomerativeClustering(n_clusters=n_clusters,
affinity='euclidean', linkage=linkage_method)

    cluster_labels_train = agglomerative_cluster.fit_predict(x_train)

```

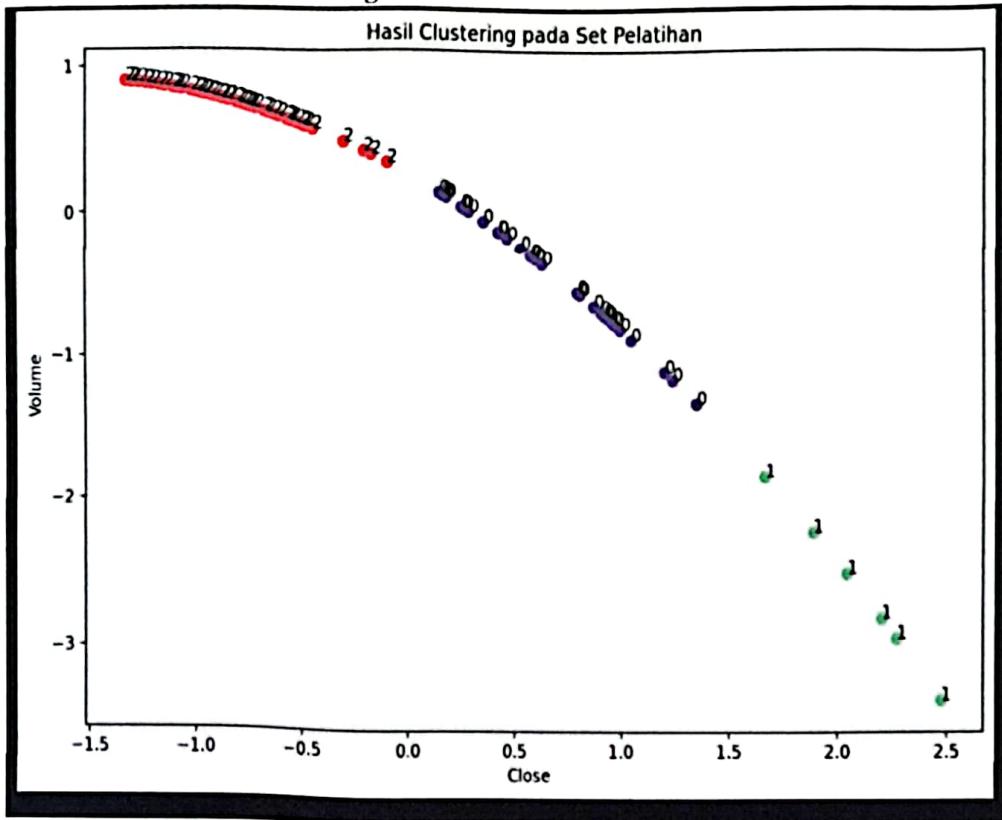
```

silhouette_avg_train = silhouette_score(x_train, cluster_labels_train)

fig, ax = plt.subplots(figsize=(10,7))
ax.set_title("Dendrogram Harga Saham Bank BRI")
dend = dendrogram(linkage(x_train, method=linkage_method), ax=ax)
st.pyplot(fig)
return silhouette_avg_train, agglomerative_cluster, cluster_labels_train

```

6. Visualisasi Hasil Clustering



Gambar 3.6 Visualisasi Hasil Clustering

Kode Program :

```

fig, ax = plt.subplots(figsize=(10, 7))

scatter = ax.scatter(x_train[:, 0], x_train[:, 1], c=cluster_labels_train,
cmap='rainbow')

ax.set_title('Hasil Clustering pada Set Pelatihan')
ax.set_xlabel(fitur[0])
ax.set_ylabel(fitur[1])
st.pyplot(fig)

```

```
# Tambahkan anotasi untuk setiap titik
for i in range(len(x_train)):
    ax.annotate(cluster_labels_train[i], (x_train[i, 0], x_train[i, 1]))

st.pyplot(fig)

