2. LANDASAN TEORI

2.1 Tinjauan Pustaka

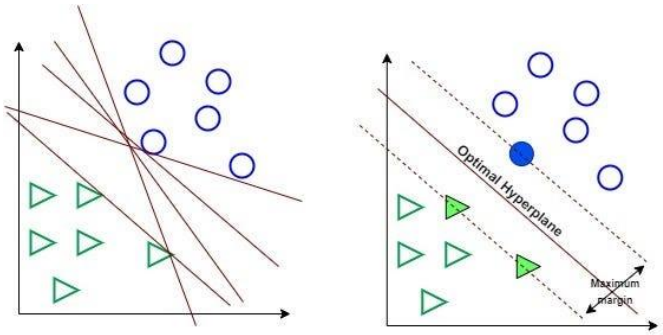
Tinjauan pustaka akan menjelaskan landasan teori yang digunakan dalam proses pembuatan skripsi ini.

2.1.1 Penyakit Stroke

Stroke adalah kondisi yang terjadi ketika pasokan darah ke otak mengalami gangguan atau berkurang akibat penyumbatan (stroke iskemik) atau pecahnya pembuluh darah (stroke hemoragik). Tanpa pasokan darah, otak tidak akan mendapatkan asupan oksigen dan nutrisi, sehingga sel-sel pada sebagian area otak akan mati. Kondisi ini menyebabkan bagian tubuh yang dikendalikan oleh area otak yang rusak tidak dapat berfungsi dengan baik. Stroke merupakan kondisi gawat darurat yang perlu ditangani secepatnya, karena sel otak dapat mati hanya dalam hitungan menit (Halodoc, 2020). Tindakan penanganan yang cepat dan tepat dapat meminimalkan tingkat kerusakan otak dan mencegah kemungkinan munculnya komplikasi.

2.1.2 *Support Vector Machine*

Support Vector Machine (SVM) merupakan salah satu metode dalam *supervised learning* yang biasanya digunakan untuk klasifikasi. *SVM* pertama kali diperkenalkan oleh seorang professor dari Columbia, Amerika Serikat yang bernama Vladimir N Vapnik pada tahun 1993. Tujuan dari algoritma *SVM* adalah untuk menemukan *hyperplane* terbaik dalam ruang berdimensi-N (ruang dengan N-jumlah fitur) yang berfungsi sebagai pemisah yang jelas bagi titik-titik data input (Setia, 2021).

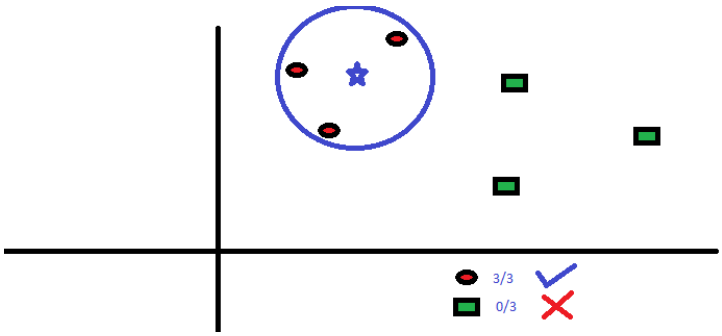


Gambar 2.1 Ilustrasi Cara Kerja *Hyperplane SVM*

Pada Gambar 2.1 diatas, bagian sebelah kiri menunjukkan beberapa kemungkinan bidang (*hyperplane*) untuk memisahkan data lingkaran dan data segitiga. Algoritma *SVM* kemudian mencari *hyperplane* terbaik yang dapat memisahkan kedua kelas secara optimal. Seperti tampak pada Gambar 2.1 di sebelah kanan, sebuah *hyperplane* optimal berhasil dibuat dan mampu memisahkan kedua kelas sehingga memiliki margin yang paling maksimal. Garis *maxium* margin yang menyentuh suatu kelas disebut *support vector*.

2.1.3 *K-Nearest Neighbors*

Algoritma *K-Nearest Neighbors* atau *KNN* adalah salah satu algoritma dalam *supervised learning* dimana hasil dari *instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori k terdekat (Team, 2021). Tujuan dari algoritma ini adalah untuk mengklasifikasikan objek baru berdasarkan atribut dan sampel-sampel dari *training* data. Algoritma *K-Nearest Neighbors* ini menggunakan metode *neighborhood classification* sebagai nilai prediksi dari nilai *instance* yang baru. *KNN* dapat digunakan di metode klasifikasi dan juga regresi untuk mengatasi beberapa permasalahan. Namun, sebagian besar lebih banyak menggunakan *KNN* dalam masalah klasifikasi di industri.



