DAFTAR PUSTAKA

Abbasimehr, H., & Shabani, M. (2019). A new methodology for customer behavior analysis using time series clustering: A case study on a bank's customers. *Kybernetes*, *50*(2), 221–242. https://doi.org/10.1108/K-09-2018-0506

Abdi, H., & Williams, L. (2019). Honestly significant difference (HSD) test. *Encyclopedia of Research Design*, 583–585.

Adiwijaya, K., Wahyuni, S., & Mussry, J. S. (2017). Marketing ambidexterity and marketing performance: Synthesis, a conceptual framework, and research propositions. In *Enhancing Business Stability Through Collaboration*. CRC Press.

Agbangba, C. E., Sacla Aide, E., Honfo, H., & Glèlè Kakai, R. (2024). On the use of post-hoc tests in environmental and biological sciences: A critical review. *Heliyon*, *10*(3), e25131. https://doi.org/10.1016/j.heliyon.2024.e25131

Ahn, J., Hwang, J., Kim, D., Choi, H., & Kang, S. (2020). A Survey on Churn Analysis in Various Business Domains. *IEEE Access*, *8*, 220816–220839. https://doi.org/10.1109/ACCESS.2020.3042657

Alvi, R., Khan, F. M. H. (18DET05), Kadiwal, A. Y. (17ET20), Patel, S. S. (18DET10), & Qureshi, F. F. A. (18DET11). (2021). *Integrated approach of RFM, clustering, CLTV and machine learning algorithm for forecasting*. http://localhost:8080/xmlui/handle/123456789/3435

Anand Ramalingam, P., Fathima, N., Supriya, P., Shetty, P., Sanyal, M., Yeshaswini, P., Rao, M., Darapaneni, N., & Paduri, A. R. (2023). Data Complexity for Identifying Suitable Algorithms. *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, 0955–0961. https://doi.org/10.1109/CCWC57344.2023.10099201

Aria, M., Cuccurullo, C., & Gnasso, A. (2021). A comparison among interpretative proposals for Random Forests. *Machine Learning with Applications*, *6*, 100094. https://doi.org/10.1016/j.mlwa.2021.100094

Ashfania, G. A. M., Prahasto, T., Widodo, A., & Warsokusumo, T. (2022). Penggunaan Algoritma Random Forest untuk Klasifikasi berbasis Kinerja Efisiensi Energi pada Sistem Pembangkit Daya. *ROTASI*, *24*(3), 14–21. https://doi.org/10.14710/rotasi.24.3.14-21

Ayyagari, M. (2019). A Framework for Analytical CRM Assessments Challenges and Recommendations. *International Journal of Business and Social Science*, *10*. https://doi.org/10.30845/ijbss.v10n6p2

Banggang, W. U., Liu, Y., & Yubo, C. (2020). An Empirical Study of Purchase Rate and Dropout Rate Between Mobile and PC Customers. *Journal of Systems & Management*, *29*(5), 924. https://doi.org/10.3969/j.issn.1005-2542.2020.05.010

Bavykina, E. M. (2022). Analysis of the concept of. *Siberian Financial School*. https://www.semanticscholar.org/paper/Analysis-of-the-concept-of-%22cluster%22-and-its-Bavykina/afd6b71eb9999eef90e3c95312588779272aa7fc

Bhuyan, R., & Borah, S. (2013). *A Survey of Some Density Based Clustering Techniques*. https://doi.org/10.13140/2.1.4554.6887

Blessing, E., & Klaus, H. (2023). *Normalization and Standardization: Methods to preprocess data to have consistent scales and distributions*. *2237*, 10.

Buckinx, W., & Van den Poel, D. (2005). Customer base analysis: Partial defection of behaviourally loyal clients in a non-contractual FMCG retail setting. *European Journal of Operational Research*, *164*(1), 252–268. https://doi.org/10.1016/j.ejor.2003.12.010

Cam, L. M. L., & Neyman, J. (1967). *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press.

Chen, X., Zhang, K., Jiao, S., & Chen, J. (2023). An improved data mining algorithm in cloud computing platform. *International Conference on Mathematics, Modeling, and Computer Science (MMCS2022)*, *12625*, 812–817. https://doi.org/10.1117/12.2670436

Chen, Y., Xie, X., Lin, S.-D., & Chiu, A. (2018). WSDM Cup 2018: Music Recommendation and Churn Prediction. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 8–9. https://doi.org/10.1145/3159652.3160605

Christy, A. J., Umamakeswari, A., Priyatharsini, L., & Neyaa, A. (2021). RFM ranking – An effective approach to customer segmentation. *Journal of King Saud University - Computer and Information Sciences*, *33*(10), 1251–1257. https://doi.org/10.1016/j.jksuci.2018.09.004

Erlin, E., Desnelita, Y., Nasution, N., Suryati, L., & Zoromi, F. (2022). Dampak SMOTE terhadap Kinerja Random Forest Classifier berdasarkan Data Tidak seimbang. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, *21*, 677–690. https://doi.org/10.30812/matrik.v21i3.1726

Farris, P. W., & Buzzell, R. D. (1979). Why Advertising and Promotional Costs Vary: Some Cross-Sectional Analyses. *Journal of Marketing*, *43*(4), 112–122. https://doi.org/10.1177/002224297904300413

Fernandes, V., Carvalho, G., Pereira, V., & Bernardino, J. (2024). Analyzing Data Reduction Techniques: An Experimental Perspective. *Applied Sciences*, *14*(8), Article 8. https://doi.org/10.3390/app14083436

Ferreira-Santos, D., & Pereira Rodrigues, P. (2018). Finding Groups in Obstructive Sleep Apnea Patients: A Categorical Cluster Analysis. *2018 IEEE 31st International*

Symposium on Computer-Based Medical Systems (CBMS), 387–392. 2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS). https://doi.org/10.1109/CBMS.2018.00074

Gattermann-Itschert, T., & Thonemann, U. W. (2022). Proactive customer retention management in a non-contractual B2B setting based on churn prediction with random forests. *Industrial Marketing Management*, *107*, 134–147. https://doi.org/10.1016/j.indmarman.2022.09.023

Gil-Gomez, H., Guerola-Navarro, V., Oltra-Badenes, R., & Lozano-Quilis, J. A. (2020). Customer relationship management: Digital transformation and sustainable business model innovation. *Economic Research-Ekonomska Istraživanja*, *33*(1), 2733–2750. https://doi.org/10.1080/1331677X.2019.1676283

Gormley, I. C., Murphy, T. B., & Raftery, A. E. (2023). Model-Based Clustering. *Annual Review of Statistics and Its Application*, *10*(1), 573–595. https://doi.org/10.1146/annurev-statistics-033121-115326

Guerola-Navarro, V., Gil-Gomez, H., Oltra-Badenes, R., & Soto-Acosta, P. (2022). Customer relationship management and its impact on entrepreneurial marketing: A literature review. *International Entrepreneurship and Management Journal*. https://doi.org/10.1007/s11365-022-00800-x

Ha, J., & (Shawn) Jang, S. (2010). Perceived values, satisfaction, and behavioral intentions: The role of familiarity in Korean restaurants. *International Journal of Hospitality Management*, *29*(1), 2–13. https://doi.org/10.1016/j.ijhm.2009.03.009

Hanif, T. T., Adiwijaya, A., & Al-Faraby, S. (2017). Analisis Churn Prediction Pada Data Pelanggan Pt. Telekomunikasi Menggunakan Underbagging Dan Logistic Regression. *eProceedings of Engineering*, *4*(2), Article 2. https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/1355

Hardjono, B., & San, L. (2017). Customer Relationship Management Implementation and its Implication to Customer Loyalty in Hospitality Industry. *Jurnal Dinamika Manajemen*, *8*. https://doi.org/10.15294/jdm.v8i1.10414

Hasan, B. M. S., & Abdulazeez, A. M. (2021). A Review of Principal Component Analysis Algorithm for Dimensionality Reduction. *Journal of Soft Computing and Data Mining*, *2*(1), Article 1.

Hiray, D. A. P., & Anjum, D. A. (2022). Customer Value: A Systematic Literature Review. *Journal of Positive School Psychology*, *6*(2), Article 2.

Hooley, G., Piercy, N. F., Nicoulaud, B., & Rudd, J. M. (n.d.). *Marketing Strategy & Competitive Positioning*.

Howell, D. C. (2010). *Statistical methods for psychology* (7th ed). Thomson Wadsworth.

Huang, M.-W., Tsai, C.-F., Tsui, S.-C., & Lin, W.-C. (2023). Combining data discretization and missing value imputation for incomplete medical datasets. *PLOS ONE*, *18*(11), e0295032. https://doi.org/10.1371/journal.pone.0295032

Humaira, H., & Rasyidah, R. (2020, January 1). *Determining The Appropiate Cluster Number Using Elbow Method for K-Means Algorithm*. https://doi.org/10.4108/eai.24-1-2018.2292388

Jancey, R. (1966). Multidimensional group analysis. *Australian Journal of Botany*, *14*(1), 127. https://doi.org/10.1071/BT9660127

Januzaj, Y., Beqiri, E., & Luma, A. (2023). Determining the Optimal Number of Clusters using Silhouette Score as a Data Mining Technique. *International Journal of Online and Biomedical Engineering (iJOE)*, *19*(04), Article 04. https://doi.org/10.3991/ijoe.v19i04.37059

Jin, X., & Han, J. (2010). Partitional Clustering. In C. Sammut & G. I. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 766–766). Springer US. https://doi.org/10.1007/978-0-387-30164-8_631

Joshi, M. A. P., & Patel, B. V. (2021). Data Preprocessing: The Techniques for Preparing Clean and Quality Data for Data Analytics Process. *Oriental Journal of Computer Science and Technology*, *13*(2,3), 78–81.

