# Vol. 11 No. 2, Juli 2021

p-ISSN: 2087-7897 ; e-ISSN : 2460-5344 DOI : 10.30700/jst.v11i2.1134



Mengoptimalkan Teknik Pemilihan Fitur dan Pengurangan Fitur untuk Mengatasi Curse of Dimensionality dalam Model Pembelajaran Mesin

Optimizing Feature Selection and Feature Reduction Techniques to Alleviate Curse of Dimensionality in Machine Learning Models

**Toufan Marifan Nazmudin\*1, Muhamad Indra2, Agus Priyanto3, Ilham Maulana4, Yulita Ayu Wardani5**

Universitas Nusamandiri; Jl. Raya Jatiwaringin No.2, RT.8/RW.13, Cipinang Melayu, Kec. Makasar, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13620 Tel. (021)28534471

Program Studi Ilmu Komputer, Fakultas Teknologi dan Informasi, Jakarta

e-mail: **\***[114230002@nusamandiri.ac.id,](about:blank) [214230008@nusamandiri.ac.id](mailto:214230008@nusamandiri.ac.id), [314230013@nusamandiri.ac.id](mailto:214230008@nusamandiri.ac.id), [414230018@nusamandiri.ac.id](mailto:414230018@nusamandiri.ac.id), [514230021@nusamandiri.ac.id](mailto:514230021@nusamandiri.ac.id)

***Abstrak***

*Curse of Dimensionality menggambarkan sifat eksplosif dari peningkatan dimensi data dan akibatnya adalah peningkatan eksponensial dalam upaya komputasi yang diperlukan untuk pemrosesan dan/atau analisisnya. Istilah ini pertama kali diperkenalkan oleh Richard E. Bellman, untuk menjelaskan peningkatan volume ruang Euclidean terkait dengan penambahan dimensi ekstra, di bidang pemrograman dinamis. Dalam Machine Learning Curse of Dimensionality mulai mempengaruhi kesalahan generalisasi pada dimensi masukan yang lebih tinggi. meskipun dimensi input dikontrol, noise pada label target sangat memperkuat kesalahan. Sedangkan untuk ukuran patch menengah, bergantung pada resolusi input maksimum kumpulan data, kutukan dimensi dikurangi dan kesalahan generalisasi dibatasi.*

***Kata kunci****— Machine Learning, Curse of Dimensionality, noise*

***Abstract***

*Curse of Dimensionality describes the explosive nature of the increasing dimensionality of data and the resulting exponential increase in the computational effort required for its processing and/or analysis. The term was first introduced by Richard E. Bellman, to describe the increase in volume of Euclidean space associated with the addition of extra dimensions, in the field of dynamic programming. In Machine Learning the Curse of Dimensionality begins to affect generalization errors at higher input dimensions. even though the input dimensions are controlled, the noise in the target labels greatly amplifies the error. Meanwhile, for intermediate patch sizes, depending on the maximum input resolution of the dataset, the curse of dimensionality is reduced and the generalization error is limited.*

***Keywords—*** *Machine Learning, Curse of Dimensionality, noise*

1. PENDAHULUAN

*Curse of dimensionality* menggambarkan fenomena di mana ruang fitur menjadi semakin sempit karena semakin banyak dimensi pada kumpulan data pelatihan berukuran tetap. Secara intuitif, kita dapat membayangkan tetangga terdekat sekalipun berada terlalu jauh dalam ruang berdimensi tinggi untuk memberikan perkiraan yang baik.

*Curse of dimensionality* menghasilkan efek negatif yang negatif pada kemampuan generalisasi. Ketika perancang algoritma menggunakan ukuran sampel yang tidak memadai untuk melatih dan mengevaluasi algoritma untuk menemukan pola dalam konstruksi yang rumit (misalnya, kesehatan manusia), besar besar dari daerah blind spot dapat menyebabkan sangat bervariasi estimasi kinerja model yang sebenarnya. Variabilitas ini membuatnya sulit untuk secara akurat memperkirakan seberapa baik model akan bekerja pada data yang tidak terlihat, memberikan perancang algoritme pengertian yang tidak akurat seberapa baik kinerja model selama pengembangan. Jika Jika perkiraan yang salah adalah perkiraan yang terlalu tinggi dari kinerja yang sebenarnya, kegagalan besar dapat terjadi setelah model digunakan [1].

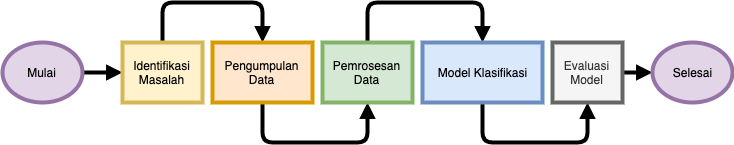
Regularisasi adalah salah satu cara untuk menghindari overfitting. Namun, dalam model yang regularisasinya tidak dapat diterapkan, seperti pohon keputusan dan KNN, kita dapat menggunakan teknik pemilihan fitur dan reduksi dimensi untuk membantu kita menghindari kutukan dimensi. Pengertian ini erat kaitannya dengan masalah overfitting. Karena data pelatihan tidak mencukupi, berisiko menghasilkan model yang sangat baik dalam memprediksi kelas target pada kumpulan data pelatihan, namun gagal total saat dihadapkan dengan data baru. Artinya, model tidak memiliki kekuatan generalisasi. Itulah mengapa sangat penting untuk mengevaluasi metode berdasarkan data yang sebelumnya tidak terlihat. Dalam *Supervised Learning* informasi dari satu jaringan neuron bertindak sebagai sinyal instruktif untuk mempengaruhi pola konektivitas di jaringan lain. Akibatnya, jaringan yang diinstruksikan belajar memproses informasi sehingga mencapai tujuan tertentu atau transformasi yang ditentukan oleh sinyal instruktif tercapai.

Dengan demikian, penelitian yang diawasi membentuk pola konektivitas secara efisien dan presisi yang tidak diperlukan. dan, seringkali, tidak dapat dikodekan dalam genom. Dengan mempelajari label tambalan, dan memberikan model teoritis sederhana yang dapat menjelaskan kinerja generalisasinya dengan mengurangi kutukan dimensi dan memperoleh batas atas kesalahan generalisasi.

Pada penelitian ini menggunakan dataset Darwin, dataset Toxicity, dataset Gene Expression Cancer RNA-Seq, dengan data tersebut, penelitian ini bertujuan untuk menghasilkan model terbaik untuk mendapatkan prediksi yang paling tinggi menggunakan berbagai metode. Untuk mendapatkan akurasi paling bagus, dalam penelitian ini digunakan beberapa data cleaning, Outlier Identification and Removal, Feature Selection, Splitting The Data into Training And Testing Dataset ,dan Uji metode tersebut dengan sedikitnya 5 Klasifikasi yang paling cocok untuk dataset.