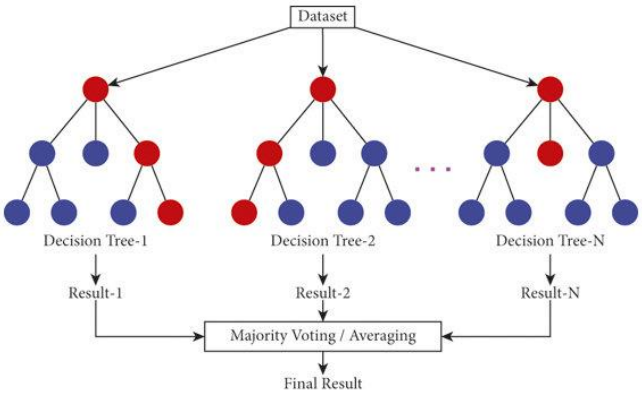
Gambar 2.2 Ilustrasi Cara Kerja *K-Nearest Neighbors*

Seperti penjelasan di atas, algoritma *KNN* digunakan untuk mengklasifikasikan data. Maka dari itu, apabila ingin mengetahui kelas dari sebuah objek terlebih dahulu harus menentukan nilai k. Contohnya terdapat pada Gambar 2.2 di atas, *KNN* akan digunakan untuk mengetahui kelas dari objek berbentuk bintang biru. Objek tersebut bisa termasuk ke dalam kelas objek berbentuk lingkaran merah ataupun kelas objek berbentuk kotak hijau. Nilai k bisa ditetapkan secara sembarang, dalam kasus ini, nilai k ditetapkan 3. Dikarenakan nilai k tiga, maka akan dibuat sebuah lingkaran dengan objek bintang biru sebagai pusatnya yang hanya mencakup tiga data saja (dikarenakan nilai k nya 3). Dapat dilihat bahwa tiga titik terdekat pada objek bintang biru

semuanya adalah lingkaran berwarna merah. Dikarenakan hal tersebut maka dapat dikatakan bahwa objek bintang biru termasuk ke dalam kelas lingkaran merah.

2.1.4 *Random Forest*

Random Forest adalah algoritma *machine learning* yang menggabungkan keluaran dari beberapa *decision tree* untuk mencapai satu hasil. Algoritma *Random Forest* diperkenalkan oleh Leo Breiman dan Adele Cutler. Algoritma ini didasarkan pada konsep *ensemble learning*, yakni proses menggabungkan beberapa pengklasifikasi untuk memecahkan masalah yang kompleks dan untuk meningkatkan kinerja model. Sesuai namanya, *Forest* atau 'hutan' dibentuk dari banyak *tree* (pohon) yang diperoleh melalui proses *bagging* atau *bootstrap aggregating* (Trivusi, 2022). Setiap *tree* pada *Random Forest* akan mengeluarkan prediksi kelas. Prediksi kelas dengan vote terbanyak menjadi kandidat prediksi pada model. Semakin banyak jumlah *tree* maka akan menghasilkan akurasi yang lebih tinggi dan mencegah masalah *overfitting*.



Gambar 2.3 Ilustrasi Cara Kerja *Random Forest*

Cara kerja algoritma *Random Forest* dapat dijabarkan dalam langkah-langkah berikut:

- Algoritma memilih sampel acak dari *dataset* yang disediakan.
- Membuat *decision tree* untuk setiap sampel yang dipilih. Kemudian akan didapatkan hasil prediksi dari setiap *decision tree* yang telah dibuat.
- Dilakukan proses *voting* untuk setiap hasil prediksi. Untuk masalah klasifikasi menggunakan modus (nilai yg paling sering muncul).
- Algoritma akan memilih hasil prediksi yang paling banyak dipilih (*vote* terbanyak) sebagai prediksi akhir.

2.1.5 *Exploratory Data Analysis*

Exploratory Data Analysis (EDA) adalah bagian dari proses *data science*. *EDA* menjadi sangat penting sebelum melakukan *feature engineering* dan *modeling* karena dalam tahap ini peneliti harus memahami datanya terlebih dahulu (Chandra, 2019). *Exploratory Data Analysis* memungkinkan peneliti memahami isi data yang digunakan, mulai dari distribusi, frekuensi, korelasi dan lainnya.

2.1.6 *Data Visualization*

Data Visualization atau visualisasi data adalah rangkaian proses menampilkan data atau informasi dalam bentuk yang mudah dipahami, seperti grafik, angka, bagan, dan lain sebagainya (Henny, 2022). Singkatnya, *visualization* menyederhanakan kumpulan data untuk ditampilkan. Dalam penerapannya, menggunakan elemen visual bertujuan untuk mempermudah pembaca dalam melihat dan memahami tren, *outliers*, dan pola dalam suatu data. *Visualization* memungkinkan para *decision maker* melihat analitik yang disajikan secara visual. Sehingga, mereka dapat memahami konsep yang sulit atau mengidentifikasi pola baru. Dengan begitu, pembuatan strategi dan pengambilan keputusan pun menjadi lebih tepat.

2.1.7 *Grid Search Cross Validation*

Grid Search Cross Validation adalah metode pemilihan kombinasi model dan *hyperparameter* dengan cara menguji coba satu persatu kombinasi dan melakukan validasi untuk setiap kombinasi. Tujuannya adalah menentukan kombinasi yang menghasilkan performa model terbaik yang dapat dipilih untuk dijadikan model untuk prediksi (Hidayat, 2021). *Grid Search Cross Validation* membantu menghindari *overfitting* pada *dataset* tertentu dan memberikan pemahaman yang lebih baik tentang seberapa baik model akan berkinerja pada data yang belum pernah dilihat sebelumnya. Teknik ini sangat bermanfaat ketika bekerja dengan model *machine learning* yang memiliki banyak *hyperparameter* dan perlu menentukan kombinasi yang optimal untuk meningkatkan kinerja model.