Joshua, T., & Preethi, N. (2022, May 27). *Customer Segmentation in the Field of Marketing*. https://ieeexplore.ieee.org/document/9781964

Kadhim, A. I., & Jassim, A. K. (2021). Combined Chi-Square with k-Means for Document Clustering. *IOP Conference Series: Materials Science and Engineering*, *1076*(1), 012044. https://doi.org/10.1088/1757-899X/1076/1/012044

Kamil, F., Salkiawati, R., & Alexander, A. D. (2023). Analisis Churn Pelanggan Produk Fashion Campus Menggunakan Metode RFM Analysis dan Algoritma Naïve Bayes (Studi Kasus Yayasan Bakti Achmad Zaky). *Jurnal Esensi Infokom : Jurnal Esensi Sistem Informasi Dan Sistem Komputer*, *7*(2), Article 2. https://doi.org/10.55886/infokom.v7i2.629

Kiangala, S. K., & Wang, Z. (2021). An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment. *Machine Learning with Applications*, *4*, 100024. https://doi.org/10.1016/j.mlwa.2021.100024

Kim, T. K. (2017). Understanding one-way ANOVA using conceptual figures. *Korean Journal of Anesthesiology*, *70*(1), 22–26. https://doi.org/10.4097/kjae.2017.70.1.22

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, 1137–1143.

Kumari, P. S. (2022). Customer segmentation. *The Fairchild Books Dictionary of Fashion*. https://doi.org/10.1002/9780470685815.ch5

Laksono, B., & Wulansari, I. (2021). PEMODELAN DAN PENERAPAN METODE RFM PADA ESTIMASI NILAI KONSUMEN (CUSTOMER LIFETIME VALUE) MENGGUNAKAN K-MEANS CLUSTERING MACHINE LEARNING. *Seminar Nasional Official Statistics*, *2020*, 1277–1285. https://doi.org/10.34123/semnasoffstat.v2020i1.689

LaRose, R., & Coyle, B. (2020). Robust data encodings for quantum classifiers. *Physical Review A*, *102*(3), 032420. https://doi.org/10.1103/PhysRevA.102.032420

Lee, E., Kim, B., Kang, S., Kang, B., Jang, Y., & Kim, H. K. (2018). Profit Optimizing Churn Prediction for Long-Term Loyal Customers in Online Games. *IEEE Transactions on Games*, *PP*, 1–1. https://doi.org/10.1109/TG.2018.2871215

Liu, X., Rokunojjaman, M., Kumar Er, R., Nazila, R., & Vugar, A. (2023). Study on Intelligent Analysis and Processing Technology of Computer BigData Based on Clustering Algorithm. *Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering)*, *16*(2), 150–158. https://doi.org/10.2174/2352096515666220823093929

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129–137. IEEE Transactions on Information Theory. https://doi.org/10.1109/TIT.1982.1056489

Lu, C., Wang, D., Zhang, D., & Yu, A. (2023). *Speech image data mining algorithm based on multimodal decision fusion*. 19–24. https://doi.org/10.1145/3590003.3590007

Ma, S. (2009). *On Optimal Time for Customer Retention in Non-Contractual Setting* (SSRN Scholarly Paper 1529284). https://papers.ssrn.com/abstract=1529284

Madzík, P., Čarnogurský, K., Hrnčiar, M., & Zimon, D. (2021). Comparison of demographic, geographic, psychographic and behavioural approach to customer segmentation. *International Journal of Services and Operations Management*, *40*(3), 346–371. https://doi.org/10.1504/IJSOM.2021.119802

Manai, E., Mejri, M., & Fattahi, J. (2023). *Impact of Feature Encoding on Malware Classification Explainability* (arXiv:2307.05614). arXiv. https://doi.org/10.48550/arXiv.2307.05614

Marisa, F., Zahma, A., Bau, A. M., Noviansa, E., Neno, A. S., & Anastasia, A. (2021). Digitizing Paddy Harvest Productivity Based on K-Means Clustering. *SMARTICS Journal*, *7*(1), Article 1. https://doi.org/10.21067/smartics.v7i1.5270

Markos, A., Moschidis, O., & Chadjipadelis, T. (2020). Sequential dimension reduction and clustering of mixed-type data. *International Journal of Data Analysis Techniques and Strategies*, *12*, 28–30. https://doi.org/10.1504/IJDATS.2020.10028842

Marukatat, S. (2023). Tutorial on PCA and approximate PCA and approximate kernel PCA. *Artificial Intelligence Review*, *56*(6), 5445–5477. https://doi.org/10.1007/s10462-022-10297-z

Masarifoğlu, M., & Buyuklu, A. (2019). Applying Survival Analysis to Telecom Churn Data. *American Journal of Theoretical and Applied Statistics*, *8*, 261. https://doi.org/10.11648/j.ajtas.20190806.18

Mcdonald, M. (2010). Viewpoint: Existentialism – A School of Thought Based on a Conception of the Absurdity of the Universe. *International Journal of Market Research*, *52*(4), 427–430. https://doi.org/10.2501/S1470785309201363

McNicholas, P. D. (2016). Model-Based Clustering. *Journal of Classification*, *33*(3), 331–373. https://doi.org/10.1007/s00357-016-9211-9

Mohamed, A. S. T., Adnan, N. I. M., Husain, Q. N., & Kamarudin, A. N. (2023, September 17). *PERFORMANCE OF 4253HT SMOOTHER BY DIFFERENT HANNINGS: APPLICATION IN RAINFALL DATA | Jurnal Teknologi*. https://journals.utm.my/jurnalteknologi/article/view/19720

Mohammadzadeh, M., Hoseini, Z. Z., & Derafshi, H. (2017). A data mining approach for modeling churn behavior via RFM model in specialized clinics Case study: A public sector hospital in Tehran. *Procedia Computer Science*, *120*, 23–30. https://doi.org/10.1016/j.procs.2017.11.206

Nadarajan, R., & Sulaiman, N. (2023). Evaluation of K-fold value in breast cancer diagnosis technique using SVM and bio-inspired optimization algorithm (JA-ABC5). *2023 IEEE 13th Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, 130–135. 2023 IEEE 13th Symposium on Computer Applications & Industrial Electronics (ISCAIE). https://doi.org/10.1109/ISCAIE57739.2023.10165432

Nanda, A., Mahapatra, A., Mohapatra, B., & mahapatra, abinash. (2021). Multiple comparison test by Tukey's honestly significant difference (HSD): Do the confident level control type I error. *International Journal of Applied Mathematics and Statistics*, *6*, 59–65. https://doi.org/10.22271/maths.2021.v6.i1a.636

Nezampour, M., Rojuee, M., & Jafari Titkanloo, S. (2022). Market Segmentation of Nuts and Dried Fruits Based on Customers Expected Values Using DBSCAN Algorithm. *Commercial Surveys*, *19*(111), 45–68. https://doi.org/10.22034/bs.2022.47046

Nugroho, H. (2023). A Review: Data Quality Problem in Predictive Analytics. *IJAIT (International Journal of Applied Information Technology)*, *7*, 79. https://doi.org/10.25124/ijait.v7i02.5980

Nugroho, N., & Adhinata, F. (2022). Penggunaan Metode K-Means dan K-Means++ Sebagai Clustering Data Covid-19 di Pulau Jawa. *Teknika*, *11*, 170–179. https://doi.org/10.34148/teknika.v11i3.502

Olufemi Ogunleye, J. (2022). *The Concept of Data Mining*. *8*. https://doi.org/10.5772/intechopen.99417

Pandey, S., K, A., Shaikh, M. R., P, D. Y., & Vishwakarma, Y. (2023). Data Integration and Transformation using Artificial Intelligence. *2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, 844–849. https://doi.org/10.1109/IDCIoT56793.2023.10053513

Payne, A., & Frow, P. (2005). A Strategic Framework for Customer Relationship Management. *Journal of Marketing*, *69*(4), 167–176. https://doi.org/10.1509/jmkg.2005.69.4.167

Periáñez, Á., Saas, A., Guitart, A., & Magne, C. (2016). Churn Prediction in Mobile Social Games: Towards a Complete Assessment Using Survival Ensembles. *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 564–573. https://doi.org/10.1109/DSAA.2016.84

Qu, W., Xiu, X., Chen, H., & Kong, L. (2023). A Survey on High-Dimensional Subspace Clustering. *Mathematics*, *11*(2), Article 2. https://doi.org/10.3390/math11020436

Refialy, L. P., Maitimu, H., & Pesulima, M. (2021). Perbaikan Kinerja Clustering K-Means pada Data Ekonomi Nelayan dengan Perhitungan Sum of Square Error (SSE) dan Optimasi nilai K cluster. *Techno.Com*, *20*, 321–329. https://doi.org/10.33633/tc.v20i2.4572

Rodriguez, M. Z., Comin, C. H., Casanova, D., Bruno, O. M., Amancio, D. R., Costa, L. da F., & Rodrigues, F. A. (2019). Clustering algorithms: A comparative approach. *PLoS ONE*, *14*(1), e0210236. https://doi.org/10.1371/journal.pone.0210236

Ruddle, R. (2023). Using Well-Known Techniques to Visualize Characteristics of Data Quality: *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 89–100. https://doi.org/10.5220/0011664300003417

Sajid, A., & Amin, B. (2023, July 26). *Data Mining Techniques in Decision Making | Semantic Scholar*. https://www.semanticscholar.org/paper/Data-Mining-

Techniques-in-Decision-Making-Sajid-Amin/cd9f120679fa5cb4819db9e4e0e49c901982a6d0

Salihoun, M. (2020). State of Art of Data Mining and Learning Analytics Tools in Higher Education. *International Journal of Emerging Technologies in Learning (iJET)*, *15*, 58. https://doi.org/10.3991/ijet.v15i21.16435