X_train_df = pd.DataFrame(x_train, columns=fitur)
X_train_df['Cluster'] = cluster_labels_train
st.write("Hasil Clustering pada Set Pelatihan:")
st.write(X_train_df)
```

BAB IV

HASIL DAN KESIMULAN

1. Hasil Clustering (Complete Linkage)

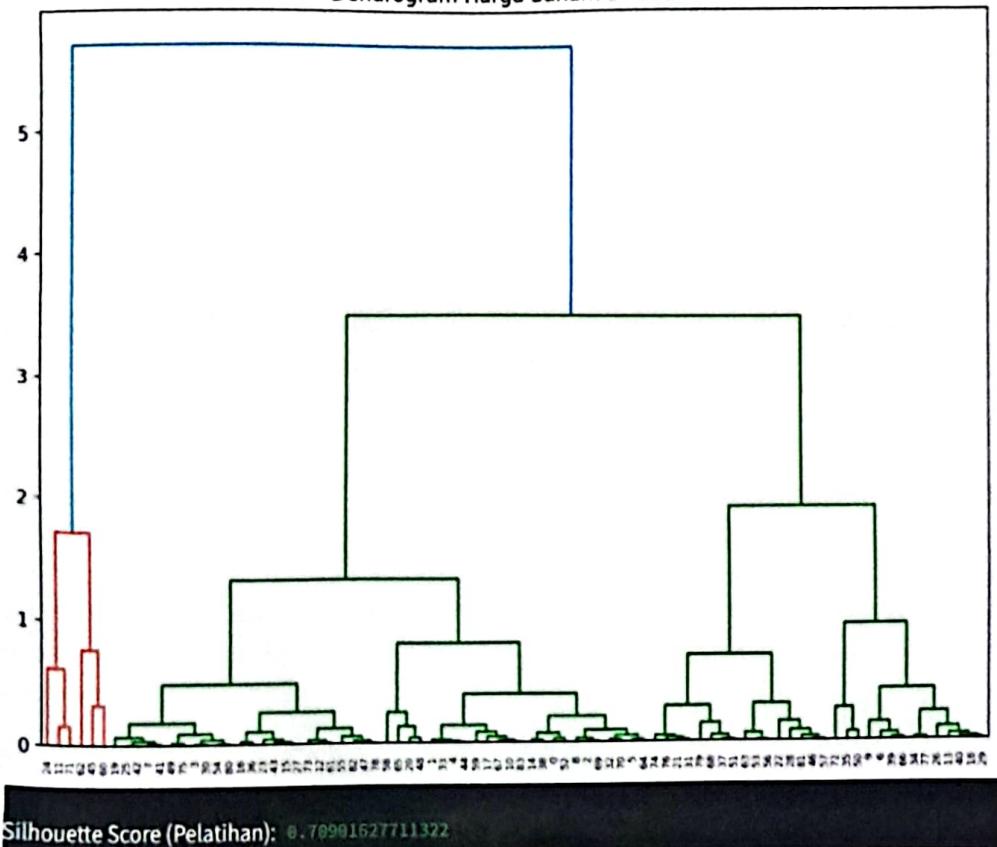
Cluster	Jumlah	Anggota Cluster
0	30	21,74,31,76,68,37,51,22,28,81,46,57,79,63,53,56,72,35,50,26,34,77,6,9,13,18,29,66,39,40
1	6	24,11,71,62,45,60
2	48	54,80,3,30,19,41,49,75,7,25,42,16,36,23,27,25,43,73,12,61,82,55,47,78,65,58,20,48,33,44,1,4,59,17,67,10,83,14,38,0,52,8,5,2,69,32,64,70

Tabel 2.8 Hasil Clustering

Data testing sebanyak 20% dibentuk 3 cluster dengan menggunakan data sebanyak 84 data training, setiap satu cluster dicari jarak terpendek (yang paling mirip) dengan cluster lainnya, kemudian digabungkan menjadi satu. setelah itu cluster yang baru dihitung lagi jaraknya dengan cluster lain, dicari terjauh, kemudian digabungkan sampai akhirnya menjadi satu cluster.

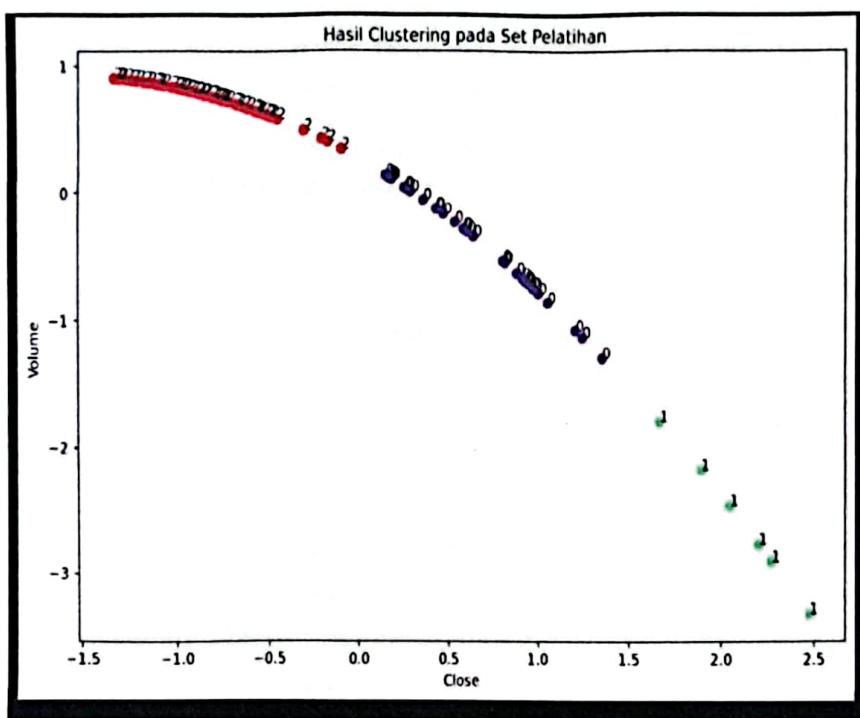
2. Hasil Visualisasi Dendrogram dan Plot Clustering

Dendrogram Harga Saham Bank BRI



Gambar 3.7 Hasil Visualisasi Dendrogram

Dendrogragan menunjukkan struktur hirarkis dari data, dengan setiap tingkatan pada pohon dendrogram menunjukkan langkah baru dalam proses pengelompokan. Titik bagian bawah data mewakili satu setiap data set saham. Setiap dua cabang digabungkan menjadi satu cabang baru itu menunjukkan bahwa dua data atau kelompok telah dikelompokkan menjadi satu cluster. Hasil dari perhitungan jarak menggunakan perhitungan Euclidean dan Complete Linkage untuk mencari jarak terjauh dari seluruh data yang dihitung, diperoleh pembentukan 3 cluster dalam membentuk dendrogram dengan Silhouette Score (Pelatihan): 0.70901627711322. Dari perolehan hasil Silhouette Score dapat dilihat pada gambar 3.7 membentuk cluster yang terstruktur dengan baik jika mengacu pada tabel interpretasi nilai Silhouette Score.



Gambar 3.8 Hasil Visualisasi Plot Clustering

