BAB II
TINJAUAN PUSTAKA

Tinjauan pustaka dari penelitian terdiri dari pencapaian terdahulu, dan tinjauan teoritis yang diperoleh peneliti. Oleh karena itu, penjabaran dari tinjauan pustaka adalah berikut ini.

2.1 Pencapaian Terdahulu

Penelitian pada pengujian akurasi K-Nearest Neighbors dan Decision Tree dalam pemodelan harga mobil bekas mengacu pada penelitian terdahulu. Pencapaian terdahulu dijadikan sebagai acuan pembanding dan studi literatur dalam penelitian ini. Referensi penelitian terdahulu dengan algortima yang sama dijadikan acuan. Pertama, jurnal " *The prediction of scholarship recipients in higher education using K-Nearest Neighbor algorithm*" oleh Kurniadi, Abdurachman, Warnars, dan Suparta. Pada penelitian tersebut dilakukan prediksi untuk menentukan calon penerima beasiswa menggunakan algoritma K-Nearest Neighbors dan terbukti performa algoritma tersebut baik dengan tingkat akurasi yang tinggi mencapai 95,83% berdasarkan masukan variasi nilai (Kurniadi et al., 2018, p. 6).

Kedua, jurnal "*An optimized K-Nearest Neigbors based breast cancer detection*" oleh Assegie Tsehay. Penelitian tersebut dilakukan untuk melakukan optimalisasi prediksi kanker payudara dengan mencari K terbaik yang digunakan dalam algoritma K-Nearest Neighbors. Nilai K terbaik menghasilkan akurasi mencapai 94.35 % (Assegie, 2021, p. 117). Ketiga, jurnal "Penerapan Metode K-Nearest Neighbors Untuk Sistem Rekomendasi Pemilihan Mobil" oleh Ni Luh Gede. Pada penelitian tersebut dilakukan penerapan K-Nearest Neighbors untuk sistem rekomendasi pemilihan mobil. Tiga penelitian tersebut memiliki algoritma yang sama dengan studi kasus yang berbeda.

Kemudian jurnal internasional "*ICA Learning Approach for Predicting RNA-Seq Data Using KNN and Decision Tree Classifiers*" oleh Marion O.A., Ayodele A.A., Olatunji O., dan Micheal O. Penelitian ini dilakukan untuk melakukan prediksi dan deteksi penyakit malaria melalui RNA-Seq menggunakan KNN dan Decision Tree dengan menghasilkan akurasi mencapai 81,77% dan 73,3% (Adebiyi

et al., 2020). Kelima, artikel internasional "*Analysis of Decision Tree and K-Nearest Neighbor Algorithm in the Classification of Breast Cancer*" oleh Harikumar Rajaguru, dan Sannasi C. Penelitian ini dilakukan untuk melakukan analisis klasifikasi dari kanker payudara dengan algoritma K-NN dan Decision Tree dengan hasil yang diperoleh dalam penelitian adalah perbandingan performa dari kedua algoritma dalam bentuk diagram batang. Pada diagram batang tersebut dapat disimpulkan peforma K-NN lebih baik dari Decision Tree. Selanjutnya jurnal dari politeknik Caltex Riau, "Model Prediksi Kemenangan Tim dalam Game League Of Legend Menggunakan Algoritma Decision Tree" oleh Green Arther Sandag. Penelitian ini mendapatkan akurasi dari Decision Tree mencapai 96,42%.

Ketujuh, jurnal ilmiah Indonesia dengan judul "Prediksi Harga Mobil Bekas dengan Machine Learning" oleh Bambang Kriswantara, Kurniawati, dan Hilman F. Pardede. Penelitian tersebut menggunakan algoritma *Deep Neural Network* (DNN), Decision Tree, dan Random Forest. Akurasi dari Decision Tree pada penelitian tersebut mencapai 72%, Random Forest mencapai 84%, kemudian DNN menghasilkan R^2 sebesar 0.88. Pada penelitian ini fitur yang digunakan adalah merek, model, transmisi, tipe *body*, dan tipe bahan bakar. Terakhir, jurnal sistem informasi dan teknologi informasi dengan judul "Estimasi Harga Jual Mobil Bekas Menggunakan Metode Regresi Linier Berganda" oleh Evi Dewi Sri Mulyani, Firham Mulady, Dendi Ramadhan, Ari Ariyantono, Dikri Ramdani, Robi Wahyundana, dan M. Gilang.