1. METODE PENELITIAN

Tahapan penelitian yang digunakan yaitu, (1) Identifikasi Masalah, (2) Pengumpulan Data, (3) Pemrosesan Data, (4) Pembutan Model Klasifikasi, (5) Evaluasi Model Klasifikasi (6) Selesai. Tahapan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

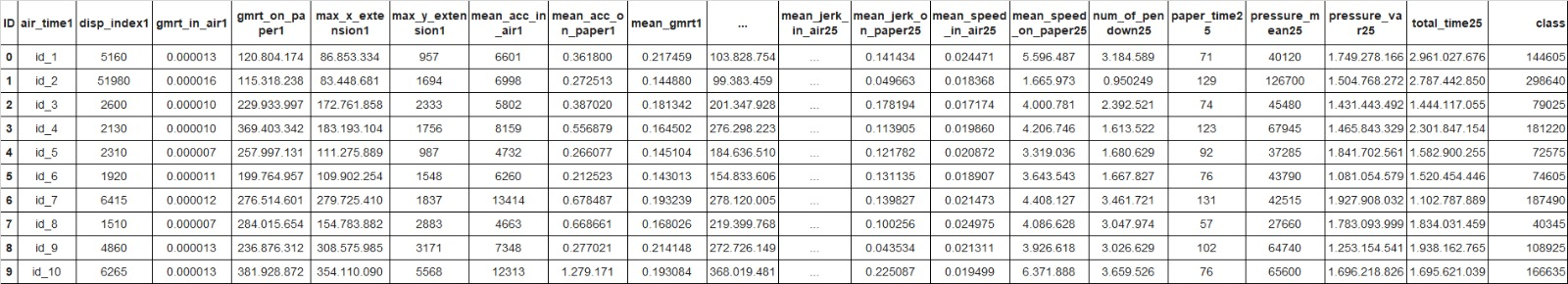
* 1. Identifikasi Masalah

Berdasarkan latar belakang yang telah dijabarkan, masalah yang ditemukan yaitu melakukan perbandingan algoritma *Logistic Regresion, Decision Tree,*  *support vector machine, random forest dan Random Forest* untuk mengetahui model klasifikasi mana yang memiliki akurasi terbaik dalam pengelompokan data.

* 1. Pengumpulan Data

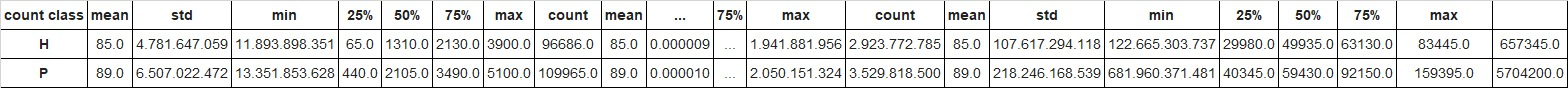
Data yang digunakan dalam penelitian ini diambil dari kumpulan dataset milik Darwin. Dataset darwin diperoleh dari UCI Machine Learning Repository yang diakses pada 5 November 2020. Mencakup data tulisan tangan dari 174 peserta. Tugas klasifikasi terdiri dari membedakan pasien penyakit Alzheimer dari orang sehat. Format data yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Fitur Dataset Darwin



Kelompok tulis tangan diambil berdasarkan *database* tipe pasien Alzheimer. Setiap kelompok pasien dikategorikan dengan huruf alphabet dan memiliki sekitar 21 variabel. Setiap jenis class juga memiliki deksripsinya sendiri dengan informasi penting seperti penjelasan umum, batas penggunaan, hingga informasi. Contoh kelompok pasien dapat dilihat pada Tabel 2.

Tabel 2. Contoh Kelompok Pasien



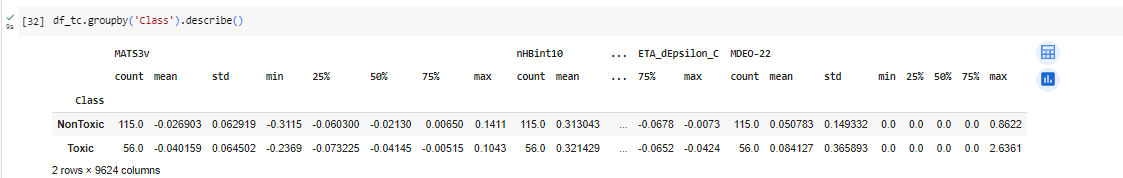
Dataset Toxicity dataset ini mencakup 171 molekul yang dirancang untuk domain fungsional protein jam inti, CRY1, yang bertanggung jawab untuk menghasilkan ritme sirkadian. 56 dari molekul-molekul tersebut bersifat toksik dan sisanya tidak toxic. Format data yang digunakan dapat dilihat pada Tabel 3.

Tabel 3. Fitur Dataset Toxicity



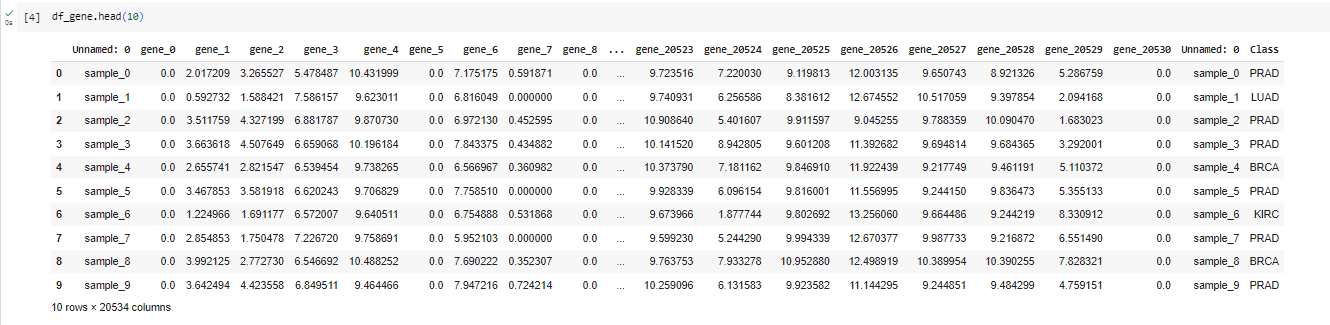
Kelompok dataset ini diambil berdasarkan *database* tipe molekul Nontoxic dan toxic, Contoh kelompok dataset toxicity dapat dilihat pada Tabel 4

Tabel 4. Contoh Kelompok Molekul



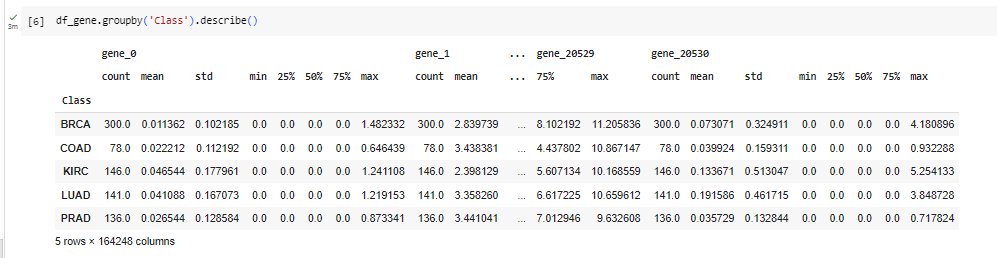
Dataset gene expression cancer RNA-Seq Donated merupakan data ekstraksi acak dari beberapa sample data ekspresi gen pasien yang memiliki jenis tumor yang berbeda: BRCA, KIRC, COAD, LUAD dan PRAD. Format data yang digunakan dapat dilihat pada Tabel 5

Tabel 5. Fitur Dataset gene expression



Kelompok dataset ini diambil berdasarkan *database* tipe gen pasien, Contoh kelompok dataset toxicity dapat dilihat pada Tabel 6

Tabel 6. Contoh Kelompok



* 1. Pemrosesan Data
     1. Penanganan *Missing Data*

Data yang didapatkan masih memiliki beberapa *field* kosong pada beberapa fitur sehingga membuat data dianggap belum memiliki nilai yang baik (*missing value*) sehingga diperlukan metode untuk melakukan normalisasi data[2]. Beberapa pendekatan untuk menangani masalah ini diantaranya dapat dilakukan mengganti data yang hilang dengan nilai rata- rata dari data historis yang tersedia [3].

* + 1. Penanganan Data Kategori (*label encoding*)

Klasifikasi membutuhkan pengelompokan setiap fitur. Data yang dimiliki saat ini merupakan data yang berjenis string sehingga kita dapat melakukan perubahan *string* menjadi nominal. Sebagai contoh *Class A* kita ubah dengan kategori H, *Class B* dengan kategori P dan seterusnya. Metode yang digunakan adalah *ordinal encoding* yaitu memberikan bilangan bulat pada setiap kategori serta tidak menambahkan kolom baru pada data, tetapi menyiratkan perintah ke variabel yang mungkin sebenarnya tidak ada [4].