Septiadi, A., & Ramadhani, W. K. (2020). Penerapan Metode Anova untuk Analisis Rata-rata Produksi Donat, Burger, dan Croissant pada Toko Roti Animo Bakery. *Bulletin of Applied Industrial Engineering Theory*, *1*(2), Article 2. https://jim.unindra.ac.id/index.php/baiet/article/view/2845

Singh, U. K., & Rout, M. (2023). Genetic Algorithm based Feature Selection to Enhance Breast Cancer Classification. *2023 IEEE International Conference on Contemporary Computing and Communications (InC4)*, *1*, 1–5. https://doi.org/10.1109/InC457730.2023.10263100

Singhal, R., & Rana, R. (2015). Chi-square test and its application in hypothesis testing. *Journal of the Practice of Cardiovascular Sciences*, *1*. https://doi.org/10.4103/2395-5414.157577

Smith, W. R. (1956). Product Differentiation and Market Segmentation as Alternative Marketing Strategies. *Journal of Marketing*, *21*(1), 3–8. https://doi.org/10.1177/002224295602100102

Soni, A., Arora, C., Kaushik, R., & Upadhyay, V. (2023). Evaluating the Impact of Data Quality on Machine Learning Model Performance. *Journal of Nonlinear Analysis and Optimization*, *14*, 13–18. https://doi.org/10.36893/JNAO.2023.V14I1.0013-0018

Sreejit Ramakrishnan. (2023). The Importance of Data Mining & Predictive Analysis. *International Journal of Engineering Technology and Management Sciences*, *7*(4), 593–598. https://doi.org/10.46647/ijetms.2023.v07i04.081

Steinhaus, H. (1956). *Sur la division des corps matériels en parties. Bull. Acad. Polon. Sci, 1(804), 801.* https://scholar.google.com/scholar_lookup?title=Sur%20la%20division%20des%20corps%20mat%C3%A9riels%20en%20parties&publication_year=1956&author=H.%20Steinhaus

Sud, K., Erdogmus, P., & Kadry, S. (2020). *Introduction to Data Science and Machine Learning*. BoD – Books on Demand.

Sullivan, L. (2019). *Hypothesis Testing—Analysis of Variance (ANOVA)*. https://sphweb.bumc.bu.edu/otlt/mph-modules/bs/bs704_hypothesistesting-anova/bs704_hypothesistesting-anova_print.html

Sun, Y., Liu, H., & Gao, Y. (2023). Research on customer lifetime value based on machine learning algorithms and customer relationship management analysis model. *Heliyon*, *9*(2), e13384. https://doi.org/10.1016/j.heliyon.2023.e13384

Syakur, M. A., Khotimah, B. K., Rochman, E. M. S., & Satoto, B. D. (2018). Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster. *IOP Conference Series: Materials Science and Engineering*, *336*(1), 012017. https://doi.org/10.1088/1757-899X/336/1/012017

Tsou, H.-T., & Huang, Y.-W. (2018). Empirical Study of the Affecting Statistical Education on Customer Relationship Management and Customer Value in Hi-tech Industry. *Eurasia Journal of Mathematics, Science and Technology Education*, *14*(4), 1287–1294. https://doi.org/10.29333/ejmste/81295

Tsoulfidis, L., & Athanasiadis, I. (2022). A new method of identifying key industries: A principal component analysis. *Journal of Economic Structures*, *11*(1), 2. https://doi.org/10.1186/s40008-022-00261-z

Ullah, I., Raza, B., Malik, A., Imran, M., Islam, S., & Kim, S. W. (2019). A Churn Prediction Model Using Random Forest: Analysis of Machine Learning Techniques for

Churn Prediction and Factor Identification in Telecom Sector. *IEEE Access*, *PP*, 1–1. https://doi.org/10.1109/ACCESS.2019.2914999

Umagapi, I. T., Umaternate, B., Hazriani, H., & Yuyun, Y. (2023). Uji Kinerja K-Means Clustering Menggunakan Davies-Bouldin Index Pada Pengelompokan Data Prestasi Siswa. *Prosiding SISFOTEK*, *7*(1), Article 1.

.V, K., Go, D., Sarveshwaran, V., J J, J., Kiruthika, M., & Remu, Y. (2023). *Comparative Analysis of Diverse Classification Algorithms of Machine Learning by Using Various Quality Metrics*. 551–556. https://doi.org/10.1109/ICCCMLA58983.2023.10346683

Vicedo, P., Gil-Gómez, H., Oltra-Badenes, R., & Guerola-Navarro, V. (2020). A bibliometric overview of how critical success factors influence on enterprise resource planning implementations. *Journal of Intelligent & Fuzzy Systems*, *38*(5), 5475–5487. https://doi.org/10.3233/JIFS-179639

Wang, C. (2019). *BEYOND TECHNOLOGY: DESIGN A VALUE-DRIVEN INTEGRATIVE PROCESS MODEL FOR DATA ANALYTICS THE A2E PROCESS MODEL FOR DATA ANALYTICS*.

Wang, C.-M. J. (1994). On estimating approximate degrees of freedom of chi-squared approximations. *Communications in Statistics-Simulation and Computation - COMMUN STATIST-SIMULAT COMPUT*, *23*, 769–788. https://doi.org/10.1080/03610919408813198

Wang, X., Liesaputra, V., Liu, Z., Wang, Y., & Huang, Z. (2024). An in-depth survey on Deep Learning-based Motor Imagery Electroencephalogram (EEG) classification. *Artificial Intelligence in Medicine*, *147*, 102738. https://doi.org/10.1016/j.artmed.2023.102738

Wedel, M., & Kamakura, W. A. (2000). *Market Segmentation: Conceptual and Methodological Foundations*. Springer Science & Business Media.

Wen, J., Xu, Y., & Liu, H. (2020). Incomplete Multiview Spectral Clustering With Adaptive Graph Learning. *IEEE Transactions on Cybernetics*, *50*(4), 1418–1429. https://doi.org/10.1109/TCYB.2018.2884715

Wen, T. (2020). Data Aggregation. In L. A. Schintler & C. L. McNeely (Eds.), *Encyclopedia of Big Data* (pp. 1–4). Springer International Publishing. https://doi.org/10.1007/978-3-319-32001-4_296-1

Xie, B., & Zhang, F. (2022). Design and Implementation of Data Mining in Information Management System. *2022 International Conference on Knowledge Engineering and Communication Systems (ICKES)*, 1–5. 2022 International Conference on Knowledge Engineering and Communication Systems (ICKECS). https://doi.org/10.1109/ICKECS56523.2022.10060029

Zhang, K., Wu, L., Lv, G., Chen, E., Ruan, S., Liu, J., Zhang, Z., Zhou, J., & Wang, M. (2023). *Description-Enhanced Label Embedding Contrastive Learning for Text Classification* (arXiv:2306.08817). arXiv. https://doi.org/10.48550/arXiv.2306.08817

Zhang, S., Tan, X., Wang, J., Chen, J., & Lai, X. (2019). Modeling Customers' Loyalty Using Ten Years' Automobile Repair and Maintenance Data: Machine Learning Approaches. *Proceedings of the 2019 2nd International Conference on Data Science and Information Technology*, 242–248. https://doi.org/10.1145/3352411.3352449

LAMPIRAN

Lampiran 1. Data Primer Penelitian

FrameSerialNo	WorkOrderNo	WorkOrderDate	ServiceInvoice	InvoiceDate	DealerType	VehicleUnit	WorkOrderStatus	CurrentMileageRecord	VehicleModel	Dealer	JobPrice	Revenue
MHKM5EA3JHK065000	20102/SWO/18/04/00095	2018-04-03 11:28:14	20102/INV/19/01/00432	2019-01-12 09:13:23	External	B-2343-SZH	Settlement	10506.0	AVANZA	TUNAS TOYOTA	175000.00	7.734091e+05
MHKM5EA3JGK009432	20102/SWO/18/04/00175	2018-04-05 11:00:17	20102/INV/19/02/00259	2019-02-09 10:29:48	Internal	DD-1290-UU	Settlement	49448.0	AVANZA	HK.GSO	127273.73	6.609094e+05
MHKM5EA3JHK061292	20102/SWO/18/04/01120	2018-04-28 11:40:39	20102/INV/19/01/00865	2019-01-23 09:29:23	Internal	DD-8255-XY	Settlement	19937.0	AVANZA	HK.GSO	127272.73	6.609091e+05
MHFJW8EM5G2304991	20102/SWO/18/05/00320	2018-05-09 08:15:50	20102/INV/19/10/00181	2019-10-04 16:25:18	Internal	DD-1037-UU	Waiting For Payment	39848.0	INNOVA	HK.URIP	230000.00	5.750000e+05
MHKA4DA3JGJ087767	20102/SWO/18/05/01282	2018-05-30 14:52:59	20102/INV/19/01/00866	2019-01-23 09:30:44	Internal	DD-1071-SK	Settlement	49110.0	AGYA	HK.GSO	127272.73	1.654545e+06
MHKM5EA3JHK082617	20102/SWO/18/06/00473	2018-06-11 14:15:00	20102/INV/19/01/00527	2019-01-15 15:55:19	Internal	DD-1878-SS	Settlement	18997.0	AVANZA	HK.GSO	127272.73	6.609091e+05
MHKA4DA3JGJ099666	20102/SWO/18/07/00296	2018-07-07 09:58:41	20102/INV/19/02/00872	2019-02-22 14:27:46	Internal	DD-1324-IB	Settlement	49684.0	AGYA	HK.GSO	127273.73	1.654546e+06
MHFJW8EM6G2304871	20102/SWO/18/07/00430	2018-07-10 09:38:08	20102/INV/19/02/00653	2019-02-18 15:32:41	Internal	DD-1017-MC	Settlement	51882.0	INNOVA	HK COKRO	145454.55	2.007980e+06
MHKM5FB4JGK007437	20102/SWO/18/07/00633	2018-07-16 09:50:02	20102/INV/19/02/00419	2019-02-13 15:17:50	Internal	DD-1300-XQ	Settlement	41735.0	AVANZA	HK.GSO	822727.00	2.280252e+06
MHFJW8EM0F2300567	20102/SWO/18/10/00121	2018-10-03 12:09:53	20102/INV/19/06/00475	2019-06-18 11:00:13	Internal	DD-3102-MI	Settlement	21588.0	INNOVA	HK COKRO	230000.00	6.670000e+05