2.2 Tinjauan Teoritis

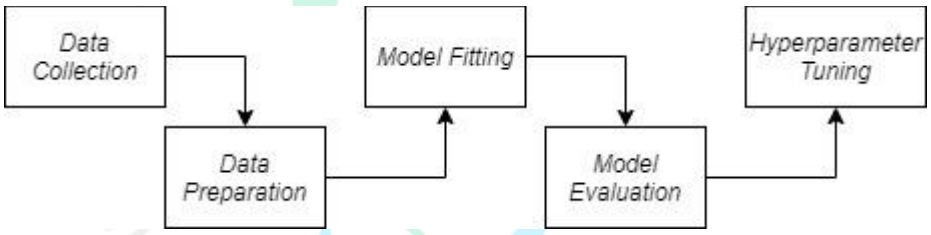
Referensi teoritis merupakan bagian yang akan membahas hasil dari mempelajari teori dari topik yang akan diangkat dengan bantuan dari beberapa dokumen ilmiah, seperti jurnal ilmiah, dan artikel ilmiah.

2.2.1 *Machine Learning*

Machine Learning salah satu teknologi yaitu mesin yang dikembangkan untuk dapat melakukan pembelajaran dengan sendirinya tanpa arahan dari pengguna (Takdirillah, 2020). Ada beberapa bentuk pembelajaran dari *machine learning*; *supervised*, *unsupervised*, *semi-supervised*, dan *reinforcement learning*

(Edward, 2018). Metode yang dilakukan oleh machine learning umumnya adalah klasifikasi dan prediksi. Berdasarkan ciri-ciri atau syarat tertentu dapat melakukan klasifikasi suatu objek. Kemudian prediksi untuk menemenukan perkiraan *output* dari sebuah data yang dimasukan. *Machine learning* membutuhkan data yang akan dipelajari atau dapat disebut data *training* untuk melakukan proses pelatihan atau pembelajaran. Istilah-istilah yang ada dalam machine learning adalah sebagai berikut ini.

(1) Dataset atau sekumpulan data.
(2) Fitur adalah bagian dari data atau dapat disebut atribut untuk membantu memahami masalah yang akan dimasukkan ke dalam *machine learning*.
(3) Model adalah representasi yang dibentuk untuk machine learning melakukan pembelajaran data, dan akan menjadi keluaran yang akan didapatkan setelah melatih data.



Gambar 2.1 *Process of Machine Learning*

Proses sebuah *machine learning* melakukan pembelajaran atau pemodelan ditunjukan dalam gambar 2.1. Pada tahap pertama, *data collection* atau pengumpulan data yang akan dipelajari oleh algoritma. Setelah pengumpulan, dilakukan *data preparation* atau *pre-processing* data dengan melakukan rekayasa data ke dalam format yang optimal, serta menentukan fitur yang penting. Kemudian *model fitting*, algoritma belajar dengan data yang telah dikumpulkan dan disiapkan. Setelah algoritma melakukan pelatihan, dilakukan *model evaluation* atau menguji sebuah model untuk melihat kinerja model tersebut. Terakhir, *tuning* atau penyetelan dengan menyempurnakan model untuk memaksimalkan kinerja.

2.2.2 *Data Collection*

Data yang dikumpulkan dalam penelitian ini merupakan data sekunder. Data diperoleh dari *platform* Kaggle. Dengan data yang diperoleh, algoritma akan mempelajari data tersebut untuk hasil pemodelan harga mobil bekas. Kaggle merupakan komunitas online yang menampung para peminat data *science* yang ingin belajar lebih mengenai machine learning. Kaggle telah memiliki lebih dari 1000 dataset untuk berbagai kegiatan yang dilakukan *data scientist* (Rahmalia, 2021).

Kegiatan yang dilakukan Kaggle ada kompetisi *machine learning*, dan forum berbagi penulisan kode untuk membantu sesama komunitas. Data yang dikumpulkan dari Kaggle merupakan kumpulan dari platform OLX tahun 2019. Data dapat diperoleh di https://www.kaggle.com/fadhrigabestari/olxmobilbekas .

