

BAB II
TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian sebelumnya yang menjadi latar belakang pada penelitian ini, dijabarkan pada tabel dibawah ini :

Tabel 2.1 Penelitian Terkait

No	Peneliti	Judul	Metode	Pengambilan Data	Hasil
1	Gusti Nyoman Pardomuan, Ni Nyoman Parwati, Ketut Agustini [8] Tahun Publish : 2020	Sistem Personalisasi E-Learning Berorientasi Felder Silverman Learning Style Model Pada Mata Pelajaran Teknik Pengambilan Gambar	Felder-Silverman, E-Learning	Kuisisioner dengan jumlah responden 11 orang. Variabel yang digunakan adalah nama, kelas, gaya belajar, nilai pretest, nilai posttest	Terdapat perbedaan yang signifikan hasil belajar Teknik Pengambilan Gambar Bergerak kelas XI semester genap, antara sebelum dan sesudah menggunakan e-learning berorientasi FSLSM, yang berarti pula bahwa media yang dihasilkan dalam penelitian ini efektif untuk digunakan dalam proses pembelajaran
2	Septianti Wulansari [4] Tahun Publish : 2020	Kemampuan Penalaran Ilmiah Siswa Dalam Memecahkan Masalah Matematika Dibedakan Berdasarkan	Felder-Silverman	Kuisisioner dengan jumlah responden 4 orang yang diambil dari kelas VIII-D di SMPN 22 Surabaya. Variabel yang	Hasil Penalaran ilmiah siswa dalam memecahkan masalah matematika adalah sebagai berikut:

		Gaya Belajar Felder-Silverman		digunakan adalah nama, kelas, penalaran ilmiah, nilai matematika	<p>1. Gaya belajar sensing-active dalam memecahkan masalah matematika tergolong sedang.</p> <p>2. Gaya belajar sensing-reflective dalam memecahkan masalah matematika tergolong sedang.</p> <p>3. Gaya belajar intuitive-active dalam memecahkan masalah matematika tergolong tinggi.</p> <p>4. Gaya belajar intuitive-reflective dalam memecahkan masalah matematika tergolong sedang.</p>
3	Dani Indrobani Brahmantio, Yeni Anistiyasari [2] Tahun Publish : 2020	Studi Literatur Pengaruh Gaya Belajar Terhadap E-Learning Adaptive Berbasis Web	Felder-Silverman, E-Learning	E-Learning dengan data yang tidak diperoleh secara langsung dari subyek peneliti maupun objek penelitian melainkan sudah tersedia dan diperoleh dari pada penelitian orang lain. Analisis disini sendiri menggunakan acuan 11 jurnal.	Data sekunder yang diambil dari sebagian jurnal membuktikan bahwa E-learning Adaptive berbasis Web dengan menggunakan gaya belajar Felder-Silverman Learning Style Model (FSLSM), dan

				Variabel yang digunakan adalah sensorik, intuitif, visual, verbal, aktif, reflektif, sekuensial, global	algoritma Fuzzy C Means sering digunakan, dan dibutuhkan dalam menentukan materi pembelajaran yang sesuai dengan gaya belajar peserta didik.
4	Che Ghani Che Kob, Mai Shihah Abdulah, Arasinah Kamis, Zaliza Hanapi, Ridzuan Che Rus [3] Tahun Publish : 2016	Amalan gaya pembelajaran pelajar cemerlang di Politeknik Seberang Perai: Kajian pelajar Malaysia berdasarkan model Felder Silvermen	Felder-Silverman	E-Learning dengan jumlah responden 16 orang pelajar. Variabel yang digunakan adalah nama, sensorik, intuitif, visual, verbal, aktif, reflektif, sekuensial, global, purata himpunan nilai mata, index learning style	Tingkat praktik gaya belajar siswa untuk setiap dimensi bukanlah faktor utama pencapaian prestasi akademik yang sangat baik tetapi bagaimana siswa menggunakan kekuatan dan mengatasi kelemahan adalah pendorong utama pencapaian siswa. Siswa dapat memilih dan mempraktekkan tingkat gaya belajar yang mereka sukai.
5	Marzoan [1] Tahun Publish : 2016	Gaya Belajar Felder-Silverman dan Hasil Belajar Sains di Sekolah Dasar (SD)	Felder-Silverman	Kuisisioner dengan jumlah responden 33 siswa SD Negeri 2 Tanjung Kabupaten Lombok Utara. Variabel yang	Masing-masing siswa belajar dengan menggunakan dimensi gaya belajar aktif-reflektif, Sebagian besar