Dalam analisis harga saham, dendrogram dan plot clustering dikelompokkan berdasarkan atribut karakteristik pergerakannya seperti harga penutup (close) dan volume perdagangan. Harga penutup sering digunakan sebagai referensi utama dalam analisis teknikal karena mencerminkan nilai suatu saham pada akhir suatu perdagangan. Harga penutup juga sering digunakan untuk mengidentifikasi tren harga jangka pendek maupun panjang. Sedangkan volume menunjukkan jumlah saham yang diperdagangkan dalam periode tertentu. Volume yang tinggi biasanya mengindikasikan adanya minat yang kuat dari pasar dan seringkali mendahului perubahan signifikan dalam harga.

Cluster 2 : Saham dengan Tren Naik

- Harga saham yang menunjukkan pola kenaikan konsisten selama periode waktu tertentu.

- Menunjukkan kepercayaan investor terhadap pertumbuhan perusahaan.

Cluster 1: Saham dengan Tren Turun

- Harga saham yang menunjukkan pola penurunan konsisten selama periode waktu tertentu.
- Menunjukkan kemungkinan adanya isu internal atau eksternal yang mempengaruhi kinerja perusahaan.

Cluster 0: Saham dengan Tren Sideways

- Harga saham yang bergerak dalam rentang harga yang relatif sempit tanpa adanya tren naik atau turun yang jelas.
- Menunjukkan periode konsolidasi di mana investor mungkin menunggu katalis tertentu sebelum harga bergerak lebih jauh.

3. Kesimpulan

Dari hasil pembahasan dan analisis yang telah dilakukan dapat ditarik kesimpulan:

1. Agglomerative Hierarchical Clustering dapat digunakan untuk pengelompokkan harga saham berdasarkan karakteristiknya dan dikaitkan dengan strategi investasi.
2. Hasil pengelompokan data sebanyak 84 data training diperoleh cluster 0 dengan 30 anggota, cluster 1 dengan 6 anggota dan cluster 2 dengan 48 anggota.
3. Evaluasi hasil metode Silhouette Score mendapatkan hasil yang jika dilihat pada tabel interpretasi nilai score dikatakan baik dan terstruktur yaitu: 0.70901627711322.
4. Hasil dari analisis ini dapat membantu investor dalam membuat keputusan investasi yang lebih baik dan terinformasi.

4. Saran

Berdasarkan hasil penelitian yang diperoleh beberapa saran yang dapat meningkatkan efektivitas proyek pengelompokan harga saham Bank Mandiri menggunakan algoritma Agglomerative Hierarchical Clustering:

1. Streamlit Dashboard: Kembangkan dashboard interaktif menggunakan Streamlit yang memungkinkan pengguna untuk unggah data, melakukan clustering, dan melihat hasil serta interpretasi secara real-time.
2. Integrasi API: Pertimbangkan integrasi dengan API untuk mendapatkan data saham real-time dan memperbarui hasil clustering secara berkala.

LAMPIRAN

Kode Program :