* + 1. Penskalaan fitur (*Feature Scaling*)

Dalam tahap pemrosesan data *feature scaling* menjadi salah satu langkah yang penting untuk menyatukan variabel mandiri atau rentang fitur dalam data [5]. Dalam *feature scaling* kita dapat menggunakan teknik *normalization*. Teknik ini melakukan normalisasi data kolom fitur yang dimiliki pada kisaran [0,1] dengan menggunakan *min-max scalling.* Rumus transformasi standar dapat dilihat pada persamaan (1).

|  |  |
| --- | --- |
| 𝑋 − 𝑋𝑚𝑖𝑛    𝑋𝑠𝑡𝑎𝑛𝑑𝑎𝑟 = 𝑋 − 𝑋  𝑚𝑎𝑥 𝑚𝑖𝑛 | (1) |

Keterangan Rumus:

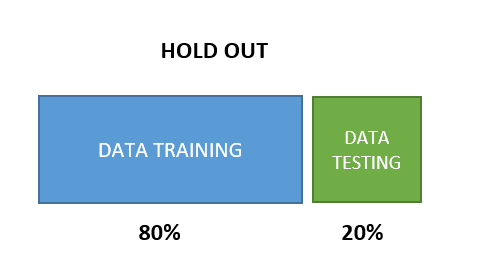
Xstandar = Nilai setelah normalisasi

X = Nilai sebelum normalisasi

Xmin = Nilai sebelum normalisasi minimal Xmax = Nilai sebelum normalisasi maksimal

* + 1. Pembagian Data (*data splitting*)

Dalam *machine learning* setelah melewati pemrosesan data sehingga menjadi data yang bernilai, dilakukan pembagian data yaitu *training*, *testing.* Kita melatih model klasifikasi dengan data *training*, dan mengujinya dengan data *testing*. Pembagian data ini dikenal dengan nama *hold out* dimana membagi dataset menjadi data dengan perbandingan tertentu seperti 80% untuk *training* dan sisanya 20% untuk *testing* [6]. Ilustrasi pembagian data dapat dilihat pada Gambar 3.

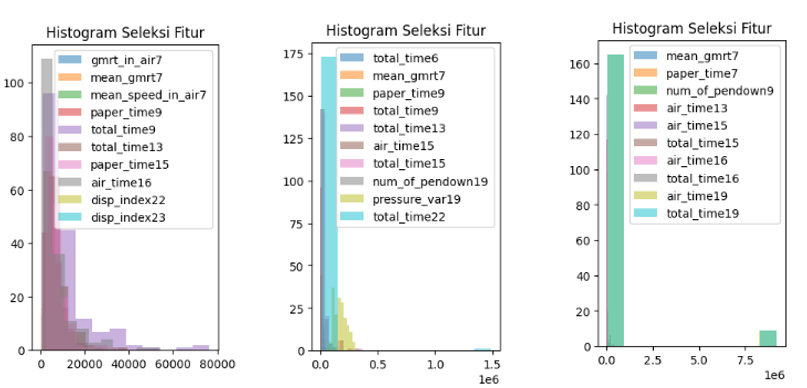


Gambar 3. *Hold out data*

* + 1. Pemilihan fitur (*Feature Selection*)

Metode pemilihan fitur diklasifikasikan menjadi pembungkus, metode filter,dan metode berbasis tertanam. Metode-metode ini menggunakan metode filter untuk melakukan praproses data dan mendapatkan peringkat setiap fitur dan menerapkan fitur peringkat tinggi ke sebuah prediktor Sedangkan pada metode pemilihan wrapper, fitur yang dipilih adalah fitur yang memberikan tingkat prediksi tertinggi [7].

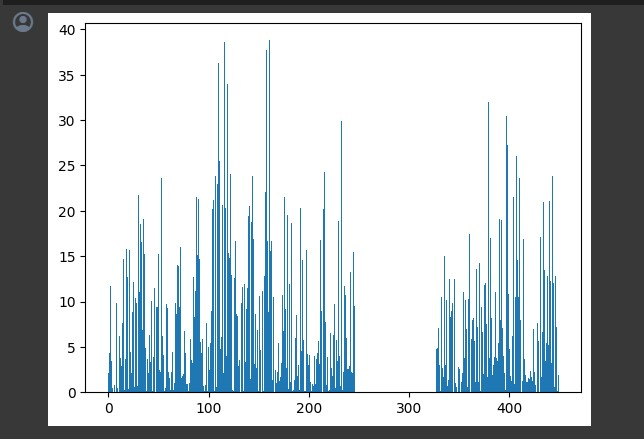
Histogram adalah representasi grafis dari distribusi warna citra digital. Ordinat vertikal mewakili piksel dengan nilai total setiap seri bin pada sumbu horizontal. Dalam statistik, histogram adalah representasi grafis dari tabel frekuensi yang diplot dalam diagram batang untuk menunjukkan pengelompokan data. Setiap grafik batang menunjukkan proporsi frekuensi di setiap rangkaian kategori yang berdekatan dengan interval yang tidak tumpang tindih seperti contoh gambar 4 di bawah ini [8].



Gambar 4. Histogram Selection Fitur

2.3.5.1 Feature Selection Anova

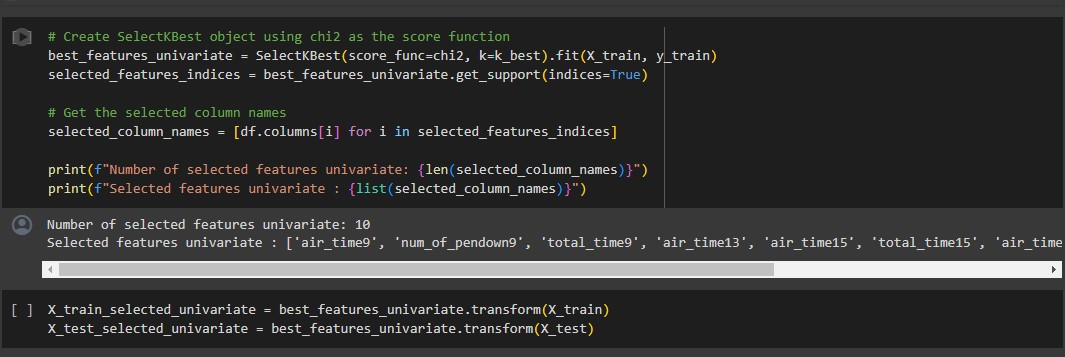
Berdasarkan statistik uji F Anova satu arah Skema ini diterapkan untuk menentukan fitur paling penting yang berkontribusi terhadap klasifikasi spam email. Fitur ini seleksi berdasarkan uji F Anova satu arah digunakan untuk mengurangi tingginya dimensi data ruang fitur sebelum proses klasifikasi. Percobaan skema yang diusulkan dilakukan dengan menggunakan kumpulan data benchmarking berbasis spam untuk mengevaluasi kelayakan metode yang diusulkan. Ilustrasi evaluasi data pada Anova dapat dilihat pada Gambar 5