2.2.3 *Pre-processing* Data

Data *pre-processing* merupakan proses persiapan data mentah yang telah dikumpulkan. Persiapan data perlu diolah terlebih dahulu dengan dua tahap yaitu data *cleansing* dan normalisasi dataset. Data *cleansing* dilakukan untuk mengeluarkan data yang tidak lengkap yang terdapat pada dataset. Data yang tidak lengkap tidak akan digunakan pada proses selanjutnya. Data akan dibagi menjadi dua tipe, yaitu data *train* dan *test* data (Salam et al., 2020, p. 532).

Kemudian dilakukan normalisasi pada dataset menggunakan teknik *min-max scaler* dari sklearn (Wiranda & Sadikin, 2019, p. 188). *Min-max scaler* merupakan suatu metode untuk melakukan transformasi linear dengan menggunakan nilai minimum dan maksimum untuk menghasilkan keseimbangan data satu dengan lainnya pada rentang 0 dan 1 (Suryanegara et al., 2021, p. 117). Skala nilai dari data menjadi lebih kecil tetapi bobot nilai sama. Dengan skala nilai atribut data yang baru dapat membantu menaikan kinerja model. Hal tersebut terjadi karena dapat menghapus fitur dengan noise yang tinggi dan relevansi yang rendah. Rumus untuk melakukan normalisasi data adalah sebagai berikut ini.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Dimana x' adalah nilai hasil normalisasi, x adalah nilai data aktual yang akan dinormalisasi, x_{max} adalah nilai maksimum dari data aktual, dan x_{min} adalah nilai minimum dari data aktual.

2.2.4 K-Nearest Neighbors

K-Nearest Neighbors merupakan bentuk *supervised machine learning* yang dapat menyelesaikan masalah klasifikasi dan regresi. Algoritma yang mempelajari suatu hal dengan menerapkan pola yang sama dari data yang sudah ada. Oleh karena itu, dibutuhkan data untuk menjadi data pelatihan. Algoritma K-NN dapat menyelesaikan suatu objek atau masalah berdasarkan data terdekat (Kurniadi et al., 2018).

K-Nearest Neighbors menemukan sekelompok sampel yang paling dekat dengan sampel data yang ingin diuji berdasarkan fungsi jarak. K memiliki peran yang penting dalam kinerja algoritma K-NN (Thanh Noi & Kappas, 2017, p. 7). Dengan diberikan titik uji, maka akan ditemukan sejumlah cluster atau K tetangga dalam data pelatihan yang paling dekat dengan titik uji. Fungsi jarak di dalam K-Nearest Neighbors menggunakan rumus jarak Euclidean. Teknik di dalam K-NN sederhana dan mudah serta mirip seperti clustering, pengelompokan data berdasarkan jaraknya ke beberapa data terdekat (Lubis et al., 2020, p. 327). Proses atau alur dari algoritma K- Nearest Neighbors adalah sebagai berikut ini (Harrison, 2018).

1. Pemuatan data atau *load* data.
2. Menentukan nilai K.
3. Menghitung jarak antara data training dengan data yang mau dilakukan pengujian. Dalam menentukan jarak di dalam K-Nearest Neighbors menggunakan rumus Euclidean. Dapat juga disebut "Jarak Euclidean".

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

Dimana d adalah jarak, n adalah nomor atribut, x_i adalah data *test*, dan y_i adalah data *training*.

4. Menentukan banyaknya nilai K yang merupakan nilai banyaknya tetangga atau data yang terdekat dengan data yang diuji. RMSE yang kecil menghasilkan nilai K yang optimal.
5. Urutkan data jarak sebanyak nilai K dari yang paling terkecil dan paling besar atau jauh.
6. Hitung rata-rata dari data yang telah diurutkan untuk kasus prediksi, dan hitung nilai yang sering muncul atau *mode* untuk klasifikasi.

Optimalisasi algoritma K-Nearest Neighbors dapat dilakukan dengan menggunakan K-optimal. Pada penelitian ini, peneliti mencari K-optimal menggunakan metode Elbow. Metode Elbow merupakan metode untuk menghasilkan informasi jumlah cluster terbaik berdasarkan nilai *Mean Square Error* (MSE) atau rata-rata kuadrat kesalahan. Formula dari MSE adalah sebagai berikut ini.

$$MSE = \frac{\sum_{i=1}^{n}(Y_i - \hat{y}_i)^2}{n} \quad (3)$$

Dimana Y adalah original data *test* , $\hat{Y}$ adalah prediksi data *test*, dan n adalah jumlah dari data *test*.