Lampiran 2. Codingan Segmentasi dan Random Forest

```
import numpy as np
import pandas as pd
import datetime as dt
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from yellowbrick.cluster import KElbowVisualizer
from scipy.stats import probplot
from yellowbrick.cluster import SilhouetteVisualizer
from imblearn.under_sampling import RandomUnderSampler
from imblearn.over_sampling import SMOTE
from sklearn.metrics import confusion_matrix
import plotly.express as px
from sklearn.metrics import classification_report
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import RobustScaler
from sklearn.naive_bayes import GaussianNB
from datetime import date
from lifetimes import BetaGeoFitter, GammaGammaFitter
from lifetimes.utils import \
    calibration_and_holdout_data, \
    summary_data_from_transaction_data, \
    calculate_alive_path
from lifetimes.plotting import \
    plot_frequency_recency_matrix, \
    plot_probability_alive_matrix, \
    plot_period_transactions, \
    plot_history_alive, \
    plot_cumulative_transactions, \
    plot_calibration_purchases_vs_holdout_purchases
import warnings
warnings.filterwarnings('ignore')

pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)
df_data = pd.read_csv('/content/drive/MyDrive/PENELITIAN/preprocessed_data
(1).csv')

missing_values = df_data.isnull().sum()
# Kolom Missing Value
print("Kolom Missing value")
print(missing_values[missing_values > 0])

kolom = ['VehicleModel', 'CreditNoteNo', 'ProgramName', 'DiscountType']
for col in kolom:
    df_data[col] = df_data[col].fillna("None")

missing_values = df_data.isnull().sum()
# Kolom Missing Value
print("Kolom Missing value")
print(missing_values[missing_values > 0])
```

```
customer = pd.read_excel('/content/drive/MyDrive/Predictive Sparepart
Maintenance/dataset/Data Customer.xlsx')
customer.head(10)

missing_values = customer.isnull().sum()
# Kolom Missing Value
print("Kolom Missing value")
print(missing_values[missing_values > 0])

df_data['InvoiceDate'].isna().sum()
df_data.head(5)
df_data['InvoiceDate'] = pd.to_datetime(df_data['InvoiceDate'])

count_per_tahun = df_data['InvoiceDate'].dt.year.value_counts().sort_index()
# Membuat bar plot
plt.figure(figsize=(10, 6))
bars = count_per_tahun.plot(kind='bar', color='skyblue')
plt.title('Jumlah Pengamatan per Tahun')
plt.xlabel('Tahun')
plt.ylabel('Jumlah Pengamatan')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)
# Menambahkan label jumlah di atas setiap bar
for bar in bars.patches:
    plt.text(bar.get_x() + bar.get_width()/2 - 0.2, bar.get_height() + 10,
str(int(bar.get_height())),
             ha='center', va='bottom', color='black', fontsize=9)
plt.show()

# Menggabungkan dataframe
df_com = df_data.merge(customer, on='FrameSerialNo', how='inner')
# Menampilkan DataFrame hasil
df_com.shape

# Pembentukan Nilai RFM dan T
today_date = dt.datetime(2023, 1, 31)

cltv_df = df_com.groupby('FrameSerialNo').agg({
    'InvoiceDate_x': [
        lambda x: (x.max() - x.min()).days,  # recency
        lambda x: (today_date - x.min()).days  # T
    ],
    'ServiceInvoice': lambda x: x.nunique(),  # frequency
    'Revenue': lambda x: x.sum()  # monetary
})

cltv_df.columns = cltv_df.columns.droplevel(0)
cltv_df.columns = ['recency', 'T', 'frequency', 'monetary']
cltv_df = cltv_df[cltv_df['monetary'] > 0]
# nilai moneter adalah pendapatan rata-rata per transaksi
cltv_df['monetary'] = cltv_df['monetary'] / cltv_df['frequency']
# mengubah hari menjadi minggu
cltv_df['recency'] = cltv_df['recency'] / 7
cltv_df['T'] = cltv_df['T'] / 7

# Model Beta Geometri
```

```
bgf = BetaGeoFitter(penalizer_coef=1e-06)
bgf.fit(cltv_df['frequency'],
        cltv_df['recency'],
        cltv_df['T'])

# Melihat eksperasi transaksi 1 minggu dan 1 bulan kedepan
bgf.conditional_expected_number_of_purchases_up_to_time(1,
                                                        cltv_df['frequency']
,
                                                        cltv_df['recency'],
                                                        cltv_df['T']).sort_v
alues(ascending=False).head(10)
cltv_df["expected_purc_1_week"] = bgf.predict(1,
                                              cltv_df['frequency'],
                                              cltv_df['recency'],
                                              cltv_df['T'])
bgf.predict(4,
            cltv_df['frequency'],
            cltv_df['recency'],
            cltv_df['T']).sort_values(ascending=False).head(10)
cltv_df["expected_purc_1_month"] = bgf.predict(4,
                                               cltv_df['frequency'],
                                               cltv_df['recency'],
                                               cltv_df['T'])
cltv_df.sort_values("expected_purc_1_month", ascending=False).head(10)

# Model Gamma Gamma
ggf = GammaGammaFitter(penalizer_coef=0.01)
ggf.fit(cltv_df['frequency'], cltv_df['monetary'])
# 10 pelanggan paling teratas
ggf.conditional_expected_average_profit(cltv_df['frequency'],
                                        cltv_df['monetary']).sort_values(asc
ending=False).head(10)
cltv_df["expected_average_profit"] =
ggf.conditional_expected_average_profit(cltv_df['frequency'],

 cltv_df['monetary'])
cltv_df.sort_values("expected_average_profit", ascending=False).head(10)

# Pembentukan Nilia CLTV
cltv = ggf.customer_lifetime_value(bgf,
                                   cltv_df['frequency'],
                                   cltv_df['recency'],
                                   cltv_df['T'],
                                   cltv_df['monetary'],
                                   time=6,   # 6 month
                                   freq="W", # Weekly
                                   discount_rate=0.01)
cltv.head()
cltv = cltv.reset_index()
cltv.sort_values(by= "CLTV", ascending=False).head()
# 50 pelanggan paling berharga dalam 6 bulan
cltv.sort_values(by="CLTV", ascending=False).head(20)
cltv_final.describe()
cltv_final.head(2)
```

```
latest_transactions =
df_com.loc[df_com.groupby('FrameSerialNo')['InvoiceDate_x'].idxmax()]
# Menggabungkan CreditNoteNo dan DiscountType dengan koma
combined_data = df_com.groupby('FrameSerialNo').agg({
    'CreditNoteNo': lambda x: ','.join(x.dropna().astype(str)),
    'DiscountType': lambda x: ','.join(x.dropna().astype(str))
}).reset_index()
# Gabungkan data transaksi terbaru dengan data yang digabungkan
result = pd.merge(latest_transactions, combined_data, on='FrameSerialNo',
how='left')
# Menampilkan hasil
print("Data pelanggan dengan transaksi terbaru:")

df_use = result[['FrameSerialNo', 'Branch_x',
                 'DealerType','CurrentMileageRecord',
                 'Branch_y','Provinsi'
                 ]]

# Pembentukan kategori lokasi
dalam_kota = ['URIP SUMOHARJO GENERAL REPAIR', 'URIP SUMOHARJO BODY PAINT',
'SERUI GENERAL REPAIR','ALAUDDIN GENERAL REPAIR']
# Fungsi untuk mengkategorikan cabang
def categorize_branch(branch):
    if branch in dalam_kota:
        return 'DalamKota'
    else:
        return 'LuarKota'
# Terapkan fungsi untuk membuat kolom baru 'kategori lokasi'
df_use_filtered['ServiceLocationCategory'] =
df_use_filtered['Branch_x'].apply(categorize_branch)
df_use_filtered['DealerBeliLocationCategory'] =
df_use_filtered['Branch_y'].apply(categorize_branch)

df_use1 = df_use_filtered[['FrameSerialNo',
                           'ServiceLocationCategory',
                           'DealerBeliLocationCategory',
                           'DealerType',
                           'CurrentMileageRecord'
                           ]]

# Menggabungkan dataframe CLTV dan dataframe pelanggan
df_usage = pd.merge(df_use1, cltv_final, on='FrameSerialNo', how='inner')

# Mengecek outlier
numeric_columns = df_usage.select_dtypes(include=[np.number]).columns
num_cols = len(numeric_columns)
fig, axes = plt.subplots(nrows=num_cols, ncols=1, figsize=(20, num_cols *
5))
# Membuat boxplot untuk setiap kolom numerik
for i, col in enumerate(numeric_columns):
    sns.boxplot(x=df_usage[col], ax=axes[i])
    axes[i].set_title(f'Boxplot for {col}')

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

```
# Mengidentifikasi kolom numerik
numeric_columns = df_usage.select_dtypes(include=[np.number]).columns
# Ulangi setiap kolom angka
for col in numeric_columns:
    # Hitung rentang interkuartil (IQR)
    Q1 = df_usage[col].quantile(0.25)
    Q3 = df_usage[col].quantile(0.75)
    IQR = Q3 - Q1
    # Tentukan batas outlier
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    # Ganti outlier dengan batas terdekat
    df_usage[col] = df_usage[col].clip(lower=lower_bound, upper=upper_bound)
# Menampilkan hasil data
df_usage.head()