				digunakan adalah nama, aktif, reflektif, index learning style	siswa cenderung pada salah satu gaya. Dalam penelitian ini, anak kelas V SD Negeri 2 Tanjung lebih banyak memiliki gaya belajar aktif, yaitu 19 orang siswa dari 33 siswa, sisanya 14 orang memiliki kecenderungan gaya belajar reflektif
6	Ayi Muhammad Iqbal Nasuha, Mira Suryani [9] Tahun Publish : 2015	Pengembangan Personalisasi Gaya Belajar pada E-learning dengan Menggunakan Felder Silverman Learning Style Model untuk Sekolah Menengah Kejuruan (SMK)	Felder-Silverman, E-Learning	Kuisisioner dan E-Learning dengan jumlah responden 40 siswa dari kelas XI RPL. Variabel yang digunakan adalah nama, sensorik, intuitif, visual, verbal, aktif, reflektif, sekuensial, global, nilai	Penilaian sistem menurut 39 siswa adalah 82,14%. Di mana berdasarkan kategori kelayakan sistem, nilai tersebut masuk dalam kategori sangat baik. Penilaian sistem menurut 2 orang ahli adalah 75,79%. Berdasarkan kategori kelayakan sistem, nilai tersebut masuk dalam kategori sangat baik.

2.2. Gaya Belajar

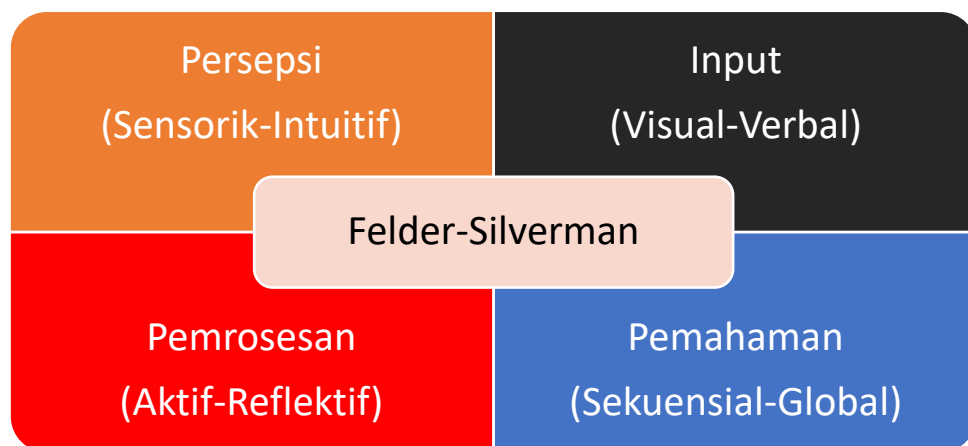
Secara umum, gaya belajar diartikan sebagai cara seorang individu dalam memproses informasi dengan tujuan mempelajari dan menerapkannya. Vermunt (1992) menggunakan istilah gaya belajar sebagai keseluruhan dari tiga domain yaitu proses kognisi dan afeksi terhadap materi, model belajar mental, dan orientasi belajar. Setiap individu memiliki perbedaan dalam memahami dan memproses informasi yang diberikan kepadanya. Perbedaan ini dinamakan dengan gaya belajar yang diartikan sebagai preferensi siswa terhadap proses atau aktivitas di dalam pembelajaran.

Grasha (1974) menekankan bahwa memahami tentang gaya belajar dan orientasinya membantu keberhasilan pembelajaran, peserta didik pun akan lebih senang dan lebih produktif bila mereka belajar dengan cara yang sesuai dengan gaya mereka sendiri. Beberapa ahli membagi gaya belajar melalui perspektif yang bervariasi sehingga didapatkan varian-varian pembagian gaya belajar. Curry (1987) membagi gaya belajar berdasarkan kelompok model pembelajaran Joyce dan Weil, yaitu:

1. Model pengolahan informasi, yaitu gaya belajar Felder-Silverman, Kolb.
2. Model personal, di dalamnya adalah gaya belajar Myers-Biggs Type Indicator, Witkin.
3. Model interaksi sosial, yaitu gaya belajar Reichman dan Grasha, Perry, Belenky.
4. Model pembelajaran behavior, dalam model ini ada Canfield, Dunn dan Dunn dan VAK. Adapun dalam penelitian ini difokuskan pada gaya belajar Felder-Silverman.