2.1.8 *Dataset*

Dataset merupakan kumpulan data yang terurut dan diperoleh dari kumpulan informasi (Raharja, 2022). Kumpulan informasi sendiri diperoleh dari pengamatan, pengukuran, studi, atau analisis hingga menjadi data. Data bisa berupa fakta, angka, nama, atau bahkan deskripsi.

Oleh karena itu, *dataset* berkaitan erat dengan kegiatan *data mining* yang membantu para *data scientist* untuk menganalisis data menjadi suatu informasi koheren.

2.2 Tinjauan Studi

2.2.1 *Stroke Prediction using Machine Learning Method with Extreme Gradient Boosting Algorithm* (Rahim et al., 2022)

- Masalah yang diangkat di penilitian ini adalah penyakit stroke mewakili 32% dari semua penyebab kematian secara global. Dampak dari *COVID-19* juga berpengaruh dalam perawatan stroke karena terjadi infeksi lanjutan dari infeksi *COVID-19* yang dapat meningkatkan risiko terkena stroke dan terbukti 1.4% pasien yang terinfeksi *COVID-19* terkena penyakit stroke. Dilihat dari banyaknya penggunaan teknologi yang dapat membantu dalam dunia kesehatan, menggunakan model *machine learning* dianggap bisa dipakai untuk memudahkan dalam memprediksi penyakit dengan menunjukkan akurasi. Tujuan dari penelitian ini adalah untuk membantu tenaga medis untuk mendiagnosa pasien yang berpotensi mengalami penyakit stroke.
- Metode yang diusulkan dari penelitian ini adalah menggunakan *Extreme Gradient Boost*.
- Hasil dari penelitian ini adalah menunjukkan bahwa penggunaan metode *XGBoost* dalam memprediksi penyakit stroke mendapatkan akurasi yang cukup tinggi yaitu 96%.
- Perbedaan penelitian yang dilakukan dengan skripsi ini adalah penggunaan metode pada penelitian ini hanya menggunakan satu metode saja yaitu *XGBoost* sedangkan pada skripsi ini menggunakan tiga metode yaitu *Support Vector Machine, K-Nearest Neighbors* dan *Random Forest.* Lalu pada *dataset* penelitian ini memiliki data yang tidak seimbang jumlahnya, terletak pada jumlah data pada variabel target. Sedangkan pada skripsi ini menggunakan dataset yang seimbang jumlahnya.

2.2.2 *Penerapan Metode Adaboost Untuk Mengoptimasi Prediksi Penyakit Stroke Dengan Algoritma Naïve Bayes* (Byna & Basit, 2020)

- Masalah yang diangkat di penelitian ini adalah penyakit stroke merupakan penyakit yang paling mematikan di dunia menurut *WHO*. Penderitanya mengalami cedera pada sistem saraf. Karena hal ini pakar kesehatan khususnya dibidang keperawatan membutuhkan bantuan perhatian khusus. Saat ini perkembangan Era Revolusi Industri 4.0 yang berkolaborasi di bidang teknologi dan ilmu kesehatan sehingga menjadi

sesuatu yang bermanfaat dengan menggunakan *machine learning.* Banyak sekali manfaat yang digunakan dalam memprediksi beberapa penyakit yang dapat diantisipasi. Tujuan dari penelitian ini adalah untuk membantu tenaga medis untuk mendiagnosa pasien yang berpotensi mengalami penyakit stroke.

- Metode yang diusulkan dalam penelitian yang dilakukan adalah menggunakan *Naïve Bayes* dan *Naïve Bayes-Adaboost.*
- Hasil penelitian untuk menunjukkan nilai akurasi algoritma Naïve Bayes memiliki nilai 97.6% dengan *split data* 80/20, sedangkan untuk nilai akurasi optimasi *Adaboost* dengan *Naïve Bayes* senilai 98.1% *split data* 70/30 kedua model tersebut memiliki diagnosa *Excellent Classification*, dalam pengujian prediksi penyakit stroke dengan dengan 11 variabel dengan jumlah data 28,500 untuk *data training* dan 572 untuk *data testing*.
- Algoritma *Adaboost* terbukti dapat mengoptimasi hasil akurasi prediksi jika digabungkan dengan metode *Naïve Bayes*.
- Perbedaan penelitian yang dilakukan dengan skripsi ini adalah penelitian ini hanya berfokus pada metode *Naïve Bayes* dan tidak dilakukan perbandingan dengan metode *Machine Learning* lainnya. Lalu pada penelitian ini tidak ada tahapan *Exploratory Data Analysis* dan *Grid Search Cross Validation* seperti yang akan dilakukan dalam skripsi ini.