```
import pandas as pd
import numpy as np
import streamlit as st
import pickle

from sklearn.preprocessing import StandardScaler, Normalizer
from sklearn.cluster import AgglomerativeClustering
from sklearn.metrics import silhouette_score
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
from scipy.cluster.hierarchy import dendrogram, linkage

# Memuat Fungsi data dari file csv
@st.cache_data
def load_data(file):
    df = pd.read_csv(file)
    return df

# Fungsi untuk Clustering dan visualisasi dendrogram
def perform_clustering(x_train, n_clusters, linkage_method):
    agglomerative_cluster = AgglomerativeClustering(n_clusters=n_clusters,
affinity='euclidean', linkage=linkage_method)

    cluster_labels_train = agglomerative_cluster.fit_predict(x_train)

    silhouette_avg_train = silhouette_score(x_train, cluster_labels_train)

    fig, ax = plt.subplots(figsize=(10,7))
    ax.set_title("Dendrogram Harga Saham Bank BRI")
    dend = dendrogram(linkage(x_train, method=linkage_method), ax=ax)
    st.pyplot(fig)
```

```

# Fungsi untuk menyimpan model_clustering
with open('simpan_model.pkl', 'wb') as f:
    pickle.dump(agglomerative_cluster, f)

return silhouette_avg_train, agglomerative_cluster, cluster_labels_train

# Fungsi untuk menentukan jumlah cluster terbaik menggunakan metode Silhouette
def find_best_cluster(X, linkage_method):
    silhouette_scores = []
    for n_clusters in range(2, 11):
        agglomerative_cluster = AgglomerativeClustering(n_clusters=n_clusters,
affinity='euclidean', linkage=linkage_method)
        cluster_labels = agglomerative_cluster.fit_predict(X)
        silhouette_avg = silhouette_score(X, cluster_labels)
        silhouette_scores.append(silhouette_avg)

    jumlah_clusters_terbaik = np.argmax(silhouette_scores) + 2 # Karena range dimulai dari 2
    return jumlah_clusters_terbaik, silhouette_scores

# Fungsi untuk memuat simpan model dari file simpan_model.pkl
def load_simpan_model(file_name='simpan_model.pkl'):
    with open(file_name, 'rb') as f:
        simpan_model = pickle.load(f)
    return simpan_model

# Aplikasi pengelompokan saham
def main():
    st.title("Pengelompokan Harga Saham Bank BRI Menggunakan Agglomerative
Hierarchical Clustering")

```

```
# Unggah file csv
upload_file = st.file_uploader("Unggah file CSV", type=["csv"])
if upload_file is not None:
    df = load_data(upload_file)

    st.write("Data Harga Saham:")
    st.write(df)

# Pilih fitur
fitur = st.multiselect("Pilih Fitur", df.columns.tolist())

if fitur:
    x = df[fitur].values

    # Checkbox untuk memilih normalisasi dan skala
    use_normalizer = st.checkbox("Gunakan Normalizer")
    use_scaler = st.checkbox("Gunakan Standard Scaler")

    if use_normalizer:
        normalizer = Normalizer()
        x = normalizer.fit_transform(x)

    if use_scaler:
        scaler = StandardScaler()
        x = scaler.fit_transform(x)