2.3. Model Gaya Belajar Felder-Silverman

Felder merupakan ilmuwan yang mengkaji gaya pembelajaran, pertama bersama Solomon (Felder & Solomon 1970) dan kajian yang sama diteruskan lagi bersama dengan Silverman (Felder & Silverman 1988) dengan sedikit perubahan. Menurut Felder-Silverman, gaya belajar siswa dikategorikan menjadi empat dimensi, yaitu pemrosesan (aktif atau reflektif), dimensi input (visual atau verbal), persepsi (kepekaan atau intuisi) dan kefahaman (sequential atau global). Untuk mengetahui gaya belajar seorang siswa, maka dilakukan pengukuran menggunakan Index of Learning Style (ILS) Questionnaire. Berikut ini adalah penjelasan masing-masing dimensi gaya belajar menurut Felder-Silverman [1][10].



Gambar 2.1 Dimensi Felder-Silverman

1. Dimensi Persepsi (Sensorik-Intuitif)

Sensorik : siswa lebih suka belajar fakta-fakta dan materi pembelajaran yang kongkrit. Mereka suka menyelesaikan problem dengan pendekatan yang baku dan cenderung lebih sabar dengan yang detil/rinci. Mereka lebih realistik dan bijaksana dan cenderung lebih praktis dibanding siswa yang intuitive. Mereka lebih suka menghubungkan materi yang dipelajari dengan dunia nyata. Intuitif : siswa lebih suka belajar materi pembelajaran abstrak,

lebih mampu menemukan kemungkinan-kemungkinan, relasi/hubungan dan cenderung lebih inovatif dan kreatif dibanding siswa yang memiliki gaya belajar sensing.

2. Dimensi Input (Visual-Verbal)

Visual : siswa dengan ciri mampu mengingat terbaik melalui belajar dengan apa yang mereka lihat. Verbal : siswa dengan ciri lebih suka belajar materi pembelajaran melalui representasi tekstual, baik berbentuk teks maupun narasi.

3. Dimensi Pemrosesan (Aktif-Reflektif)

Aktif : siswa belajar terbaik melalui bekerja secara aktif atas materi pembelajaran, dengan penerapan dan mencobanya. Disamping itu mereka cenderung tertarik untuk berkomunikasi dengan yang lain dan belajar secara berkelompok guna mendiskusikan materi yang telah dipelajari. Reflektif : siswa lebih suka berpikir dan merefleksikan materi pelajaran. Mereka lebih suka bekerja secara mandiri atau dalam sebuah kelompok kecil dengan teman baiknya.

4. Dimensi Pemahaman (Sekuensial-Global)

Sequential : siswa belajar dengan langkah peningkatan yang kecil, dan karenanya mempunyai kemajuan yang linier. Mereka cenderung mengikuti alur langkah-langkah yang logis dalam menemukan solusi. Global : siswa menggunakan proses berpikir yang holistik dan belajar dengan lompatan yang besar. Mereka cenderung menyerap materi belajar hampir secara acak tanpa melihat koneksinya, namun sesudah mereka belajar cukup atas materi

pembelajaran, secara tiba-tiba mereka memperoleh gambaran yang utuh/menyeluruh atas materi pembelajaran.

2.4. Klasifikasi

Klasifikasi adalah kata serapan dari bahasa Belanda, *classificatie* yang berarti pengelompokan. Menurut KBBI, klasifikasi adalah penyusunan bersistem dalam kelompok atau golongan menurut kaidah atau standar yang ditetapkan.

Menurut kamus Merriam-Webster, klasifikasi adalah pengaturan sistematis dalam kelompok atau kategori sesuai dengan kriteria yang ditetapkan. Sementara menurut Cambridge Dictionary, klasifikasi adalah tindakan atau proses dari membagi hal-hal ke dalam kelompok-kelompok sesuai dengan jenis mereka.