2.2.5 Evaluasi Model

Di dalam penelitian ini, terdapat dua parameter yang dijadikan alat ukur kinerja dari model. Pertama, *Root Mean Square Error* (RMSE) merupakan alat ukur kinerja akurasi dari sebuah model. Dalam melakukan perkiraan atau prediksi dibutuhkan evaluasi keakuratan hasil perkiraan tersebut. Pada umumnya, metode evaluasi model regresi atau prediksi data menggunakan RMSE. Nilai RMSE rendah menunjukan bahwa variasi nilai yang dihasilkan oleh sebuah model dekat dengan variasi nilai yang ada pada data pengamatan atau pembelajaran sehingga merupakan hasil yang baik (Salam et al., 2020, p. 534) .

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Y_i - \hat{y}_i)^2}{n}} \quad (4)$$

Dimana Y adalah original data *test* , $\hat{Y}$ adalah prediksi data *test*, dan n adalah jumlah dari data *test*.

Perhitungan RMSE dilakukan untuk mengukur kesalahan model dalam memperkirakan data kuantitatif. Dalam melakukan evaluasi RMSE cukup kecil atau tidak dalam model yang kita bangun itu dilakukan berdasarkan pada seberapa akurat kita membutuhkan model kita untuk diberikan. Tidak ada rumus matematika untuk menghitung ambang batas yang baik karena semua tergantung dengan keputusan yang berada dalam lingkup pengetahuan dan kebijaksanaan manusia (Moody, 2019) .

Kedua, parameter yang digunakan adalah koefisien determinasi atau R *squared*. Metode untuk melakukan evaluasi model yang mempunyai fokus untuk menunjukan pengaruh dari dua atau lebih banyak fitur atau variabel bebas terhadap fitur yang diuji (Andhika, 2020). Hasil perhitungan semakin baik apabila mendekati angka satu. Rumus dari R^2 adalah sebagai berikut ini.

$$R^2 = 1 - \frac{RSS}{\sum(y_i - \bar{y})^2} \quad (5)$$

Dimana RSS adalah perhitungan Residual Sum of Squares, y_i adalah data aktual atau data yang sebenarnya, dan $\bar{y}$ adalah rata-rata dari *sample*.

Saeedi, M. (2020) mengungkapkan bahwa berdasarkan rule of thumb, dapat dikatakan bahwa nilai RMSE di bawah 0,5 menunjukan bahwa model relatif dapat memprediksi data secara akurat. Sementara itu, Alhyari, S. (2016) mengungkapkan bahwa jika nilai R-kuadrat > 0,7 adalah nilai yang sangat baik untuk menunjukan akurasi. Referensi yang digunakan adalah Moore,D.S., Notz, W.I, & Flinger, M.A. (2013). Praktek dasar statistic (edisi ke-6). New York, NY: W.H. Freeman and Company. Halaman(138). Oleh karena itu, peneliti menetapkan ambang batas yang digunakan oleh RMSE dapat dikatakan akurat apabila kurang dari 0,05 dan R-kuadrat dapat dikatakan akurat apabila melebihi 0,7.

2.2.6 Decision Tree

Pohon keputusan atau *Decision Tree* adalah salah satu metode atau teknik dalam klasifikasi data mining yang digunakan untuk membangun sebuah model berdasarkan atribut dari kumpulan data (Hamoud et al., 2018, p. 27). Pemodelan yang dihasilkan merupakan prediktif teknik untuk melakukan predikat, klasifikasi, atau kategori data yang diberikan objek berdasarkan model yang sudah ada sebelumnya yang digunakan sebagai pelatihan kumpulan data dengan fitur atau atribut yang sama.

Struktur pohon yang dihasilkan berupa simpul akar, node internal, dan daun (terminal). Simpul akar adalah simpul paling atas yang tidak mempunyai simpul masuk tetapi mempunyai satu ada lebih simpul keluar atau disebut *edges*. Simpul tengah atau node internal memiliki satu sisi masuk dan satu atau lebih tepi keluar, setiap simpul tengah menunjukan pengujian pada atribut dan setiap sisinya mewakili hasil tes. Daun atau simpul terakhir merepresentasikan sugesti terakhir untuk atribut prediksi dari objek data.