    # Pembagian data menjadi data pelatihan dan pengujian set
    x_train, x_test = train_test_split(x, test_size=0.2, random_state=20)

# Pilih parameter clustering
```

```

linkage_method = st.selectbox("Pilih metode linkage", ["complete", "average",
"single"])

n_clusters = st.slider("Pilih Jumlah Cluster", 2, 10, 4)

jumlah_clusters_terbaik, silhouette_scores = find_best_cluster(x_train,
linkage_method)

st.write(f'Jumlah Cluster Terbaik: {jumlah_clusters_terbaik}')

# Plot silhouette scores

fig, ax = plt.subplots(figsize=(10, 7))

ax.plot(range(2, 11), silhouette_scores, marker='o')

ax.set_title('Silhouette Scores untuk Berbagai Jumlah Cluster')
ax.set_xlabel('Jumlah Cluster')
ax.set_ylabel('Silhouette Score')
st.pyplot(fig)

if st.button("Lakukan Clustering"):

    silhouette_avg_train, agglomerative_cluster, cluster_labels_train =
perform_clustering(x_train, n_clusters, linkage_method)

    st.write("Silhouette Score (Pelatihan):", silhouette_avg_train)

    train_clusters, train_counts = np.unique(cluster_labels_train, return_counts=True)

    st.write("Jumlah Anggota dalam Setiap Cluster Pada Set Pelatihan:")

    train_clusters_summary = pd.DataFrame({'Cluster': train_clusters, 'Jumlah
Anggota': train_counts})

    st.write(train_clusters_summary)

    fig, ax = plt.subplots(figsize=(10, 7))

    scatter = ax.scatter(x_train[:, 0], x_train[:, 1], c=cluster_labels_train,
cmap='rainbow')

    ax.set_title('Hasil Clustering pada Set Pelatihan')
    ax.set_xlabel(fitur[0])

```

```

ax.set_ylabel(fitur[1])
st.pyplot(fig)

# Tambahkan anotasi untuk setiap titik
for i in range(len(x_train)):
    ax.annotate(cluster_labels_train[i], (x_train[i, 0], x_train[i, 1]))

st.pyplot(fig)

X_train_df = pd.DataFrame(x_train, columns=fitur)
X_train_df['Cluster'] = cluster_labels_train
st.write("Hasil Clustering pada Set Pelatihan:")
st.write(X_train_df)

# Muat model yang disimpan
cluster_model = load_simpan_model()

cluster_labels_test = cluster_model.fit_predict(x_test)

fig, ax = plt.subplots(figsize=(10, 7))
scatter = ax.scatter(x_test[:, 0], x_test[:, 1], c=cluster_labels_test, cmap='rainbow')
ax.set_title('Hasil Clustering pada Set Pengujian')
ax.set_xlabel(fitur[0])
ax.set_ylabel(fitur[1])
st.pyplot(fig)

# Tambahkan anotasi untuk setiap titik
for i in range(len(x_test)):
    ax.annotate(cluster_labels_test[i], (x_test[i, 0], x_test[i, 1]))

```

```

st.pyplot(fig)

silhouette_avg_test = silhouette_score(x_test, cluster_labels_test)
st.write("Silhouette Score (Test):", silhouette_avg_test)

test_clusters, test_counts = np.unique(cluster_labels_test, return_counts=True)
st.write("Jumlah Anggota dalam Setiap Cluster pada Set Pengujian:")
test_cluster_summary = pd.DataFrame({'Cluster': test_clusters, 'Jumlah Anggota': test_counts})
st.write(test_cluster_summary)