Klasifikasi adalah proses di mana ide dan objek dikenali, dibedakan, dan dipahami. Klasifikasi adalah proses memasukkan sesuatu ke dalam kategori. Klasifikasi melibatkan menempatkan sesuatu ke dalam kelas atau kelompok menurut karakteristik tertentu sehingga lebih mudah untuk memahaminya.

Klasifikasi adalah proses yang bertujuan mempermudah memahami beragam hal. Klasifikasi membantu untuk menempatkan berbagai hal dalam kategori tertentu. Ini membuat orang yang mempelajarinya tahu segera apa yang mereka lakukan dan bagaimana mereka berbeda dari objek lain.

Selain untuk mempermudah memahami objek, klasifikasi juga membuat segalanya lebih teratur. Menjaga klasifikasi tetap terorganisir memungkinkan orang lain untuk memperluas pekerjaan yang dilakukan para ilmuwan dalam meneliti dan bereksperimen.

2.5. Algoritma K – Nearest Neighbors (KNN)

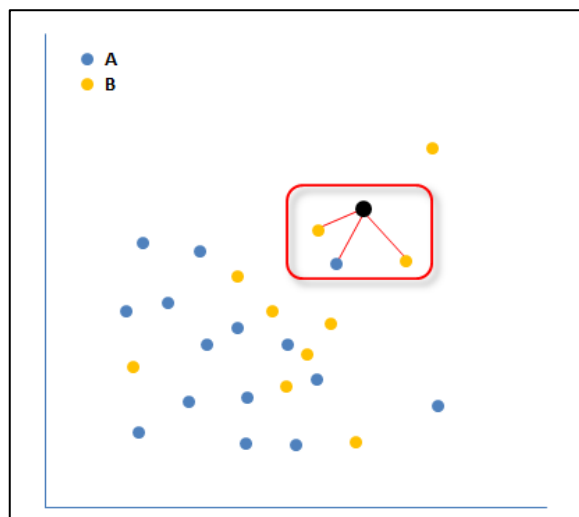
Algoritma K-Nearest Neighbors (KNN) merupakan algoritma klasifikasi yang bekerja dengan mengambil sejumlah K data terdekat (tetangganya) sebagai acuan untuk menentukan kelas dari data baru. Algoritma ini mengklasifikasikan data berdasarkan similarity atau kemiripan atau kedekatannya terhadap data lainnya.

Dalam K-Nearest Neighbor, data point yang berada berdekatan disebut “neighbor” atau “tetangga” [11].

Secara umum, cara kerja algoritma KNN adalah sebagai berikut :

1. Tentukan jumlah tetangga (K) yang akan digunakan untuk pertimbangan penentuan kelas.
2. Hitung jarak dari data baru ke masing-masing data point di dataset.
3. Ambil sejumlah K data dengan jarak terdekat, kemudian tentukan kelas dari data baru tersebut.

Perhatikan gambar ilustrasi berikut.



Gambar 2.2 Ilustrasi KNN

Dari gambar di atas, ada sejumlah data point yang terbagi menjadi dua kelas yaitu A (biru) dan B (kuning). Misalnya ada data baru (hitam) yang akan kita prediksi kelasnya menggunakan algoritma KNN. Dari contoh di atas, nilai K yang digunakan adalah 3. Setelah dihitung jarak antara titik hitam ke masing-masing data point lainnya, didapatkan 3 titik terdekat yang terdiri dari 2 titik kuning dan satu titik biru seperti yang diilustrasikan di dalam kotak merah, maka kelas untuk data baru (titik hitam) adalah B (kuning).

Menghitung jarak dengan Euclidean Distance. Untuk menghitung jarak antara dua titik pada algoritma KNN digunakan metode Euclidean Distance yang dapat digunakan pada 1-dimensional space, 2-dimensional space, atau multi-dimensional space. 1-dimensional space berarti perhitungan jarak hanya menggunakan satu variabel bebas (independent variable), 2-dimensional-space berarti ada dua variabel bebas, dan multi-dimensional space berarti ada lebih dari dua variabel.