Dalam decision tree kategori klasifikasi atau *Classification Tree* terdapat rumus *entropy* untuk mengukur derajat ketidakteraturan dari suatu sistem atau data (Hakim, 2019). Rumus *entropy* adalah sebagai berikut ini.

$$S \equiv \sum_{i=1}^{c} -\rho_i \, log\, 2\, \rho_i \quad (6)$$

Dimana S adalah *entropy*, c adalah jumlah kelas atau atribut dan ρ_i adalah jumlah data yang menjadi milik kelas tersebut. Setelah menghitung *entropy* terdapat rumus untuk perolehan informasi atau dapat diartikan dengan penurunan yang diharapkan dalam *entropy* untuk membangun pohon keputusan. Dengan penentuan atribut menggunakan nilai *information gain* atau perolehan informasi yang terbesar. Rumus *information gain* adalah sebagai berikut ini.

$$Gain\,(S, A) \equiv Entropy\,(S) - \sum_{v \in Value\ A} \frac{|s_v|}{|s|}\ Entropy\,(s_v) \quad (7)$$

Dimana S adalah nilai entropy yang sudah dihitung, A adalah atribut yang diinginkan, |S| adalah jumlah data, dan $|S_v|$ adalah jumlah data A.

Sementara itu, penelitian ini menggunakan decision tree kategori regresi atau dapat juga disebut *Regression Tree* dalam menentukan prediksi. Dalam menentukan *root node* dan proses pembagian pohon terdapat rumus *Residual Sum of Squares* (RSS).

$$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (8)$$

Dimana $\hat{y}_i$ adalah prediksi dalam leaf node atau rata-rata suatu bagian, dan y_i adalah data yang sebenarnya. *Root node* terbaik untuk membagi data lebih lanjut ditunjukan pada nilai RSS yang rendah (Nancy, 2019). Pada bagian penerapan kode program menggunakan *Mean Squared Error* (MSE) untuk menghitung rata-rata RSS.

2.2.7 Bahasa Pemrograman Python

Python salah satu bahasa pemrogram yang banyak digunakan oleh perusahaan besar karena mudah dipahami (Muhardian, 2018). Bahasa pemrograman python merupakan interpreter sistem atau menerjemahkan bahasa baris perbaris yang dibuat oleh Guido van Rossum. Jika ada kesalahan setelah menjalankan maka program akan berhenti ditengah. Python adalah alat gratis yang dapat digunakan oleh siapapun tanpa batasan.

Python mempunyai banyak perpustakaan yang terdiri dari paket yang dibuat oleh perusahaan untuk memudahkan pengguna dalam melakukan pengetikan kode program. Sintaks yang terdapat pada Python mudah dipahami dan memiliki gaya penulisan yang mirip dengan bahasa Inggris. Kemudian sintaks kode di dalam Python sangat jelas dan juga dilengkapi dengan fungsionalistas perpustakaan standar yang besar dan komprehensif (Pulungan et al., 2021, p. 105).

Library di Python merupakan kumpulan dari *package* dan modul yang mempunyai fungsi dan tujuan untuk memudahkan pembuatan kode program. Library yang digunakan oleh peneliti adalah sebagai berikut ini.

a. Pandas

   Dapat memuat dan membaca sebuah *file* dalam bentuk seperti csv, excel, txt, dan lainnya. Pandas menggunakan sistem *dataframe* untuk memuat sebuah *file* (Arslan, 2017).

b. Numpy

   Dapat digunakan untuk operasi matriks dan vektor. Numpy juga dapat mengelola *array*. Salah satu library yang digunakan oleh library lain untuk kebutuhan analisis.

c. Matplotlib

   Digunakan untuk membuat visualisasi data dengan baik. Matplotlib dapat membuat grafik dengan warna yang menarik dan interaktif. Oleh karena itu, peneliti dapat membuat visualisasi data berupa grafik.

d. Scikit-Learn

   Scikit-learn memberikan *package* dan modul untuk keperluan pengujian algoritma.