# Kembali melihat hasil dari handling outlier
numeric_columns = df_usage.select_dtypes(include=[np.number]).columns
num_cols = len(numeric_columns)
fig, axes = plt.subplots(nrows=num_cols, ncols=1, figsize=(20, num_cols *
5))
# Membuat boxplot untuk setiap kolom numerik
for i, col in enumerate(numeric_columns):
    sns.boxplot(x=df_usage[col], ax=axes[i])
    axes[i].set_title(f'Boxplot for {col}')
plt.tight_layout()
plt.show()

# Melakukan Label Encode pada data kategori
from sklearn.preprocessing import LabelEncoder
label_encoders = {}
categorical_columns = ['ServiceLocationCategory',
'DealerBeliLocationCategory', 'DealerType']
for col in categorical_columns:
    le = LabelEncoder()
    df_usage[col] = le.fit_transform(df_usage[col])
    label_encoders[col] = le

# Melihat mapping dari label endcode
for col in categorical_columns:
    le = label_encoders[col]
    print(f"Mapping for {col}:")
    for class_index in range(len(le.classes_)):
        print(f"{class_index} -> {le.classes_[class_index]}")

from sklearn.preprocessing import MinMaxScaler
# Menyimpan kolom FrameSerialNo
frame_serial_no = df_usage['FrameSerialNo']
# Menghapus kolom FrameSerialNo dari df_usage
df_usage_noframe = df_usage.drop(['FrameSerialNo'], axis=1)
# Inisialisasi MinMaxScaler
scaler = MinMaxScaler(feature_range=(0, 1))
# Melakukan scaling pada fitur
scaled_features = scaler.fit_transform(df_usage_noframe)

from sklearn.decomposition import PCA
```

```
# Reduksi kolom dengan melihat hasil PCA
pca_full = PCA()
pca_full.fit(scaled_features)

# Hitung rasio varians kumulatif yang dijelaskan
explained_variance_ratio = pca_full.explained_variance_ratio_
cumulative_explained_variance = explained_variance_ratio.cumsum()
# Gambarkan rasio varians kumulatif yang dijelaskan untuk menemukan jumlah
komponen yang optimal
plt.figure(figsize=(10, 6))
plt.plot(range(1, len(cumulative_explained_variance) + 1),
cumulative_explained_variance, marker='o', linestyle='--')
plt.title('Cumulative Explained Variance by PCA Components')
plt.xlabel('Number of PCA Components')
plt.ylabel('Cumulative Explained Variance')
plt.grid(True)
plt.axhline(y=0.95, color='r', linestyle='-')  # 95% variance line for
reference
plt.text(0.5, 0.85, '95% cut-off threshold', color = 'red', fontsize=16)
# Tentukan jumlah komponen yang menjelaskan setidaknya 95% varians
optimal_num_components =
len(cumulative_explained_variance[cumulative_explained_variance >= 0.95]) +
1
# Sorot jumlah komponen optimal pada plot
plt.axvline(x=optimal_num_components, color='g', linestyle='--')
plt.text(optimal_num_components + 1, 0.6, f'Optimal Components:
{optimal_num_components}', color = 'green', fontsize=14)
plt.show()
# Mengembalikan jumlah komponen yang optimal
optimal_num_components

optimal_num_components = 6
# Menerapkan PCA dengan jumlah komponen yang optimal
pca = PCA(n_components=optimal_num_components)
pca.fit_transform(scaled_features)
pca.transform(scaled_features)
pca_result = pca.fit_transform(scaled_features)

# Terapkan PCA dengan jumlah komponen optimal
pca = PCA(n_components=optimal_num_components)
pca_result = pca.fit_transform(scaled_features)
# Verifikasi hasil PCA
print(f"Shape hasil PCA: {pca_result.shape}")
# Dapatkan komponen PCA (loadings)
pca_components = pca.components_
# Buat DataFrame untuk visualisasi dan analisis lebih lanjut
pca_loadings_df = pd.DataFrame(pca_components,
columns=df_usage_noframe.columns, index=[f'PC{i+1}' for i in
range(optimal_num_components)])
# Tampilkan loadings
print(pca_loadings_df)
# Heatmap dari loadings
plt.figure(figsize=(12, 6))
sns.heatmap(pca_loadings_df, cmap="YlGnBu", annot=True)
plt.title('PCA Loadings')
plt.show()
```

```
# Creating a 2D plot for the first two PCA components
plt.figure(figsize=(10, 8))
plt.scatter(pca_result[:, 0], pca_result[:, 1])
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.title('2D PCA Results')
plt.grid(True)
plt.show()

# Extract the absolute values of the PCA loadings
pca_loadings_analysis = pd.DataFrame(
    np.abs(pca.components_),
    columns=df_usage_noframe.columns,
    index=[f'PC{i+1}' for i in range(pca.n_components)]
)
# Identify the top contributing features for each principal component
top_features_per_pc = pca_loadings_analysis.apply(lambda s:
s.nlargest(5).index.tolist(), axis=1)
pd.options.display.max_colwidth = 200
# Display the top contributing features for each principal component
top_features_per_pc

# Optimal_num_components seharusnya sesuai dengan jumlah komponen di
pca_result
optimal_num_components = pca_result.shape[1]  # Menyesuaikan dengan hasil
PCA yang sebenarnya
# Membuat DataFrame dari hasil PCA dan menambahkan kembali kolom
'FrameSerialNo'
pca_df = pd.DataFrame(pca_result, columns=[f'PC{i+1}' for i in
range(optimal_num_components)])
pca_df['FrameSerialNo'] = frame_serial_no  # Menambahkan kembali kolom
'FrameSerialNo'
# Menampilkan beberapa baris pertama dari DataFrame dengan nilai PCA
pca_df.head()

pip install kneed matplotlib scikit-learn
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px

# Class KMeansClustering
class KMeansClustering:
    def __init__(self, X, num_clusters):
        self.K = num_clusters  # number of clusters
        self.max_iterations = 100  # max iterasi
        self.num_examples, self.num_features = X.shape  # number of
examples, number of features
        self.plot_figure = True  # plot figure

    #  inisialisasi random centroids
    def initialize_random_centroids(self, X):
        centroids = np.zeros((self.K, self.num_features))  # row, column
full with zeros
        for k in range(self.K):
```

```
            centroid = X[np.random.choice(range(self.num_examples))]  #
random centroids
            centroids[k] = centroid
        return centroids  # return random centroids

    # create clusters
    def create_clusters(self, X, centroids):
        clusters = [[] for _ in range(self.K)]
        for point_idx, point in enumerate(X):
            closest_centroid = np.argmin(np.sqrt(np.sum((point - centroids)
** 2, axis=1)))  # centroid terdekat
            clusters[closest_centroid].append(point_idx)
        return clusters

    # calculate new centroids
    def calculate_new_centroids(self, clusters, X):
        centroids = np.zeros((self.K, self.num_features))  # row, column
full with zeros
        for idx, cluster in enumerate(clusters):
            new_centroid = np.mean(X[cluster], axis=0)  # find the value for
new centroids
            centroids[idx] = new_centroid
        return centroids

    # predict clusters
    def predict_clusters(self, clusters, X):
        y_pred = np.zeros(self.num_examples)  # row filled with zeros
        for cluster_idx, cluster in enumerate(clusters):
            for sample_idx in cluster:
                y_pred[sample_idx] = cluster_idx
        return y_pred

    # plot scatter plot
    def plot_fig(self, X, y):
        fig = px.scatter(X[:, 0], X[:, 1], color=y)
        fig.show()  # visualize

    # fit data
    def fit(self, X):
        centroids = self.initialize_random_centroids(X)  # initialisasi
random centroids
        for _ in range(self.max_iterations):
            clusters = self.create_clusters(X, centroids)  # create clusters
            previous_centroids = centroids
            centroids = self.calculate_new_centroids(clusters, X)  #
menghitung new centroids
            diff = centroids - previous_centroids  # menghitung difference
            if not diff.any():
                break
        y_pred = self.predict_clusters(clusters, X)  # predict fungsi
        if self.plot_figure:  # if true
            self.plot_fig(X, y_pred)  # plot fungsi
        return y_pred

# mendefinisikan Elbow Method
def elbow_method(data, max_k):
```

```
    inertias = []
    for k in range(1, max_k + 1):
        kmeans = KMeansClustering(data, k)
        y_pred = kmeans.fit(data)
        inertia = 0
        for i in range(k):
            cluster_points = data[y_pred == i]
            centroid = np.mean(cluster_points, axis=0)
            inertia += np.sum((cluster_points - centroid) ** 2)
        inertias.append(inertia)
    return inertias

# mendefinisikan Silhouette Score
def silhouette_score(data, y_pred):
    n_samples = len(data)
    unique_clusters = np.unique(y_pred)
    n_clusters = len(unique_clusters)

    if n_clusters == 1 or n_clusters == n_samples:
        return -1

    silhouette_avg = 0
    for i in range(n_samples):
        same_cluster = data[y_pred == y_pred[i]]
        other_clusters = [data[y_pred == label] for label in unique_clusters
if label != y_pred[i]]

        a = np.mean([np.linalg.norm(data[i] - point) for point in
same_cluster if not np.array_equal(data[i], point)])
        b = np.min([np.mean([np.linalg.norm(data[i] - point) for point in
other_cluster]) for other_cluster in other_clusters])

        silhouette_avg += (b - a) / max(a, b)

    return silhouette_avg / n_samples

np.random.seed(37)
# arr_usage adalah hasil PCA
arr_usage = scaled_df.values  # Gunakan data yang sudah diskalakan tanpa
indeks

# Elbow Method
inertias = elbow_method(arr_usage, 10)

plt.plot(range(1, 11), inertias, 'bx-')
plt.xlabel('Values of K')
plt.ylabel('Inertia')
plt.title('The Elbow Method using Inertia')
plt.show()

# Silhouette Scores untuk berbagai jumlah cluster
silhouette_scores = []
for k in range(2, 11):
    kmeans = KMeansClustering(arr_usage, k)
    y_pred = kmeans.fit(arr_usage)
    score = silhouette_score(arr_usage, y_pred)
```

```
    silhouette_scores.append(score)

silhouette_df = pd.DataFrame({'Number of Clusters (K)': range(2, 11),
'Silhouette Score': silhouette_scores})
print(silhouette_df)
plt.plot(range(2, 7), silhouette_scores, 'bx-')
plt.xlabel('Number of Clusters (K)')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Scores for different K values')
plt.show()

# Mendeteksi knee point
kneedle = KneeLocator(K, inertias, curve='convex', direction='decreasing')
optimal_k_inertia = kneedle.elbow

# Plot Inertia
plt.figure(figsize=(8, 6))
plt.plot(K, inertias, 'bx-', marker='o', linestyle='--', color='blue')
plt.axvline(x=optimal_k_inertia, color='red', linestyle='--')
plt.xlabel('Values of K')
plt.ylabel('Inertia')
plt.title('Inertia for Different K')
plt.show()
print(f"Optimal number of clusters based on inertia: {optimal_k_inertia}")

# Optimal k berdasarkan silhouette score
optimal_k_silhouette = K[silhouette_scores.index(max(silhouette_scores))]
# Plot Silhouette Score
plt.figure(figsize=(8, 6))
plt.plot(K, silhouette_scores, 'gx-', marker='o', linestyle='--',
color='green')
plt.axvline(x=optimal_k_silhouette, color='red', linestyle='--')
plt.xlabel('Values of K')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score for Different K')
plt.show()
print(f"Optimal number of clusters based on silhouette score:
{optimal_k_silhouette}")

# Menentukan jumlah klaster dari 2 hingga 11
for k in range(2, 11):
    # Melakukan clustering menggunakan KMeans
    kmeans_model = KMeans(n_clusters=k, random_state=37)
    cluster_labels = kmeans_model.fit_predict(arr_usage)
    # Menghitung silhouette score
    silhouette_avg = silhouette_score(arr_usage, cluster_labels)
    # Menyimpan silhouette score
    silhouette_scores.append(silhouette_avg)
# Membuat DataFrame untuk menampilkan hasil silhouette scores
silhouette_df = pd.DataFrame({'Number of Clusters (K)': range(2, 11),
'Silhouette Score': silhouette_scores})
# Menampilkan tabel silhouette scores
print(silhouette_df)

plt.figure(figsize=(8, 6))
```

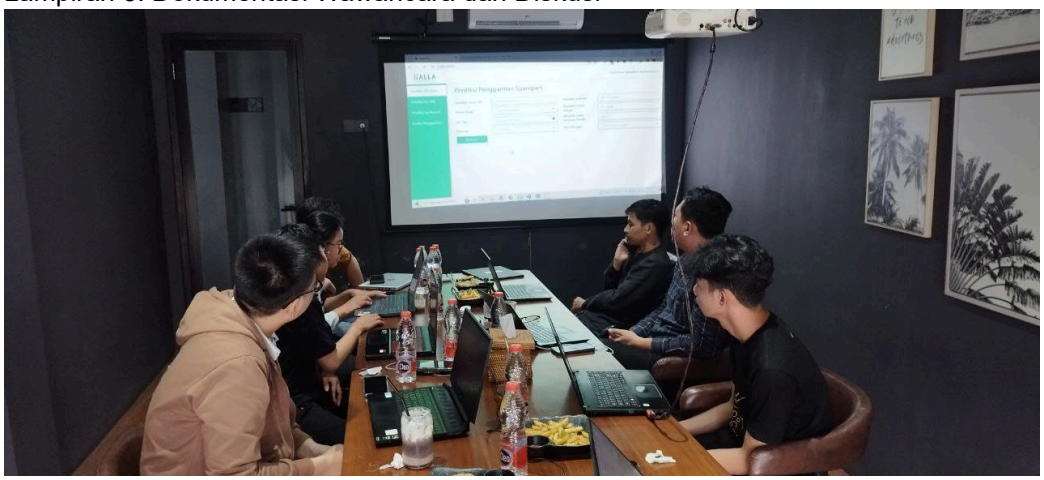
```
plt.plot(range(2, 11), silhouette_scores, marker='o', linestyle='--',
color='green')
plt.axvline(x=optimal_k_silhouette, color='red', linestyle='--')
plt.xlabel('Number of Clusters (K)')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Score for Different Number of Clusters')
plt.grid(True)
plt.show()

k =5
kmeans = KMeans(n_clusters=k)
labels_minmax = kmeans.fit_predict(arr_usage)
silhouette_minmax = silhouette_score(arr_usage, labels_minmax)
print(f"Silhoute Score Min Max: {silhouette_minmax}")

from sklearn.metrics import davies_bouldin_score, calinski_harabasz_score
# Menghitung score Davies-Bouldin Index
dbi_score = davies_bouldin_score(arr_usage, labels_minmax)
print(f"Davies-Bouldin Index: {dbi_score}")

# Mendapatkan centroids
centroids = kmeans.cluster_centers_
# Menampilkan centroids
print("Centroids:")
print(centroids)

df_seg_pca = pd.concat([df_usage.reset_index(drop=True),
pd.DataFrame(arr_usage)], axis=1)
# Rename the last 8 columns to 'Component 1', 'Component 2', ..., 'Component
dst'
df_seg_pca.columns = list(df_seg_pca.columns[:-6]) + [
    'Component 1',
    'Component 2',
    'Component 3',
    'Component 4',
    'Component 5',
    'Component 6'
]
# Add the cluster labels
df_seg_pca['Cluster_PCA'] = kmeans.labels_

centroids = kmeans.cluster_centers_

# Hitung jarak antara setiap titik data dan centroidnya
distances = []
for i in range(len(arr_usage)):
    dist = np.linalg.norm(arr_usage[i] - centroids[kmeans.labels_[i]])
    distances.append(dist)
# Buat DataFrame untuk menyimpan label dan jarak cluster
df_distances = pd.DataFrame({
    'Cluster': kmeans.labels_,
    'Distance_to_Centroid': distances
})
# Melihat the DataFrame
print(df_distances)
```

```
# Plot distribusi cluster
plt.figure(figsize=(8, 6))
sns.countplot(x='Cluster_PCA', data=df_seg_pca)
plt.title('Countplot of Clusters')
plt.xlabel('Cluster')
plt.ylabel('Count')
plt.show()

# Menghitung jumlah data pada masing-masing cluster
cluster_counts = df_seg_pca['Cluster_PCA'].value_counts()
# Membuat bar chart
plt.figure(figsize=(10, 6))
bars = plt.bar(cluster_counts.index, cluster_counts.values, color='green')
# Menambahkan label jumlah pada setiap bar
for bar in bars:
    yval = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2, yval, int(yval),
va='bottom')  # va='bottom' untuk menampilkan label di atas bar
plt.xlabel('Cluster')
plt.ylabel('Jumlah Data')
plt.title('Jumlah Data per Cluster')
plt.xticks(cluster_counts.index)  # Menambahkan label x-ticks
plt.show()

df_seg_pca.head(5)
columns_to_normalize = ['ServiceLocationCategory',
'DealerBeliLocationCategory', 'DealerType', 'CurrentMileageRecord',
'recency',
                        'T', 'frequency', 'monetary',
'expected_average_profit', 'CLTV']
# Memisahkan fitur dan label target
X = df_seg_pca[columns_to_normalize]
y = df_seg_pca['Cluster_PCA']
# Normalisasi data hanya pada fitur
scaler = MinMaxScaler(feature_range=(0,1))
X_scaled = scaler.fit_transform(X)
X_scaled_df = pd.DataFrame(X_scaled, columns=columns_to_normalize)
# Konversi label target menjadi kategori
y_cat = y.astype('category')
X_arr = X_scaled_df.to_numpy()
y_arr = y_cat.to_numpy()

from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
#kfold split data-5
kf = KFold(n_splits= 5)
fold_number = 1
for train_index, test_index in kf.split(X_arr):
  X_train = X.iloc[train_index]
  X_test = X.iloc[test_index]
  y_train = y_cat.iloc[train_index]
  y_test = y_cat.iloc [test_index]

# Print the size of the data for the current fold
print(f"Fold {fold_number}")
print(f"Number of samples in X_train: {X_train.shape[0]}")
```

```
print(f"Number of samples in X_test: {X_test.shape[0]}")
print(f"Number of samples in y_train: {y_train.shape[0]}")
print(f"Number of samples in y_test: {y_test.shape[0]}")
print("\n" + "-"*50 + "\n")
fold_number += 1

train_scores, test_scores = list(), list()
#parameter kedalaman pohon
values = [i for i in range(1, 13)]
for i in values:
  model = RandomForestClassifier(max_depth=i)
  model.fit(X_train, y_train)
  train_yhat = model.predict(X_train)
  train_acc = accuracy_score(y_train, train_yhat)
  train_scores.append(train_acc)
  test_yhat = model.predict(X_test)
  test_acc = accuracy_score(y_test, test_yhat)
  test_scores.append(test_acc)


plt.plot(values, train_scores, '-o', label='Train')
plt.plot(values, test_scores, '-o', label='Test')
plt.legend()
plt.show()