Secara umum, formula Euclidean distance pada 1-dimensional space adalah sebagai berikut:

$$dis(x_1, x_2) = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2} \quad (2.1)$$

Formula di atas dapat digunakan jika jumlah independent variable hanya ada satu variabel. Jika ada lebih dari satu, kita dapat menjumlahkannya seperti berikut.

$$dis = \sqrt{\sum_{i=0}^n (x_{1i} - x_{2i})^2 + (y_{1i} - y_{2i})^2 + \dots} \quad (2.2)$$

2.6. Algoritma Naive Bayes

Naive Bayes Classifier merupakan sebuah metoda klasifikasi yang berakar pada teorema Bayes. Metode pengklasifikasian dengan menggunakan metode

probabilitas dan statistik yg dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dr Naive Bayes Classifier ini adalah asumsi yg sangat kuat (naif) akan independensi dari masing-masing kondisi / kejadian [12].

Naive Bayes Classifier bekerja sangat baik dibanding dengan model classifier lainnya. Hal ini dibuktikan oleh Xhemali, Hinde Stone dalam jurnalnya “Naive Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages” mengatakan bahwa “Naive Bayes Classifier memiliki tingkat akurasi yg lebih baik dibanding model classifier lainnya” [13].

Keuntungan penggunaan adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (training data) yg kecil untuk menentukan estimasi parameter yg diperlukan dalam proses pengklasifikasian. Karena yg diasumsikan sebagai variable independent, maka hanya varians dr suatu variable dalam sebuah kelas yg dibutuhkan unt menentukan klasifikasi, bukan keseluruhan dr matriks kovarians.

Bentuk umum Teorema Bayes adalah:

$$P(H | X) = \frac{P(X | H) \cdot P(H)}{P(X)} \quad (2.3)$$

Keterangan:

X : Sampel data dengan kelas yang belum diketahui

H : Hipotesis data X merupakan suatu kelas spesifik

$P(H|X)$: Probabilitas hipotesis H berdasar kondisi X (posterior probability)

$P(H)$: Probabilitas hipotesis H (prior probability)

$P(X|H)$: Probabilitas X berdasar kondisi pada hipotesis H

$P(X)$: Probabilitas dari X

Untuk menjelaskan metode Naive Bayes, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, metode Naive Bayes di atas disesuaikan sebagai berikut [14]:

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)} \quad (2.4)$$

Di mana Variabel C merepresentasikan kelas, sementara variabel $F_1 \dots F_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (Posterior) adalah peluang munculnya kelas C (sebelum masuknya sampel tersebut, sering kali disebut prior), dikali dengan peluang kemunculan karakteristik-karakteristik sampel pada kelas C (disebut juga likelihood), dibagi dengan peluang kemunculan karakteristik-karakteristik sampel secara global (disebut juga evidence). Karena itu, rumus di atas dapat pula ditulis secara sederhana sebagai berikut:

$$\text{Posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}} \quad (2.5)$$

Nilai Evidence selalu tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut nantinya akan dibandingkan dengan nilai-nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan.

Persamaan di atas merupakan model dari teorema Naive Bayes yang selanjutnya akan digunakan dalam proses klasifikasi.

Untuk klasifikasi dengan data kontinyu digunakan rumus Densitas Gauss :

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}} \quad (2.6)$$

Dimana :

P : Peluang

X_i : Atribut ke i

x_i : Nilai atribut ke i

Y : Kelas yang dicari

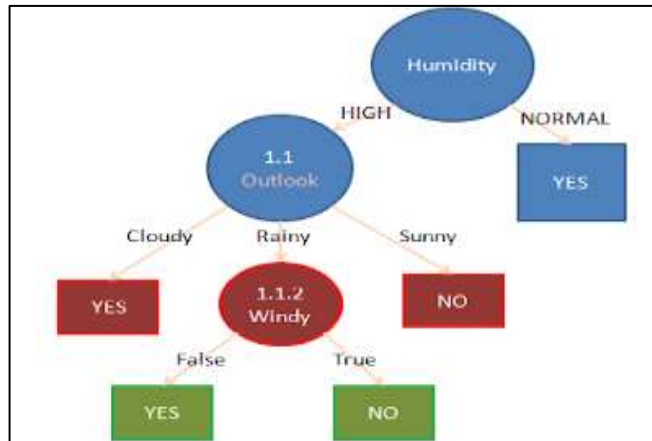
y_i : Sub kelas Y yang dicari

μ : mean, menyatakan rata – rata dari seluruh atribut

σ : Deviasi standar, menyatakan varian dari seluruh atribut.