X_test_df = pd.DataFrame(x_test, columns=fitur)
X_test_df['Cluster'] = cluster_labels_test
st.write("Hasil Clustering pada Set Pengujian:")
st.write(X_test_df)

kombinasi_df = pd.concat([X_train_df, X_test_df], ignore_index=True)
st.write("Hasil Clustering Gabungan:")
st.write(kombinasi_df)

if __name__ == "__main__":
    main()

```

Link Github : https://github.com/123pratama/Streamlit_hargasaham

Website : <https://appargasaham-ffctsxw4mtmcnspu9veyjx.streamlit.app/>

DAFTAR PUSTAKA

- A. Ambarwati, Q. J. Adrian and Y. Herdiyeni, "Analisis Pengaruh Data Scaling Terhadap Performa Algoritme Machine Learning untuk Identifikasi Tanaman," *Jurnal RESTI*, vol. 4, no. 1, pp. 117-122, 2020.
- Aji, B. P., & Wibisono, M. A. (2018). Strategi Pengambilan Keputusan Penjualan Dalam Rangka Optimasi Profit Industri Ritel Berbasis Unsupervised Machine Learning Algorithm (Studi Kasus Modern Minimarket-X).
- Board, F. S. (2017). Artificial intelligence and machine learning in financial services. November, Available at: <Http://Www. Fsb. Org/2017/11/Artificialintelligenceand-Machine-Learning-inFinancialservice/>(Accessed 30th January, 2018).
- Board, F. S. (2017). Artificial intelligence and machine learning in financial services. November, Available at: <Http://Www. Fsb. Org/2017/11/Artificialintelligenceand-Machine-Learning-inFinancialservice/>(Accessed 30th January, 2018).
- Brownlee, J. (2016). Master Machine Learning Algorithms: discover how they work and implement them from scratch. Jason Brownlee
- Das, S., & Nene, M. J. (2017). A survey on types of machine learning techniques in intrusion prevention systems. 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2296–2299. <https://doi.org/10.1109/WiSPNET.2017.8300169>
- Dharmawan, D. A., Ng, B. P., & Rahardja, S. (2018). A Modified Dolph-Chebyshev Type II Function Matched Filter for Retinal Vessels Segmentation. *Symmetry*, 10(7). <https://doi.org/10.3390/sym10070257>
- E. N. Narciso, M. Delamaro, and F. L. S. Nunes, "Test case selection using CBIR and clustering Test Case Selection Using CBIR View project Automation of test Oracles for the processing of medical images of three-dimensional models View project," 2013. [Online]. Available: <https://www.researchgate.net/publication/290332664>
- Ester, M., Kriegel, H.-P., Sander, J., Xu, X., & others. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise.
- F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 1, pp. 86–97, Jan. 2012, doi: 10.1002/widm.53.
- Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. *Machine Learning*, 3(2), 95–99.
https://www.ir-bri.com/historical_price.html
- Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, 70(1–3), 489–501
- Mitchell, T. M. (1997). Machine learning. In McGraw Hill Series in Computer Science. Retrieved from <http://www.worldcat.org/oclc/61321007>

- N. Huda, N. Debataraja, and E. Sulistianingsih, "ESTIMASI CONFIDENCE TITIK PANAS PADA KEBAKARAN HUTAN MENGGUNAKAN METODE ORDINARY KRIGING", doi: 10.13140/RG.2.2.35901.87526.
- O. Yim and K. T. Ramdeen, "Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data," 2015.
- Rahardja, U., Roihan, A., & others. (2017). Design of Business Intelligence in Learning Systems Using iLearning Media. Universal Journal of Management, 5(5), 227–235.
- Russell, S. J., & Norvig, P. (2016). Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited.,
- S. Hirano, X. Sun, and S. Tsumoto, "Comparison of clustering methods for clinical databases," Information Sciences, vol. 159, no. 3–4, pp. 155–165, Feb. 2004, doi: 10.1016/j.ins.2003.03.011.
- Somvanshi, M., & Chavan, P. (2016). A review of machine learning techniques using decision tree and support vector machine. 2016 International Conference on Computing Communication Control and Automation (ICCUBEA), 1–7. <https://doi.org/10.1109/ICCUBEA.2016.7860040>
- Sunarya, A., Santoso, S., & Sentanu, W. (2015). Sistem Pakar Untuk Mendiagnosa Gangguan Jaringan Lan. CCIT Journal, 8(2), 1–11.
- Thupae, R., Isong, B., Gasela, N., & AbuMahfouz, A. M. (2018). Machine Learning Techniques for Traffic Identification and Classification in SDWSN: A Survey. IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society, 4645–4650. <https://doi.org/10.1109/IECON.2018.8591178>
- Wu, G., Kim, M., Wang, Q., Gao, Y., Liao, S., & Shen, D. (2013). Unsupervised deep feature learning for deformable registration of MR brain images. International Conference on Medical Image Computing and Computer-Assisted Intervention, 649–656