# Mencari Paramter terbaik
from sklearn.model_selection import RandomizedSearchCV as RSCV
param_grid = {'n_estimators': np.arange(50, 200, 10),
              'max_features': np.arange(0.1, 1, 0.1),
              'max_depth': [3, 5, 7, 9, 12],
              'max_samples': [0.3, 0.5, 0,8]}
model = RSCV(RandomForestClassifier(), param_grid, n_iter=15). fit(X_train,
y_train)
model = model.best_estimator_
print("Best Model:")
print(model)

# Model Random forest
from sklearn.metrics import classification_report
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(
    n_estimators= model.n_estimators,
    max_depth= model.max_depth,
    max_features = model.max_features,
    criterion='entropy',
    random_state= 42,
    oob_score=True,
    bootstrap= True)
rfc = rfc.fit(X_train, y_train)
y_pred_train = rfc.predict(X_train)
y_pred_test = rfc.predict(X_test)

# Menampilkan pohon keputusan
from sklearn import tree
plt.figure(figsize=(40, 15))
for i in range (len(rfc.estimators_)):
```

```
  tree.plot_tree(rfc.estimators_[i],
                 feature_names=X.columns, class_names=True, filled=True)

n_trees = 1  # kita hanya ingin melihat satu pohon
for i in range(n_trees):
    plt.figure(figsize=(20, 10))  # Atur ukuran gambar untuk setiap pohon
    tree.plot_tree(rfc.estimators_[i],
                   feature_names=X.columns,
                   class_names=True,
                   filled=True,
                   max_depth=3,  # Memplot hanya 3 level dari pohon
                   fontsize=10)
    plt.title(f'Tree {i + 1}')
    plt.show()

# Melihat Fitur paling berpengaruh
model.feature_importances_
imp_df = pd.DataFrame({
    'Varname': X_train.columns,
    'Imp': model.feature_importances_
})

imp_df.sort_values(by='Imp', ascending=False)

# Proses melihat akurasi dari k fold
scores = cross_val_score(rfc, X_scaled_df, y_cat, cv = 5)
# Menampilkan hasil akurasi untuk setiap fold
for i, score in enumerate(scores):
    print(f"Fold {i+1}: Accuracy = {score:.4f}")
# Menampilkan rata-rata akurasi dan standar deviasi
print(f"Mean Accuracy: {scores.mean():.4f}")
print(f"Standard Deviation: {scores.std():.4f}")

# Evaluasi model pada data train
print(classification_report(y_train, y_pred_train))
accuracy = accuracy_score(y_train, y_pred_train)
print("Akurasi train:", accuracy)
# Evaluasi model pada data test
print(classification_report(y_test, y_pred_test))
accuracy = accuracy_score(y_test, y_pred_test)
print("Akurasi test :", accuracy)