2.7. Algoritma Decision Tree

Decision Tree merupakan salah satu cara data processing dalam memprediksi masa depan dengan cara membangun klasifikasi atau regresi model dalam bentuk struktur pohon. Hal tersebut dilakukan dengan cara memecah terus ke dalam himpunan bagian yang lebih kecil lalu pada saat itu juga sebuah pohon keputusan secara bertahap dikembangkan. Hasil akhir dari proses tersebut adalah pohon dengan node keputusan dan node daun. Sebuah node keputusan memiliki dua atau lebih cabang [15].



Gambar 2.3 Ilustrasi Pohon keputusan

Decision Tree juga berguna untuk dieksplorasi data, menemukan hubungan antara sejumlah calon variabel input dengan sebuah variabel target. Pohon keputusan eksplorasi data dan pemodelan yang salah langkah pertama yang sangat baik dalam proses pemodelan yang digunakan sebagai model akhir untuk beberapa teknik lainnya.

Kelebihan lain dari metode ini adalah mampu mengeliminasi perhitungan atau data-data yang tidak diperlukan. Karena sampel yang ada biasanya hanya diuji berdasarkan kriteria atau kelas tertentu. Meski memiliki banyak kelebihan, namun bukan berarti ini tidak memiliki kekurangan. Pohon keputusan ini mungkin tumpang tindih, terutama jika kelas dan kriteria yang digunakan sangat sering dapat meningkatkan waktu pengambilan keputusan sesuai dengan kapasitas memori yang diperlukan. [16].

Selain itu, algoritma tersebut dapat digunakan untuk mengurangi kesalahan pada program yang berulang. Algoritma sering digunakan sedemikian rupa sehingga masalah dapat diselesaikan secara logis dan berurutan. Dengan menggunakan algoritma yang tepat, program yang ada pun akan rapi, teratur, dan mudah dibuat. Jika terjadi kesalahan, bisa dapat segera menemukannya[17].

Didalam decision tree kita akan menemui entropy dan gain. Entropy adalah nilai informasi yang menyatakan ukuran ketidakpastian (impurity) dari attribut dari suatu kumpulan obyek data dalam satuan bit. Sedangkan gain adalah ukuran efektifitas suatu atribut dalam mengklasifikasikan data.

Perhitungan Entropy

$$\text{Entropy}(S) = \sum_{i=1}^n -p_i * \log_2 p_i \quad (2.7)$$

S : himpunan kasus

n : jml partisi S

Pi : proporsi dari Si terhadap S

Perhitungan Gain

$$\text{Gain}(S,A) = \text{Entropy}(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * \text{Entropy}(S_i) \quad (2.8)$$

S : Himpunan kasus

A : Atribut

n : jml partisi atribut a

|Si| : jml kasus pada partisi ke-I

|S| : jml kasus dlm S

2.8. Algoritma Random Forest

Random Forest adalah algoritma supervised learning yang dikeluarkan oleh Breiman pada tahun 2001 (Louppe, 2014). Random Forest biasa digunakan untuk menyelesaikan masalah yang berhubungan dengan klasifikasi, regresi, dan sebagainya[18]. Ada dua hal yang membuat algoritma ini disebut random, yaitu:

1. Setiap pohon tumbuh pada sampel bootstrap yang berbeda diambil dari data latih secara acak.
2. Dalam setiap node split selama pembentukan decision tree, sebagian sampel dari m variabel dipilih dari kumpulan data yang asli dan kemudian yang terbaik akan digunakan dalam node tersebut.

Algoritma ini berupa kombinasi dari beberapa tree predictors atau bisa disebut decision trees dimana setiap tree bergantung pada nilai random vector yang dijadikan sampel secara bebas dan merata pada semua tree dalam forest tersebut. Hasil prediksi dari Random Forest didapatkan melalui hasil terbanyak dari setiap individual decision tree (voting untuk klasifikasi dan rata-rata untuk regresi). Untuk RF yang terdiri dari N trees dirumuskan sebagai:

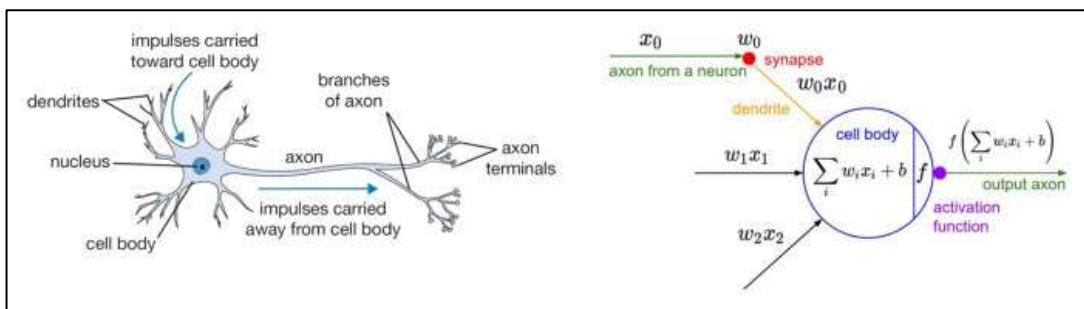
$$l(y) = \operatorname{argmax}_c \left(\sum_{n=1}^N I_{h_n(y)=c} \right) \quad (2.9)$$

Dimana I adalah fungsi indikator dan h_n adalah tree ke- n dari RF (Liparas, 2014). Random Forest memiliki mekanisme internal yang menyediakan estimasi dari generalization error-nya sendiri yang disebut out-of-bag (OOB) error estimate. Dalam pembentukan tree hanya $2/3$ dari data asli yang digunakan dalam pengambilan sampel bootstrap. Sedangkan $1/3$ sisanya diklasifikasikan oleh tree yang terbentuk dan digunakan untuk menguji performanya. OOB error estimation adalah rata-rata dari kesalahan prediksi untuk setiap kasus training y menggunakan tree yang tidak mengikutsertakan y dalam sampel bootstrap-nya. Kemudian, saat RF dibuat, semua training cases menyusuri setiap pohon dan matriks kedekatan setiap kasus dihitung berdasarkan pasangan kasus yang sampai di terminal node yang sama (Liparas, 2014). Banyak penelitian yang membuktikan bahwa Random

Forest memiliki performa prediksi yang baik dalam regresi serta klasifikasi di berbagai bidang seperti prediksi finansial, remote sensing, serta analisa genetik dan biomedis. RF juga menunjukkan performa yang lebih baik saat disandingkan dengan metode lain seperti, partial least squares regression, support vector machine dan Neural Network (Xu, 2013).

2.9. Algoritma Neural Network

Neural network adalah model yang terinspirasi oleh bagaimana neuron dalam otak manusia bekerja. Tiap neuron pada otak manusia saling berhubungan dan informasi mengalir dari setiap neuron tersebut. Gambar di bawah adalah ilustrasi neuron dengan model matematisnya[6].



Gambar 2.4 Ilustrasi Neural Network

Tiap neuron menerima input dan melakukan operasi dot dengan sebuah weight, menjumlahkannya (weighted sum) dan menambahkan bias. Hasil dari operasi ini akan dijadikan parameter dari activation function yang akan dijadikan output dari neuron tersebut.

Neural Network (Jaringan Saraf Tiruan) adalah prosesor tersebar paralel yang sangat besar dan memiliki kecenderungan untuk menyimpan pengetahuan yang bersifat pengalaman dan membuatnya siap untuk digunakan. NN ini merupakan sistem adaptif yang dapat merubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan

tersebut. Secara sederhana NN adalah sebuah alat pemodelan data statistik non-linear. NN dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola pada data. Neuron juga terdiri dari satu output. Outputnya adalah terbentuk dari pengolahan dari berbagai input oleh neuron-neuron[16].

Pada umumnya neural network dibagi berdasarkan layer-layer yaitu input layer, hidden layer dan output layer. Setiap node pada masing-masing layer memiliki suatu error rate, yang akan digunakan untuk proses training.