Gambar 5. Feature Selection Anova

2.3.5.2 Feature Selection Univariate dan Multivariate

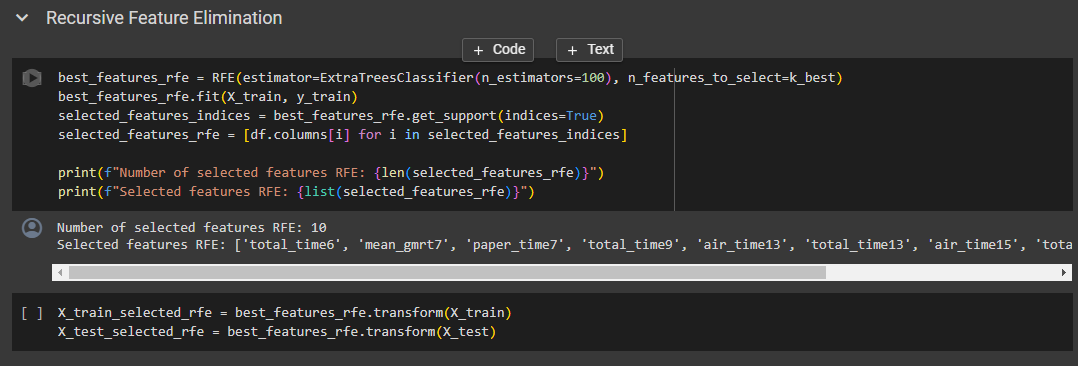
Pada dasarnya model peramalan time series ada dua jenis, yaitu univariat dan multivariat. Model univariat seperti Box Jenkins, hanya menggunakan satu variabel sebagai variabelnya input untuk membangun hubungan non-linier antara input dan output. Yang non-linier Hubungan tersebut akan digunakan untuk meramalkan keluaran masa depan yang diinginkan. Untuk penelitian ini, terhubung sepenuhnya, masukkan jaringan syaraf tiruan dengan satu yang tersembunyi algoritma pembelajaran lapisan dan propagasi balik diterapkan. seperti ilustrasi pada gambar 6.



Gambar 6. Feature Selection Univariate dan Multivariate

2.3.5.3 Feature Selection Recursive

Recursive Feature Elimination (RFE) memang merupakan metode yang dikembangkan khusus untuk pemilihan fitur dalam situasi dengan sampel terbatas, terutama dalam tugas klasifikasi dengan sedikit contoh latihan. Awalnya digunakan dalam klasifikasi kanker menggunakan data mikroarray, di mana ukuran sampelnya kurang dari 100 dan jumlah fiturnya mencapai ribuan, RFE telah terbukti efektif dalam memilih fitur yang relevan dalam sampel-sampel kecil [7]. RFE adalah contoh dari eliminasi fitur mundur yang bekerja untuk menghilangkan fitur berdasarkan aturan yang terkait dengan fungsi keputusan/diskriminasi melalui langkah-langkah iteratif [8].



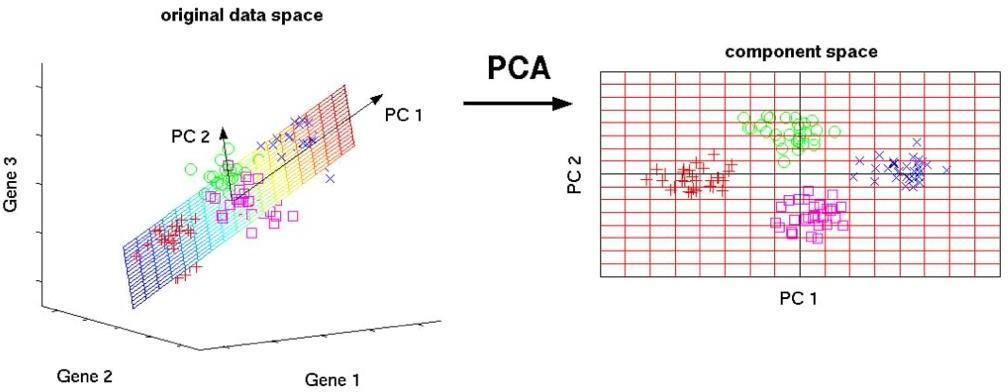
Gambar 7. Feature Selection Recursive

2.3.6 Fiture Reduction

Redudansi data membuat model klasifikasi menjadi kurang akurat (*overfitting*), oleh karena itu dibutuhkan mekanisme untuk memilih ataupun melakukan seleksi fitur dengan cara mengurangi dimensionalitas data [9].

2.3.6.1 PCA

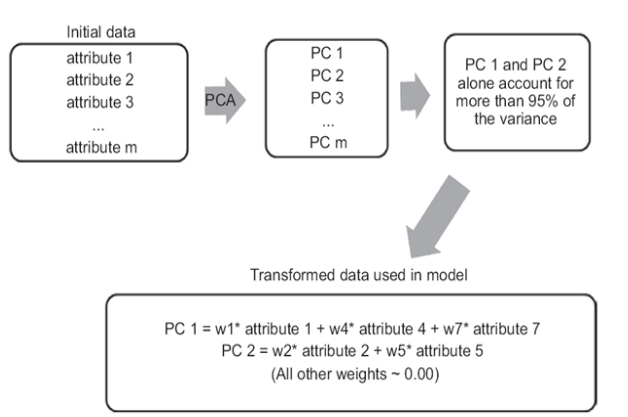
Salah satu metode feature selection yang dapat digunakan adalah *principal component analysis* yang berfungsi untuk mengurangi jumlah feature dari kumpulan data dengan mempertahankan varian sebanyak mungkin [10]. Ilustrasi penggunaan *principal component analysis* dapat dilihat pada Gambar 8.



Gambar 8. *Principal component analysis*

Dengan menggunakan PCA, variabel yang tadinya sebanyak n variabel akan diseleksi menjadi k variabel baru yang disebut principal component, dengan jumlah k lebih sedikit dari n. Dengan hanya menggunakan k principal component akan menghasilkan nilai yang sama dengan menggunakan nvariabel. Variabel hasil dari seleksi disebut principal component [11].

PCA digunakan untuk menjelaskan struktur matriks varians-kovarians dari suatu set variabel melalui kombinasi linier dari variabel-variabel tersebut. Secara umum principal component (PC) dapat berguna untuk seleksi fitur dan interpretasi variabel–variabel. Skema konseptual yang diilustrasikan bagaimana PCA dapat membantu untuk menyederhanakan dimensi dari data melalui hipotesa dataset berjumlah m variabel dapat ditunjukkan pada gambar berikut :

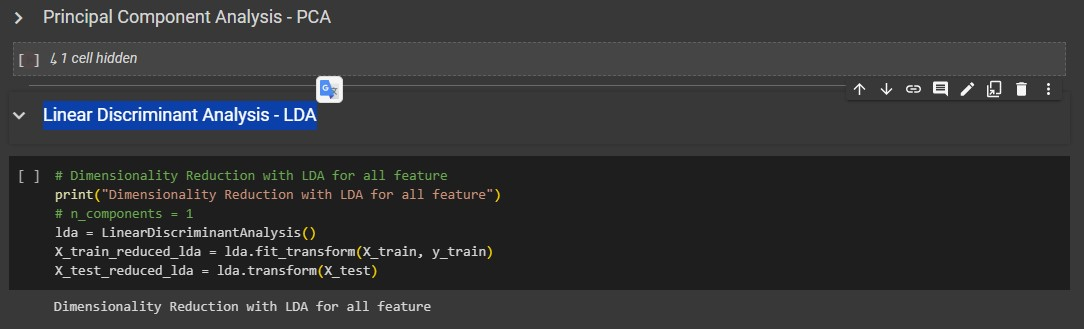


Gambar 9. Model Konseptual PCA

(Sumber: Kotu & Deshpande,2015 )

2.3.6.2 LDA

LDA merupakan pemecahan permasalahan yang berhubungan dengan klasifikasi pola, pada umumnya digunakan teknik berbasis LDA, dimana langkah awal adalah merepresentasi objek berdimensi rendah secara optimal dengan fokus pada ekstraksi ciri yang paling diskriminan, sementara hal ini sebenarnya hanya tepat untuk rekonstruksi objek. Sebagai akibatnya, kinerja klasifikasi LDA sering mengalami penurunan karena kriteria pemisahan tersebut tidak berkaitan secara langsung dengan akurasi klasifikasi di ruang output.



Gambar 10. Model Konseptual LDA

* 1. Model Klasifikasi

Kami menggunakan model klasifikasi untuk mendapatkan kelompok dataset. Klasifikasi merupakan salah satu metode yang dilakukan dengan cara *supervised learning* yaitu teknik pembelajaran data dalam machine learning menggunakan dataset (*data training*) yang sudah memiliki label [12]. Model klasifikasi ini kemudian dibuat dengan tiga algoritma *machine learning* yaitu *support vector machine*, *random forest* dan *decission tree*.

* + 1. *Support Vector Machine*

Algoritma *support vector machine* banyak digunakan untuk berbagai aktivitas seperti klasifikasi teks, pengenalan citra, analisis medis hingga prediksi [12]. SVM bekerja dengan menciptakan garis pembatas (*hyperplane*) terbaik yang menjadi fungsi pemisah. SVM akan memaksimalkan jarak dua set yang berbeda [13]. SVM menggunakan *kernel* berbentuk lurus yang membagi dua kelas dengan persamaan linier pada persamaan (2) [14].

|  |  |
| --- | --- |
| 𝑤 ∗ 𝑥 − 𝑏 = 0 | (2) |

Keterangan:

w = parameter *hyperlane* yang dicari

x = data input b = bias

Teknik mendapatkan *hyperplane* optimum pada klasifikasi SVM dapat dilakukan dengan persamaan (3) [15]..

|  |  |
| --- | --- |
| 𝑚𝑖𝑛 1 ||𝜔||2  2 | (3) |

|  |  |
| --- | --- |
| 𝑦𝑖(𝑤𝑥𝑖 + 𝑏) ≥ 1, 𝑖 = 1, … , 𝜆 | (4) |

Persamaan (4) digunakan untuk memaksimalkan kuantitas ||𝜔||2 dengan cara fokus pada pembatas 𝑦𝑖(𝑤𝑥𝑖 + 𝑏) ≥ 1, jika output data 𝑦𝑖 = +1 maka pembatas menjadi (𝑤𝑥𝑖 + 𝑏) ≥ 1 dan sebaliknya jika 𝑦𝑖 = −1 pembatas menjadi (𝑤𝑥𝑖 + 𝑏) − 1.

* + 1. *Random Forest*

Algoritma *random forest* atau *random ensembles* merupakan salah satu varian dari *bagging*. RF merupakan kombinasi pohon keputusan sedemikian hingga setiap pohon bergantung pada nilai- nilai *vector* acak yang disampling secara independen dan dengan distribusi yang sama untuk semua pohon dalam hutan tersebut. Kekuatan RF terletak pada seleksi fitur yang acak memilah setiap simpul, yang mampu menghasilkan tingkat kesalahan rendah [16].

Terdapat tiga aspek utama dalam menjalankan algoritma RF diantaranya adalah [17] :

* + - 1. Melakukan *bootstrap sampling* untuk membangun pohon prediksi.
      2. Setiap pohon keputusan memprediksi dengan prediktor acak.
      3. *Random forest* melakukan prediksi dengan mengombinasikan hasil dari setiap pohon keputusan dengan cara *majority vote* untuk melakukan klasifikasi
    1. *Decission Tree*

Algoritma *deccision tree* merupakan teknik klasifikasi yang umum digunakan untuk mengekstrak hubungan yang relevan dalam data [18]. Algoritma ini dikenal juga dengan nama *top-down induction of decission trees* yaitu sebuah teknik pembelajaran mesin membangun representasi aturan klasifikasi berstruktur sekuensial hirarki dengan cara mempartisi himpunan data *training* secara rekrusif. Pembelajaran ini menghasilkan pohon keputusan yang berupa *n-ary brancing tree* yang merepresentasikan suatu aturan klasifikasi [16].

2.4.4 Logistic Regression

Regresi Logistik (LR) adalah pendekatan klasifikasi lain yang menonjol. LR adalah teknik pembelajaran mesin yang dapat diterapkan pada masalah klasifikasi. Pierre Francais Verhulst mendefinisikan fungsi logistik dan atributnya dalam sebuah makalah yang diterbitkan di Proceedings of the Belgian Royal Academy dengan menentukan tiga parameter dan kurva yang melaluinya. Ini adalah metode pembelajaran mesin yang relatif sederhana yang digunakan secara luas. Pendekatan statistik untuk memprediksi kelas biner adalah regresi logistik, dalam hal ini, variabel dependen memiliki distribusi Bernoulli. Fungsi sigmoid, juga dikenal sebagai fungsi logistik, adalah kurva berbentuk 'S' yang mengambil nilai antara 0 dan 1. Jika kurva bergerak ke arah tak terhingga positif, 1 akan diantisipasi, dan jika bergerak ke arah tak terhingga negatif, 0 akan diprediksi [19]

2.4.5 KNN

k-nearest neighbor (k-NN) adalah algoritma supervised dimana hasil klasifikasi nya berdasarkan mayoritas [20] algoritma k-NN biasanya menggunakan Ecludean atau jarak manhattan. Namun jarak lain seperti norma Chebishev atau Mahalanob adalah jarak yang juga dapat digunakan.

x 2 = (c – a)2 + (d – b)

* 1. Evaluasi Model Klasifikasi

Evaluasi dilakukan untuk mengukur performa seberapa baik model yang digunakan. *Confusion matrix* bisa digunakan dalam tahap evaluasi ini [21]. Metode ini menghasilkan beberapa nilai yang digunakan sebagai evaluasi performa model yaitu *f1 score, accuracy, precission* dan *recall* [16].

Tabel 3. Model *Confusion Matrix*

|  |  |  |
| --- | --- | --- |
| Kelas Aktual | Kelas Hasil Prediksi | |
|  | + | - |
| + | True Positive (TP) | True Negative (TN) |
| - | False Positive (FP) | False Negative (FN) |

Tabel 3 menjelaskan bagaimana perhitungan pada *confusion matrix* [22]. TP yaitu jumlah *tuple* positif yang dilabeli dengan benar oleh model klasifikasi. TN yaitu jumlah *tuple* negatif yang dilabeli benar oleh model klasifikasi. FP yaitu jumlah *tuple* negatif yang

salah dilabeli oleh model klasifikasi dan FN yaitu jumlah tuple positif salah dilabeli oleh model klasifikasi [16]. Perhitungan nilai akurasi sebuah model dapat dilihat pada persamaan

(5). Akurasi merupakan tingkat identifikasi yang menghasilkan persentase dari jumlah data uji diklasfifikasi dengan benar.

|  |  |
| --- | --- |
| 𝑇𝑃 + 𝑇𝑁  𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦 =  𝑇𝑜𝑡𝑎𝑙 𝐷𝑎𝑡𝑎 | (5) |

Menurut [23] *precission* dapat digunakan untuk mengukur kepastian, yaitu berapa persentase tuple yang dilabeli sebagai positif adalah benar pada kenyataannya dan dirumuskan dengan persamaan (6).

|  |  |
| --- | --- |
| 𝑇𝑃  𝑃𝑟𝑒𝑐𝑖𝑠𝑠𝑖𝑜𝑛 =  𝑇𝑃 + 𝐹𝑃 | (6) |

*Recall* ialah ukuran kelengkapan, yaitu berapa persentase tuple positif yang dilabeli positif, dan dirumuskan dengan persamaan (7).

|  |  |
| --- | --- |
| 𝑇𝑃  𝑅𝑒𝑐𝑎𝑙𝑙 =  𝑇𝑃 + 𝐹𝑁 | (7) |

*F1 score* merupakan jumlah rata-rata harmonik antara precission dan recall [24] dan dirumuskan dengan persamaan (8).

|  |  |
| --- | --- |
| 𝑃𝑟𝑒𝑐𝑖𝑠𝑠𝑖𝑜𝑛 ∗ 𝑅𝑒𝑐𝑎𝑙𝑙  𝐹1 𝑠𝑐𝑜𝑟𝑒 = 2  𝑃𝑟𝑒𝑐𝑖𝑠𝑠𝑖𝑜𝑛 + 𝑅𝑒𝑐𝑎𝑙𝑙 | (8) |

1. HASIL DAN PEMBAHASAN
   1. Performa Model *Logistic Regresion*

Parameter yang digunakan pada model *Logistic Regresion* sebagai pembuat variabel dependen biner (dua kelas) dengan satu atau lebih variabel independen dalam klasifikasi. Model ini menghasilkan probabilitas bahwa instance tertentu termasuk dalam kelas positif. Hasil evaluasi dari model *Logistic Regresion* ditunjukan pada Tabel 4.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Darwin | 71.43 | 85.71 | 80 | 8 85.71 | 71.43 |
| Toxicity | 68.57 | 68.57 | 68.57 | 68.57 | 51.43 |
| Gene Expression | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |

Tabel 4. Performa Model *Logistic Regresion*

Berdasarkan Tabel 5, kita dapat menghitung nilai akurasi model *Logistic Regresion* dengan persamaan (5) feature selection. Tabelmodel *Logistic Regresion* ini dapat dijelaskan secara detail sebagai berikut:

1. Dataset darwin menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 71.43,pada feature selection *Univariate* 85.71, pada feature selection *RFE* 80. pada feature selection *PCA* 85.71, pada feature selection LDA 71.43, dengan akurasi tertinggi pada feature selection *Univariate* dan *PCA*.
2. Dataset Toxicity menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 68.57, pada feature selection *Univariate* 68.57, pada feature selection *RFE* 68.57. pada feature selection *PCA* 68.57, pada feature selection LDA 51.43, dengan akurasi tertinggi pada feature selection *Anova*,*Univariate, RFE* dan *PCA*.
3. Dataset Gene Expression menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 88.82, pada feature selection *Univariate* 88.82, pada feature selection *RFE* 88.82. pada feature selection *PCA* 88.82, pada feature selection LDA 88.82, dengan akurasi tertinggi pada semua feature selection.
   1. Performa *Random Forest*

Parameter yang digunakan pada model random forest ini yaitu *n\_estimator* yang merupakan jumlah pekerjaan yang dilakukan secara paralel dalam membuat pohon acak. Model ini menetapkan nilai n\_estimator = 100. Hasil evaluasi dari model *random forest* ditunjukan pada Tabel 5.

Tabel 5. Performa Model *Random Forest*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Darwin | 85.71 | 80 | 82.86 | 88.57 | 71.43 |
| Toxicity | 60 | 68.57 | 68.71 | 57.14 | 57.14 |
| Gene Expression | 88.2 | 88.82 | 88.82 | 88.82 | 84.47 |

Berdasarkan Tabel 6, kita dapat menghitung nilai akurasi model *Logistic Regresion* dengan persamaan (5) feature selection. Tabelmodel *Logistic Regresion* ini dapat dijelaskan secara detail sebagai berikut:

1. Dataset darwin menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 85.71,pada feature selection *Univariate* 80, pada feature selection *RFE* 82.86. pada feature selection *PCA* 88.57, pada feature selection LDA 71.43, dengan akurasi tertinggi pada feature selection *PCA*.
2. Dataset Toxicity menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 60, pada feature selection *Univariate* 68.57, pada feature selection *RFE* 68.71. pada feature selection *PCA* 57.41, pada feature selection LDA 57.41, dengan akurasi tertinggi pada feature selection *Univariate*.
3. Dataset Gene Expression menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 88.2, pada feature selection *Univariate* 88.82, pada feature selection *RFE* 88.82. pada feature selection *PCA* 88.82, pada feature selection LDA 84.47, dengan akurasi tertinggi pada feature selection *Univariate, RFE* dan *PCA*.
   1. Performa *Decission Tree*

Parameter yang digunakan pada model random forest ini yaitu *random state* yang merupakan pembuat bilangan acak dalam klasifikasi. Model ini menetapkan nilai random\_state dimulai dari angka 0. Hasil evaluasi dari model *decission tree* ditunjukan pada Tabel 6.

Tabel 6. Performa Model *Decission Tree*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Darwin | 82.86 | 77.14 | 80 | 8 82.86 | 71.43 |
| Toxicity | 54.29 | 68.57 | 54.29 | 68.57 | 57.41 |
| Gene Expression | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |

Berdasarkan Tabel 7, kita dapat menghitung nilai akurasi model *Logistic Regresion* dengan persamaan (5) feature selection. Tabelmodel *Logistic Regresion* ini dapat dijelaskan secara detail sebagai berikut:

1. Dataset darwin menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 82.86,pada feature selection *Univariate* 77.14, pada feature selection *RFE* 80. pada feature selection *PCA* 82.86, pada feature selection LDA 71.43, dengan akurasi tertinggi pada feature selection *Univariate* dan *PCA*.
2. Dataset Toxicity menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 52.29, pada feature selection *Univariate* 68.57, pada feature selection *RFE* 54.29. pada feature selection *PCA* 58.57, pada feature selection LDA 57.41, dengan akurasi tertinggi pada feature selection *Univariate* dan *PCA*.
3. Dataset Gene Expression menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 88.82, pada feature selection *Univariate* 88.82, pada feature selection *RFE* 88.82. pada feature selection *PCA* 88.82, pada feature selection LDA 88.82, dengan akurasi tertinggi pada semua feature selection.
   1. Performa *Support Vector Machine*

Parameter yang digunakan pada model random forest ini yaitu *random state* yang merupakan pembuat bilangan acak dalam klasifikasi. Model ini menetapkan nilai random\_state dimulai dari angka 0. Hasil evaluasi dari model *decission tree* ditunjukan pada Tabel 7.

Tabel 7. Performa Model *Support Vector Machine*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Darwin | 77.14 | 82.86 | 82.86 | 8 88.57 | 71.43 |
| Toxicity | 68.57 | 68.57 | 65.71 | 68.57 | 57.14 |
| Gene Expression | 88.82 | 88.82 | 88.82 | 88.2 | 88.82 |

Berdasarkan Tabel 7, kita dapat menghitung nilai akurasi model *Logistic Regresion* dengan persamaan (5) feature selection. Tabelmodel *Logistic Regresion* ini dapat dijelaskan secara detail sebagai berikut:

1. Dataset darwin menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 77.14,pada feature selection *Univariate* 82.86, pada feature selection *RFE* 82.86. pada feature selection *PCA* 88.57, pada feature selection LDA 71.43, dengan akurasi tertinggi pada feature selection *PCA*.
2. Dataset Toxicity menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 68.57, pada feature selection *Univariate* 68.57, pada feature selection *RFE* 65.71. pada feature selection *PCA* 68.57, pada feature selection LDA 57.41, dengan akurasi tertinggi pada feature selection *Univariate* dan *PCA*.
3. Dataset Gene Expression menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 88.82, pada feature selection *Univariate* 88.82, pada feature selection *RFE* 88.82. pada feature selection *PCA* 88.2, pada feature selection LDA 88.82, dengan akurasi tertinggi pada feature selection *Anova, Univariate, RFE & LDA*.

3.5 Performa *K-Nearest Neighbors*

Parameter yang digunakan pada model random forest ini yaitu *random state* yang merupakan pembuat bilangan acak dalam klasifikasi. Model ini menetapkan nilai random\_state dimulai dari angka 0. Hasil evaluasi dari model *K-Nearest Neighbors* ditunjukan pada Tabel 8.

Tabel 8. Performa Model *K-Nearest Neighbors*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Darwin | 74.29 | 77.14 | 71.43 | 8 88.57 | 71.43 |
| Toxicity | 68.57 | 62.86 | 68.57 | 51.43 | 57.14 |
| Gene Expression | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |

Berdasarkan Tabel 7, kita dapat menghitung nilai akurasi model *K-Nearest Neighbors* dengan persamaan (5) feature selection. Tabelmodel *K-Nearest Neighbors* ini dapat dijelaskan secara detail sebagai berikut:

1. Dataset darwin menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 74.29,pada feature selection *Univariate* 77.14, pada feature selection *RFE* 71.43. pada feature selection *PCA* 88.57, pada feature selection LDA 71.43, dengan akurasi tertinggi pada feature selection *PCA*.
2. Dataset Toxicity menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 68.57, pada feature selection *Univariate* 62.86, pada feature selection *RFE* 68.57. pada feature selection *PCA* 51.43, pada feature selection LDA 57.41, dengan akurasi tertinggi pada feature selection *Anova* dan *RFE*.
3. Dataset Gene Expression menggunakan feature selection, feature selection *Anova* tersebut memperoleh akurasi 88.82, pada feature selection *Univariate* 88.82, pada feature selection *RFE* 88.82. pada feature selection *PCA* 88.82, pada feature selection LDA 88.82, dengan akurasi tertinggi pada semua feature selection.

3.6 Perbandingan Performa Algoritma

Model klasifikasi yang telah dibuat kemudian dibandingkan akurasinya agar dapat terlihat algoritma mana yang memiliki performa terbaik. Perbandingan akurasi dari model klasifikasi dapat dilihat pada Tabel 9.

Tabel 9. Perbandingan Akurasi Model Klasifikasi dataset darwin

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset  darwin | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Logistic Regresion | 82.86 | 85.71 | 80 | 8 85.71 | 71.43 |
| Random Forest | 85.71 | 80 | 82.86 | 88.57 | 71.43 |
| Decission Tree | 82.86 | 77.14 | 80 | 82.86 | 71.43 |
| Support Vector Machine | 77.14 | 82.86 | 82.86 | 88.57 | 71.43 |
| K-Nearest Neighbors | 74.29 | 77.14 | 71.43 | 8 88.57 | 71.43 |

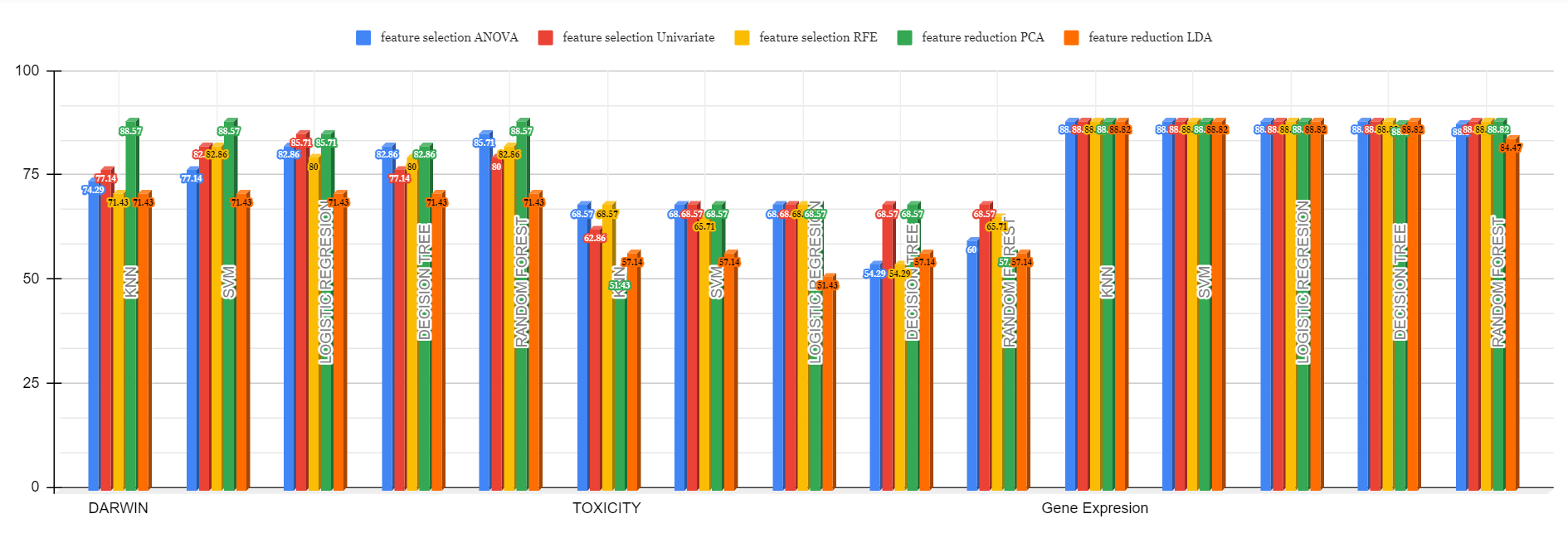
Tabel 10. Perbandingan Akurasi Model Klasifikasi dataset toxicity

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset  toxicity | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Logistic Regresion | 68.57 | 68.57 | 68.57 | 6 68.57 | 51.43 |
| Random Forest | 60 | 68.57 | 65.71 | 57.14 | 57.14 |
| Decission Tree | 54.29 | 68.57 | 54.29 | 68.57 | 57.14 |
| Support Vector Machine | 68.57 | 68.57 | 65.71 | 68.57 | 57.14 |
| K-Nearest Neighbors | 68.57 | 62.86 | 68.57 | 5 71.43 | 57.14 |

Tabel 11. Perbandingan Akurasi Model Klasifikasi dataset gene expresion

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset  gene expresion | *feature selection ANOVA* | *feature selection Univariate* | *feature selection RFE* | *feature reduction PCA* | *feature reduction LDA* |
| Logistic Regresion | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |
| Random Forest | 88.2 | 88.82 | 88.82 | 88.82 | 84.74 |
| Decission Tree | 88.82 | 88.82 | 88.82 | 88.2 | 88.82 |
| Support Vector Machine | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |
| K-Nearest Neighbors | 88.82 | 88.82 | 88.82 | 88.82 | 88.82 |

Hasil perbandingan akurasi model klasifikasi jika disajikan dalam bentuk *column chart* dapat dilihat pada Gambar 11. Kita dapat melihat adanya perbedaan nilai akurasi yang dihasilkan dari setiap model klasifikasi. Terlihat algoritma pada dataset *Toxicity* dengan menggunakan 5 model klasifikasi algoritma akurasi yang didapat masih cukup rendah di bawah 70%.



Gambar 11. Perbandingan Akurasi Model Klasifikasi

Berdasarkan Tabel 9, 10 dan 11. Dataset darwin menggunakan feature selection *PCA* memiliki nilai akurasi model sebesar 88.57% tertinggi dari fitur selection lain pada dataset darwin, lebih rendah dari dataset *Toxicity*  dengan nilai akurasi model tertinggi sebesar 67,14% pada feature selection *RFE* pada dataset toxicity*.* namun masih lebih rendah daripada dataset gene expression akurasi model algoritma pada 4 klasifikasi fitur selection *Univariate*, *RFE*, dan *PCA* dengan nilai 88,82%. Penelitian ini terkesan mendapatkan data akurasi yang sangat bagus yaitu mampu menembus angka 88,82% untuk 3 algoritma *Logistic Regresion*, *Support Vector Machine* dan *K-Nearest Neighbors* pada dataset Gene Expresion, namun kami masih perlu melakukan uji coba pada data dengan skala yang lebih besar dengan menggunakan beberapa model klasifikasi.

1. KESIMPULAN

Perbandingan algoritma *machine learning* yang telah dilakukan menghasilkan nilai akurasi yang hampir akurat yaitu *Logistic Regresion*, *Support Vector Machine* dan *K-Nearest Neighbors* dengan 5 feature selection nilai 88,82%, untuk *random forest* dengan nilai 86,64%dan *decission tree* memiliki nilai 88,51%, sementara itu pada dataset Toxicity akurasi yang di dapat masih rendah di antara dataset lain, dengan algoritma *Logistic Regresion*, *Support Vector Machine,K-Nearest Neighbors,random forest* dan *decission tree* nilai akurasi 67,14 pada fitur selection *RFE*. Penelitian ini menunjukan bahwa dataset gene expresion dengan algoritma *Logistic Regresion*, *Support Vector Machine* dan *K-Nearest Neighbors* lebih akurat dengan 5 feature selection yang digunakan daripada dataset Toxicity saat melakukan klasifikasi data. Hasil penelitian ini masih perlu diuji pada dataset lain serta dapat dibandingkan dengan algoritma klasifikasi lainnya.

1. SARAN

Saran kami pada penelitian selanjutnya yaitu mencari dan mengumpulkan *dataset* dengan pengelompokan data yang lebih banyak. Dataset dapat diolah lebih dahulu agar mendapatkan kualitas data yang lebih baik. Kita juga dapat melakukan perbandingan menggunakan algoritma *machine learning* lain yang dapat digunakan dalam masalah klasifikasi. Pengembangan penelitian harus terus dilakukan hingga mendapatkan model *machine learning* terbaik dalam upaya menemukan dan mempercepat yang tepat untuk mengelola dataset. Tahap berikutnya adalah model ini bisa diimplementasikan pada aplikasi berbasis *website* ataupun *mobile* agar lebih mudah dalam penggunaaanya.

ACKNOWLEDGEMENT

Pada penelitan ini kami menggunakan dataset Darwin, Toxicity dan Gene Expression cancer Rna Seq Dataset. Tautan situs webnya adalah sebagai berikut.

1.<https://archive.ics.uci.edu/dataset/732/darwin>

2.<https://archive.ics.uci.edu/dataset/728/toxicity-2>3.<https://archive.ics.uci.edu/dataset/401/gene+expression+cancer+rna+seq>

DAFTAR PUSTAKA

[1] Visar Berisha*.*, “Digital medicine and the curse of dimensionality,” 2021. [Online]. Available: https://doi.org/10.1038/s41746-021-00521-5.

[2] F. Bießmann *et al.*, “DataWig: Missing Value Imputation for Tables,” 2019. [Online]. Available: http://jmlr.org/papers/v20/18-753.html.

[3] P. Teknologi, D. Keselamatan, and R. Nuklir -Batan, “CLASSIFICATION OF MISSING VALUES HANDLING METHOD DURING DATA MINING: REVIEW Entin Hartini,” 2017.

[4] K. Potdar, T. S., and C. D., “A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers,” *Int J Comput Appl*, vol. 175, no. 4, pp. 7–9, Oct. 2017, doi: 10.5120/ijca2017915495.

[5] X. Wan, “Influence of feature scaling on convergence of gradient iterative algorithm,” in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jun. 2019. doi: 10.1088/1742-6596/1213/3/032021.

[6] J. Awwalu, F. Ogwueleka, and O. F. Nonyelum, “On Holdout and Cross Validation: A Comparison between Neural Network and Support Vector Machine On Holdout and Cross Validation A Comparison between Neural Network and Support Vector Machine,” 2019. [Online]. Available: www.ijtrd.com

[7] Y. Liu, F. Tang, and Z. Zeng, “Feature selection based on dependency margin,” *IEEE Trans Cybern*, vol. 45, no. 6, pp. 1209–1221, Jun. 2015, doi: 10.1109/TCYB.2014.2347372.

[8] Anggi Rizky Windra Putri, Anton Yudhana, Sunardi, “Klasifikasi Kanker Payudara Menggunakan Metode Digital Mammogram,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 9, no. 4, Dec. 2022, doi: 10.1186/s40537-020-00327-4.

[9] R. C. Chen, C. Dewi, S. W. Huang, and R. E. Caraka, “Selecting critical features for data classification based on machine learning methods,” *J Big Data*, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00327-4.

[10] R. Kohavi and G. H. John, “Wrappers for feature subset selection.” [Online]. Available: http://robotics.stanford.edu/

[11] S. Wang, J. Tang, and H. Liu, “Feature Selection,” in *Encyclopedia of Machine Learning and Data Mining*, Boston, MA: Springer US, 2016, pp. 1–9. doi: 10.1007/978-1-4899-7502-7\_101-1.

[12] S. M. Holland, “PRINCIPAL COMPONENTS ANALYSI S (PCA),” 2019.

[13] V. Kotu and B. Deshpande, *Predictive analytics and data mining : concepts and practice with RapidMiner*.

[14] E. Retnoningsih and R. Pramudita, “Mengenal Machine Learning Dengan Teknik Supervised dan Unsupervised Learning Menggunakan Python,” *BINA INSANI ICT JOURNAL*, vol. 7, no. 2, pp. 156–165, 2020, [Online]. Available: https://www.python.org/

[15] R. Umar, I. Riadi, U. Ahmad Dahlan Yogyakarta, J. D. Soepomo, and K. Umbulharjo, “Klasifikasi Kinerja Programmer pada Aktivitas Media Sosial dengan Metode Support Vector Machines,” *CYBERNETICS*, vol. 4, no. 01, pp. 32–40, 2020.

[16] A. S. Ritonga and E. S. Purwaningsih, “PENERAPAN METODE SUPPORT VECTOR MACHINE (SVM) DALAM KLASIFIKASI KUALITAS PENGELASAN SMAW (SHIELD METAL ARC WELDING),” 2018.

[17] R. Sagara and D. Maulina, “KLASIFIKASI ARTIKEL HOAX MENGGUNAKAN SUPPORT VECTOR MACHINE LINEAR DENGAN PEMBOBOTAN TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY.”

[18] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut.* . 2018.

[19] A. Primajaya and B. N. Sari, “Random Forest Algorithm for Prediction of Precipitation,” *Indonesian Journal of Artificial Intelligence and Data Mining (IJAIDM)*, vol. 1, no. 1, pp. 27–31, 2018.

[20] M. Maulidah *et al.*, “Algoritma Klasifikasi Decision Tree Untuk Rekomendasi Buku Berdasarkan Kategori Buku,” vol. 13, no. 2, pp. 89–96, 2020, [Online]. Available: http://journal.stekom.ac.id/index.php/E-Bisnis■page89

[21] A. B. Majumder, S. Gupta, D. Singh, and S. Majumder, “An intelligent system for prediction of COVID-19 case using machine learning framework-logistic regression,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Mar. 2021. doi: 10.1088/1742-6596/1797/1/012011.

[22] A. Ashari and A. M. Tjoa, “Performance Comparison between Naïve Bayes, Decision Tree and k-Nearest Neighbor in Searching Alternative Design in an Energy Simulation Tool,” 2013. [Online]. Available: www.ijacsa.thesai.org

[23] M. F. Rahman, M. Ilham Darmawidjadja, and D. Alamsah, “KLASIFIKASI UNTUK DIAGNOSA DIABETES MENGGUNAKAN METODE BAYESIAN REGULARIZATION NEURAL NETWORK (RBNN),” 2017.

[24] “554-Source Texts-1736-1-10-20191221”.

[25] I. Saputra and D. Rosiyadi, “Perbandingan Kinerja Algoritma K-Nearest Neighbor, Naïve Bayes Classifier dan Support Vector Machine dalam Klasifikasi Tingkah Laku Bully pada Aplikasi Whatsapp,” *Faktor Exacta*, vol. 12, no. 2, p. 101, Jul. 2019, doi: 10.30998/faktorexacta.v12i2.4181.

[26] G. Hackeling, *Mastering Machine Learning with scikit-learn*. 2014.