from sklearn.metrics import confusion_matrix
# Confusion Matrix untuk data train
conf_matrix_train = confusion_matrix(y_train, y_pred_train)
print("Confusion Matrix - Train:")
print(conf_matrix_train)
# Confusion Matrix untuk data test
conf_matrix_test = confusion_matrix(y_test, y_pred_test)
print("Confusion Matrix - Test:")
print(conf_matrix_test)
```

Lampiran 3. Dokumentasi Wawancara dan Diskusi



KEMENTERIAN PENDIDIKAN , KEBUDAYAAN
RISET DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA
Kampus FakultasTeknikUnhas, Jl. Poros Malino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

DAFTAR HADIR SEMINAR HASIL

Nama/Stambuk : 1. Maylinda Eka Christy Maylinda D121201050

Judul Skipsi/T.A : **" Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif Studi Kasus :Kalla Toyota)"**

Hari/Tanggal : Rabu, 28 Agustus 2024
Jam : 13.00 Wita – Selesai
Tempat : Ruang **Lab. AIMP** Departeman Teknik Informatika Gowa

No.	Jabatan	Nama Dosen	Tanda Tangan
L.	Pembimbing I	1. Dr.Ir.Ingrid Nurtanio,M.T	1.....
II.	Anggota Penguji	3. Novy Nur RA Mokobombang,ST.,Ms.,TM.,PhD	3 .....
		4. Ir. Anugrayani Butastamin,ST.,M.T	4 .....

PANITIA UJIAN

Ketua,

Dr.Ir.Ingrid Nurtanio,M.T

KEMENTERIAN PENDIDIKAN , KEBUDAYAAN
RISET DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA
Kampus FakultasTeknikUnhas, Jl. Poros Malino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

BERITA ACARA SEMINAR HASIL

Pada hari ini Jumat, tanggal **28 Agustus 2024** Pukul **13.00 WITA - Selesai** bertempat di **Ruang Lab.AIMP Departemen Teknik Informatika**, telah dilaksanakan Seminar Hasil bagi Saudara :

Nama : Maylinda Eka Christy Maylinda
No. Stambuk : D121201050
Fakultas/Departemen : Teknik/Teknik Informatika
Judul Skripsi : **" Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif Studi Kasus :Kalla Toyota)"**

Yang dihadiri oleh Tim Penguji Seminar Hasil sebagai berikut :

No.	N a m a	Jabatan	Tanda tangan
1.	Dr.Ir.Ingrid Nurtanio,M.T	Pemb I/Ketua	1.
2.	Novy Nur RA Mokobombang,ST.,Ms.,TM.,Ph.D	Anggota	2.
3.	Ir. Anugrayani Butastamin,ST.,M.T	Anggota	3.

Hasil keputusan Tim Penguji Seminar Hasil : **Lulus / Tidak lulus** dengan nilai angka ..86..... dan huruf ..A.....

Gowa, 28 Agustus 2024

Ketua/Sekretaris Panitia Ujian,

Dr.Ir.Ingrid Nurtanio,M.T

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA
Kampus Fakultas Teknik Unhas, Jl. Poros Malino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

Nomor : 1276/UN4.7.7/TD.06/2024
Lamp : -
Hal : Penerbitan Surat Penugasan Panitia/Penguji Seminar Hasil Strata Satu (S1)

Kepada Yth :

Wakil Dekan Bidang Akademik dan Kemahasiswaan
Fakultas Teknik Universitas Hasanuddin

Di-

Gowa

Dengan hormat,
Berdasarkan Persetujuan Pembimbing Mahasiswa, Bersama ini diusulkan susunan Panitia/Penguji Seminar Hasil Strata Satu (S1) bagi mahasiswa Departemen Teknik Informatika Fakultas Teknik tersebut di bawah ini :

Nama / Stambuk : Maylinda Eka Christy Maylinda D121201050
Judul TA : Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif Studi Kasus: Kalla Toyota)

Dengan ini kami sampaikan Susunan Panitia Seminar Hasil Program Strata Satu (S1) Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin dengan susunan sebagai berikut :

Pembimbing I/ Ketua : 1. Dr. Ir. Ingrid Nurtanio, M.T.
Penguji / Anggota : 2. Novy Nur RA Mokobombang, ST., Ms.TM., Ph.D.
3. Ir. Anugrayani Bustamin, ST., M.T.

Untuk dapat diterbitkan surat penugasannya

Demikian penyampaian kami, atas perhatian dan kerjasamanya diucapkan terima kasih.

Gowa, 26 Agustus 2024
Ketua Departemen Tek.Informatika,

Prof. Dr. Ir. Indrabayu, ST, MT., M.Bus.Sys., IPM, ASEAN.Eng
Nip.19750716 200212 1 004

Tembusan :
1. Arsip

Rabu, 28 Agustus 2024
Jam : 13.00
Ruang : [illegible]

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
Jalan Poros Malino Km. 6 Bontomarannu, Gowa, 92171, Sulawesi Selatan
☎ +62811 4420 909, E-mail: teknik@unhas.ac.id , https://eng.unhas.ac.id

SURAT PENUGASAN
No. 20860/UN4.7.1/TD.06/2024

Dari : Dekan Fakultas Teknik Universitas Hasanuddin

Kepada : Mereka yang tercantum namanya dibawah ini

Isi :

1. Bahwa merujuk kepada Peraturan Rektor Universitas Hasanuddin **Nomor : 29/UN4.1/2023 tentang Penyelenggaraan Program Sarjana Universitas Hasanuddin** , dengan ini menugaskan Saudara sebagai **PENGUJI/PANITIA SEMINAR HASIL** Program Strata Satu (S1) Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin dengan susunan sebagai berikut :

   Pembimbing I/ Ketua : 1. Dr. Ir. Ingrid Nurtanio, M.T.
   Penguji / Anggota : 2. Novy Nur RA Mokobombang, ST., Ms.TM., Ph.D.
   3. Ir. Anugrayani Bustamin, ST., M.T.

   Untuk menguji bagi mahasiswa tersebut dibawah ini :

   Nama/NIM : Maylinda Eka Christy Maylinda D121201050
   Program Studi : Teknik Informatika
   Judul thesis/Skrips : Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri RetailOtomotif Studi Kasus: Kalla Toyota)
2. Waktu seminar ditetapkan oleh Panitia Seminar Hasil Program Strata Satu (S1)
3. Agar Surat Penugasan ini dilaksanakan sebaik-baiknya dengan penuh rasa tanggung jawab.
4. Surat penugasa ini berlaku sejak tanggal ditetapkan sampai dengan berakhirnya seminar tersebut dengan ketentuan bahwa segala sesuatunya akan ditinjau dan diperbaiki sebagaimana mestinya apabila dikemudia hari terdapat kekeliruan dalam keputusan ini.

Ditetapkan di Gowa
Pada tanggal 26 Agustus 2024
a.n. Dekan,
Wakil Dekan Bidang Akademik dan Kemahasiswaan
Fakultas Teknik Unhas

Dr. Amil Ahmad Ilham, ST., M.IT
NIP. 197310101998021001

Tembusan :
1. Dekan Fak. Teknik Unhas
2. Ketua Departemen Teknik Informatika FT-UH
3. Mahasiswa yang bersangkutan

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN
RISET DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA
Kampus FakultasTeknikUnhas, Jl. PorosMalino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

DAFTAR HADIR UJIAN SKRIPSI MAHASISWA
FAKULTAS TEKNIK UNHAS

Nama/Stambuk : 1. Maylinda Eka Christy D121201050

Judul Skipsi/T.A : **"Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif (Studi Kasus: Kalla Toyota)"**

Hari/Tanggal : Rabu, 2 Oktober 2024
Jam : 09- 00 Wita – Selesai
Tempat : Ruang Lab. AIMP Departemen Teknik Informatika Gowa

No.	Jabatan	Nama Dosen	Tanda Tangan
I.	Pembimbing I	1. Dr.Ir. Ingrid Nurtanio,M.T	1. .....
II.	Anggota Penguji	2. Novy Nur RA Mokobombang,ST.,Ms.TM.,Pd.D	2 .....
		3. Ir. Anugrayani Bustamin,ST.,M.T	3 .....

PANITIA UJIAN

Ketua,

Dr. Ir. Ingrid Nurtanio,M.T

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN
RISET DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA

Kampus FakultasTeknikUnhas, Jl. PorosMalino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

BERITA ACARA UJIAN SKRIPSI

Pada hari ini Rabu, tanggal 2 Oktober 2024 Pukul **09.00 WITA - Selesai** bertempat di **Lab. AIMP Departemen Teknik Informatika Gowa** , telah dilaksanakan Ujian Skripsi bagi Saudara :

Nama : Maylinda Eka Christy
No. Stambuk : D121201050
Fakultas/Departemen : Teknik /Teknik Informatika
Judul Skripsi : **"Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif (Studi Kasus: Kalla Toyota)"**

Yang dihadiri oleh Tim Penguji Ujian Skripsi sebagai berikut :

No.	Nama	Jabatan	Tanda tangan
1.	Dr.Ir. Ingrid Nurtanio,M.T	Pemb I/Ketua	1.
2.	Novy Nur RA Mokobombang,ST.,Ms.TM.,Pd.D	Anggota	2.....
3.	Ir. Anugrayani Bustamin,ST.,M.T	Anggota	3.

Hasil keputusan Tim Penguji Ujian Skripsi/Tugas Akhir : **Lulus** / ~~**Tidak lulus**~~ dengan nilai angka .....88..... dan huruf (A).....

Gowa, 2 Oktober 2024

Ketua/Sekretaris Panitia Ujian,

Dr.Ir.Ingrid Nurtanio,M.T

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK INFORMATIKA
Kampus Fakultas Teknik Unhas, Jl. Poros Malino, Gowa
http://eng.unhas.ac.id/informatika, Email : informatika@unhas.ac.id

Gowa, 30 September 2024

Nomor : 1489/UN4.7.7.1/TD.06/2024
Lamp : -
Hal : Usulan Susunan Panitia/Penguji Ujian Sarjana

Yth. : Bapak Wakil Dekan Bidang Akademik dan Kemahasiswaan
Fakultas Teknik Unhas
Di
Gowa

Dalam rangka penyelesaian studi pada Departemen Teknik Informatika Fakultas Teknik Unhas, bersama ini kami usulkan susunan Panitia/Penguji Ujian Sarjana Program Strata Satu (S1) bagi mahasiswa Departemen Teknik Informatika Fakultas Teknik Uniersitas Hasanuddin atas nama :

Pembimbing / Ketua : 1. Dr. Ir. Ingrid Nurtanio, M.T
Penguji / Anggota : 2. Novy Nur RA Mokobombang, ST., Ms.TM., Ph.D
3. Ir. Anugrayani Bustamin, ST., M.T

Untuk Bertugas sebagai Penguji/ Penanggap Ujian Sarjana bagi Mahasiswa :

Nama : Maylinda Eka Christy
Stambuk : D121 20 1050

Dengan Judul Skripsi :

" Prediksi pelanggan churn menggunakan algoritma random forest dengan segmentasi k-means dalam industri retail otomotif (studi kasus: kalla toyota) "

Pada :
Hari/Tanggal : Rabu, 2 Oktober 2024
Jam : 09.00 Wita - Selesai
Tempat : Ruang Sidang Lab. AIMP

Demikian penyampaian kami, atas perhatiannya diucapkan terimah kasih.

Ketua Departemen Tek.Informatika,

Prof. Dr. Ir. Indrabayu.,ST, MT, M.Bus.Sys., IPM, ASEAN.Eng
Nip.197507016 200212 1 004

Tembusan :
1. Arsip

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
UNIVERSITAS HASANUDDIN
FAKULTAS TEKNIK
Jalan Poros Malino Km. 6 Bontomarannu, Gowa, 92171, Sulawesi Selatan
☎ +62811 4420 909, E-mail: teknik@unhas.ac.id , https://eng.unhas.ac.id

SURAT PENUGASAN

No. 24314/*UN4.7.1/TD.06/2024*

Dari : Dekan Fakultas Teknik Universitas Hasanuddin.

Kepada. : Mereka yang tercantum namanya di bawah ini.

Isi : 1. Bahwa merujuk kepada Peraturan Rektor Universitas Hasanuddin **Nomor : 29/UN4.1/2023 tentang Penyelenggaraan Program Sarjana Universitas Hasanuddin**, dengan ini menugaskan Saudara sebagai **PENGUJI/PANITIA UJIAN SARJANA** Program Strata Satu (S1) Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin dengan susunan sebagai berikut :

Pembimbing / Ketua	: 1. Dr. Ir. Ingrid Nurtanio, M.T
Penguji / Anggota	: 2. Novy Nur RA Mokobombang, ST., Ms.TM., Ph.D
	3. Ir. Anugrayani Bustamin, ST., M.T

untuk menguji bagi mahasiswa tersebut di bawah ini :

Nama/NIM	: Maylinda Eka Christy	D121201050
Program Studi	: Teknik Informatika	
Judul Thesis/Skripsi	: Prediksi Pelanggan Churn Menggunakan Algoritma Random Forest dengan Segmentasi K-Means dalam Industri Retail Otomotif (Studi Kasus: Kalla Toyota)	

2. Waktu Ujian ditetapkan oleh Panitia Ujian Sarjana Program Strata Satu (S1).
3. Agar Surat penugasan ini dilaksanakan sebaik-baiknya dengan penuh rasa tanggung jawab.
4. Surat penugasan ini berlaku sejak tanggal ditetapkan sampai dengan berakhirnya Ujian Sarjana tersebut, dengan ketentuan bahwa segala sesuatunya akan ditinjau dan diperbaiki sebagaimana mestinya apabila dikemudian hari ternyata terdapat kekeliruan dalam keputusan ini.

Ditetapkan di Gowa,
Pada tanggal 30 September 2024
a.n. Dekan
Wakil Dekan Bidang Akademik dan Kemahasiswaan
Fakultas Teknik Unhas

Dr. Amil Ahmad Ilham, ST., M.IT
NIP.197310101998021001

Tembusan :
1. Dekan Fak. Teknik Unhas
2. Ketua Departemen Teknik Informatika FT-UH
3. Kasubag. Umum dan Perlengkapan FT-UH

DAFTAR PERBAIKAN

Maylinda Eka Christy – D121201050

PREDIKSI PELANGGAN CHURN MENGGUNAKAN ALGORITMA RANDOM FOREST DENGAN SEGMENTASI K-MEANS DALAM INDUSTRI RETAIL OTOMOTIF

(Studi Kasus: KALLA TOYOTA)

No	Penjelasan Revisi	BAB
1.	Tinjau kembali abstrak untuk menyertakan penjelasan singgat mengenai *churn* dan CLTV.	Abstrak
2.	Lihat dan perhatikan kembali tiap-tiap penulisan bahasa inggris yang harus dituliskan dengan huruf miring.	Bab I, II, II dan IV
3.	Penulisan nomor urut pada setiap rumus perlu dicantumkan	Bab I
4.	Tinjau kembali tiap tiap singkatan untuk di cantumkan kepanjangannya pada awal pembahasan awal, sehingga untuk kemunculan singkatan yang sama tidak perlu ditulisakn ulang kepanjangannya	Bab I, II, I dan IV
5.	Perbagiki kualitas gambar agar lebih terang dan HD serta perbesar gambar	Bab I, II, III dan IV
6.	Berikan penjelasan mengenaik Chi-Square	Bab I
7.	Buat flow skema dari tahapan skripsi agar mempermudah dalam melihat kompleksitas skripsi	Bab II

LEMBAR PERBAIKAN SKRIPSI

PREDIKSI PELANGGAN CHURN MENGGUNAKAN ALGORITMA RANDOM FOREST DENGAN SEGMENTASI K-MEANS DALAM INDUSTRI RETAIL OTOMOTIF

(Studi Kasus: KALLA TOYOTA)

OLEH:

Maylinda Eka Christy

D121 20 1050

Skripsi ini telah dipertahankan pada Ujian Akhir Sarjana tanggal 02 Oktober 2024.

Telah dilakukan perbaikan penulisan dan isi skripsi berdasarkan usulan dari penguji dan pembimbing skripsi.

Persetujuan perbaikan oleh tim penguji:

	Nama	Tanda Tangan
Ketua	Dr. Ir. Ingrid Nurtanio, M.T.	
Anggkota	Novy Nur R.A Mokobombang,ST.,Ms.TM. Ph. D	
	Ir. Anugrayani Bustami, S.T., M.T	

Persetujuan Perbaikan oleh pembimbing:

Pembimbing	Nama	Tanda Tangan
I	Dr. Ir. Ingrid Nurtanio, M.T.