Pada Supervised Learning menggunakan Neural Network, pada umumnya Learning terdiri dari 2 tahap, yaitu training dan evaluation. Namun kadang terdapat tahap tambahan yaitu testing, namun sifatnya tidak wajib. Pada tahap training setiap weight dan bias pada tiap neuron akan diupdate terus menerus hingga output yang dihasilkan sesuai dengan harapan. Pada tiap iterasi akan dilakukan proses evaluation yang biasanya digunakan untuk menentukan kapan harus menghentikan proses training (stopping point). Proses training neural network terdiri dari 2 tahap:

- **Forward Pass**

Forward pass atau biasa juga disebut forward propagation adalah proses dimana kita membawa data pada input melewati tiap neuron pada hidden layer sampai kepada output layer yang nanti akan dihitung errornya.

$$dot_j = \sum_i^3 w_{ji}x_i + b_j$$

$$h_j = \sigma(dot_j) = \max(0, dot_j) \quad (2.10)$$

Persamaan diatas adalah contoh forward pass pada arsitektur pertama (lihat gambar arsitektur diatas) yang menggunakan ReLU sebagai activation function. Dimana i adalah node pada input layer (3 node input), j adalah

node pada hidden layer sedangkan h adalah output dari node pada hidden layer.

- **Backward Pass**

Error yang kita dapat pada forward pass akan digunakan untuk mengupdate setiap weight dan bias dengan learning rate tertentu. Kedua proses diatas akan dilakukan berulang-ulang sampai didapatkan nilai weight dan bias yang dapat memberikan nilai error sekecil mungkin pada output layer (pada saat forward pass).

2.10. K-fold Cross Validation

K-fold cross validation adalah teknik yang dapat digunakan jika memiliki jumlah data yang terbatas. Metode k-fold cross validation melakukan generalisasi dengan membagi data kedalam k bagian berukuran sama. Selama proses berlangsung, salah satu dari partisi dipilih untuk data uji, dan sisanya digunakan untuk data latih. Langkah ini di ulangi k kali sehingga setiap partisi digunakan untuk data uji tepat satu kali. Metode k-fold cross validation menetapkan $k = N$, ukuran dari data set. Pendekatan ini memiliki keuntungan dalam penggunaan data sebanyak mungkin untuk training. Test set secara efektif mencakup keseluruhan data set. Kekurangan data pendekatan ini adalah banyaknya komputasi untuk mengulangi prosedur sebanyak N kali. K-fold cross validation adalah salah satu teknik untuk mengevaluasi keakuratan model (Mustika, 2015 dikutip oleh Citra, 2015).

2.11. Confusion Matrix

Salah satu cara untuk menghitung akurasi adalah dengan menggunakan metode confusion matrix. Confusion matrix merupakan sebuah cara yang berguna untuk menganalisis seberapa baik classifier mengenali data. Metode ini

menggunakan tabel matriks seperti terlihat pada tabel 2.2 berikut ini, jika data set hanya terdiri dari dua kelas, kelas yang satu dianggap sebagai positif dan yang lainnya negatif.

Tabel 2.2 Model Confusion Matrix

Klasifikasi yang benar	Diklasifikasikan sebagai	
	Positif (+)	Negatif (-)
Positif (+)	True Positive	False Positive
Negatif (-)	True Negative	True Negative

True positive adalah jumlah record positif yang diklasifikasikan sebagai positif, false positive adalah jumlah record positif yang diklasifikasikan sebagai negatif, false negative adalah jumlah record negatif yang diklasifikasikan sebagai positif, true negative adalah jumlah record negatif yang diklasifikasikan sebagai negatif, kemudian masukan data uji (Mustika, 2015). Jumlah data akan mempengaruhi karena semakin banyak data akan semakin baik akurasi, tipe data akan berpengaruh terhadap algoritma yang dipakai dan akurasi algoritmanya, [19]. Setelah data uji dimasukan ke dalam confusion matrix, hitung nilai-nilai yang telah dimasukan tersebut untuk dihitung jumlah Akurasi, Recall, Presisi dan Specificity. Untuk menghitung digunakan persamaan dibawah berikut:

1. Akurasi merupakan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data.

$$\text{Akurasi} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{FN} + \text{TN})$$

2. Recall (Sensitifitas) merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.

$$\text{Recall} = (\text{TP}) / (\text{TP} + \text{FN})$$

3. Presisi merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif.

$$\mathbf{Precision} = (\mathbf{TP}) / (\mathbf{TP+FP})$$

4. Specificity merupakan kebenaran memprediksi negatif dibandingkan dengan keseluruhan data negatif.

$$\mathbf{Specificity} = (\mathbf{TN}) / (\mathbf{TN+FP})$$

Dimana:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative.