

BAB II

TINJAUAN PUSTAKA

2.1. Penyakit Ginjal Kronis

Penyakit Ginjal Kronis adalah hilangnya fungsi ginjal yang signifikan secara bertahap dari waktu ke waktu (*Your Kidneys & How They Work / NIDDK, 2020*). Dalam beberapa kasus, PGK dapat dicegah atau jika terdeteksi cukup awal, dapat diobati untuk mencegah atau menunda perkembangan menjadi penyakit ginjal stadium akhir (Aggarwal et al., 2016) . Beberapa kasus kerusakan ginjal berawal dari kerusakan salah satu bagian dari ginjal yaitu glomerulus yang berfungsi untuk menyaring darah untuk membiarkan air dan molekul kecil masuk ke urin tetapi mempertahankan sel dan molekul besar seperti protein (Akben, 2018). Tetapi, faktor-faktor risiko utama untuk penyakit ginjal yang pertama adalah diabetes karena kadar glukosa darah yang tinggi dan yang tidak terkontrol dapat merusak nefron di ginjal, yang kedua adalah hipertensi karena dapat merusak pembuluh darah dalam ginjal, yang ketiga adalah faktor usia, usia yang rentan terkena adalah diatas 60 tahun, yang keempat adalah riwayat keluarga yang pernah terkena penyakit ginjal dan yang terakhir adalah memiliki riwayat penyakit jantung (*Kidney Disease / Lab Tests Online, 2020.*)

PGK memiliki 5 tahap kerusakan ginjal, dari kerusakan yang sangat ringan pada tahapan pertama hingga gagal ginjal pada tahapan kelima (Webster et al., 2017). Tahapan-tahapan ini diukur dari seberapa baik ginjal menyaring limbah dan cairan ekstra dari darah. Pada tahap awal, ginjal tetap dapat menyaring dengan baik

tetapi pada tahap selanjutnya ginjal akan sulit dalam menyaring dan mungkin dapat berhenti bekerja sama sekali (Vijayarani & Mr.S.Dhayanand, 2015). Cara dokter untuk mengetahui seberapa baik ginjal dapat menyaring darah adalah dengan *estimated glomerular filtration rate* (e-GFR) (Boukenze et al., 2016). e-GFR berasal dari angka yang didasarkan test yang berasal dari limbah dalam darah atau biasa disebut juga kreatinin (*Stages of Chronic Kidney Disease (CKD) - American Kidney Fund (AKF)*, 2020.). Formula untuk menghitung e-GFR (*Estimating Glomerular Filtration Rate / NIDDK*, 2020) adalah sebagai berikut

$$\text{GFR} \left(\frac{\text{mL}}{\text{min}} \text{m}^2 \right) = 175 \times (\text{Scr})^{-1.154} \times (\text{Age})^{-0.203} \times$$

(0.742 if female) \times (1.212 if African American).

Rumus 2.1. Menghitung e-GFR

Berikut merupakan tabel 2.1 yang merupakan tahapan dalam penyakit ginjal kronis berdasarkan *American Kidney Fund* yang merupakan organisasi non-profit yang bergerak untuk melawan penyakit ginjal.

Tabel 2.1. Tahapan Penyakit Ginjal Kronis

Tahapan	Deskripsi	GFR(mL/min/1.73 m2)
1	kerusakan ginjal (ditemukannya protein dalam urin) dengan GFR normal	≥ 90
2	kerusakan ginjal dan adanya penurunan pada GFR yang masih sedikit	60-89
3B	kerusakan ginjal dan adanya penurunan pada GFR yang moderate	45-59
3A	kerusakan ginjal dan adanya penurunan pada GFR yang moderate ke berat	30-44
4	kerusakan ginjal dan adanya penurunan pada GFR yang parah	15-29

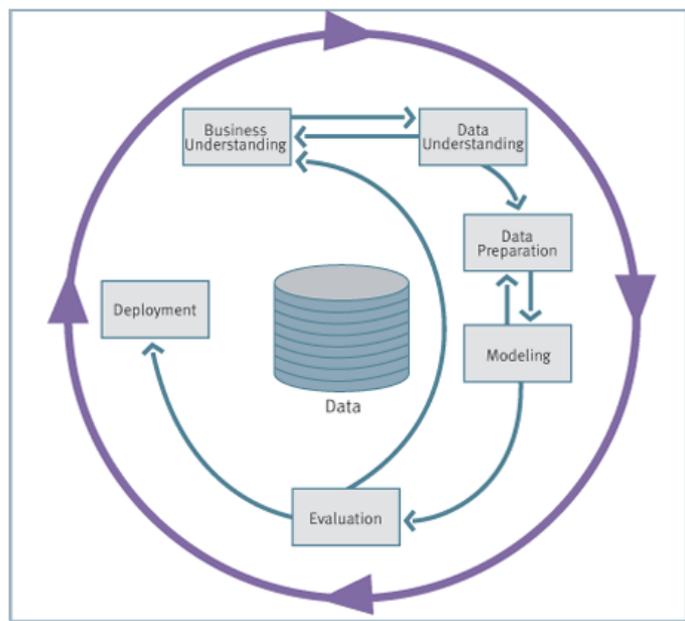
5	gagal ginjal terminal	< 15
---	-----------------------	------

Tahapan PGK pertama ditandai dengan memiliki e-GFR sebesar 90 ataupun lebih besar dan tahapan kedua dengan e-GFR berada diantara 60 dan 89. Dalam kedua tahapan ini ginjal masih dapat bekerja dengan baik tetapi memiliki tanda-tanda kerusakan ginjal lainnya. Tanda-tanda kerusakan ginjal bisa berupa protein dalam urin atau kerusakan fisik pada ginjal. Pada tahapan ketiga, e-GFR berada diantara 30 dan 59 yang berarti ada kerusakan pada ginjal sehingga ginjal sudah tidak dapat berfungsi dengan baik sebagaimana mestinya. Pada tahapan keempat, e-GFR diantara 15 dan 30 yang berarti ginjal sudah rusak dan tidak berfungsi sehingga harus ditangani dengan serius karena merupakan tahap terakhir sebelum gagal ginjal. Pada tahapan kelima memiliki e-GFR kurang dari 15 yang berarti hampir gagal ginjal, jika sudah sampai gagal ginjal maka kotoran darah akan menumpuk sehingga akan membuat pengidapnya merasa sangat sakit (*Stages of Chronic Kidney Disease (CKD) - American Kidney Fund (AKF), 2020.*).

2.2. CRISP-DM

Penerapan *data mining* di industri kesehatan sudah menjadi peran yang sangat penting seperti dalam mengolah data pasien yang kompleks dan besar, sumber daya rumah sakit, diagnosa penyakit, catatan pasien elektronik, peralatan medis ,dll (*Durairaj & Ranjani, 2013*) . Beberapa project *data mining* di industri kesehatan sudah banyak yang mengikuti *Cross Industry Standard Process for Data mining* atau biasa disingkat dengan (*CRISP-DM*) (Tahmasebian et al., 2017) yang dikembangkan oleh (Pete et al., 2000) pada awal tahun 2000.

Metode *CRISP-DM* memberikan pedoman dan kerangka kerja yang seragam untuk proyek-proyek penambangan data. Proses dapat digambarkan sebagai siklus, di mana iterasi dilakukan sebelum mencapai hasil akhir dan tujuan bisnis (Moro et al., 2011). *CRISP-DM* dibagi menjadi beberapa siklus dan berikut merupakan gambar proses siklus *CRISP-DM* yang terdiri dari 6 siklus :



Gambar 2.1. Proses CRISP-DM

Sumber : (Pete et al., 2000)

Tiap tahapan *CRISP-DM* pada gambar 2.1 diatas akan diterangkan pada penjelasan berikut ini dalam bidang kesehatan menurut (Shi-nash & R. Hardoon, 2017), yaitu:

- a. *Business Understanding* adalah tahapan pertama yang menentukan arah dan tujuan pembuatan *data mining* dan mengandung pemahaman objek suatu masalah bisnis dan persyaratan yang dibutuhkan supaya

data mining yang dibentuk dapat tercapai dan menyiapkan strategi awal dalam mencapai tujuan pembuatan *data mining*.

- b. *Data Understanding* adalah tahapan kedua yang menganalisa koleksi data dan analisis untuk membangun keakraban dengan data, mengidentifikasi kualitas masalah, wawasan tentang data, mengevaluasi kualitas data dan jika diperlukan dapat dilakukan pengambilan sebagian kecil kelompok data yang mengandung pola permasalahan.
- c. *Data Preparation* adalah tahapan ketiga dimana mengeksekusi data mentah menjadi data final, termasuk dengan *data cleansing* dan *data preprocessing*, memilah-milah data, atribut dan tabel yang akan dipakai dengan tujuan agar *dataset* tersebut cocok dengan *modeling tools* yang akan dipakai.
- d. *Modeling* adalah tahapan keempat dimana teknik *modeling* dipilih, diaplikasikan dan dioptimalkan. Jika ternyata model yang dihasilkan tidak optimal dapat kembali ke tahapan ketiga untuk mendapatkan kebutuhan dari teknik *data mining* tertentu.
- e. *Evaluation* adalah tahapan kelima yang dimana sebuah model telah selesai dibuat dan dievaluasi apakah tujuan bisnis pada tahapan pertama sudah tercapai. Serta menetapkan apakah permasalahan penting dari bisnis atau penelitian tertangani dengan baik atau tidak.
- f. *Deployment* adalah tahapan terakhir dalam *CRISP-DM* melibatkan organisasi dan presentasi pengetahuan yang dihasilkan oleh model

dalam bentuk yang mudah dimengerti dan berguna bagi *end-user*. *Deployment* menggunakan model yang telah dihasilkan sehingga terbentuknya model menandakan bahwa masalah bisnis telah terselesaikan.

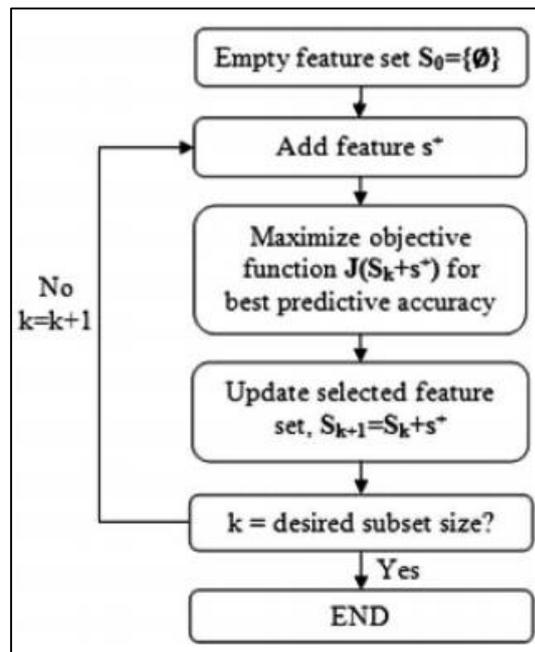
2.3. *Feature selection*

Feature selection atau yang biasa disebut dengan *feature*, *subset selection*, *attribute selection* atau *variable selection* adalah salah satu teknik data *preprocessing* di *data mining* yang dapat mengoptimalkan dan mempercepat proses dari sebuah algoritma dengan cara memilih fitur atau atribut yang baik untuk digunakan dalam sebuah proses *classification* (Supriyanti et al., 2016). konsep dari teknik ini adalah untuk memilih atribut yang ada dalam sebuah *dataset* karena tidak semua atribut mempunyai efek yang relevan terhadap masalah yang sedang dihadapi sedangkan di sisi lain, beberapa atribut dapat menyebabkan gangguan serta mengurangi akurasi, maka dari itu atribut yang tidak memiliki efek dapat dihapus untuk meningkatkan nilai akurasi (Tabakhi et al., 2014). Teknik *Feature selection* ini memiliki 2 jenis metode yaitu *forward selection* dan *backward elimination*. penggunaan *forward selection* bertujuan untuk menemukan *subset* fitur optimal untuk setiap klasifikasi serta untuk meningkatkan performa akurasi (Jiang & Wang, 2018) sedangkan penggunaan *backward elimination* bertujuan untuk meningkatkan keakuratan proses klasifikasi, dengan menambahkan metode ini pada proses pengolahan datanya diharapkan dapat diperoleh akurasi yang maksimal untuk pengklasifikasian data (Herliawan et al., 2020). Dalam penelitian

ini, kedua metode tersebut akan di uji performa akurasi nya dengan algoritma yang telah disebutkan.

2.3.1. *Forward selection*

Metode *forward selection* dimulai dengan *zero variable (empty model)* lalu variabel tersebut masuk satu per satu sampai kriteria tertentu telah terpenuhi (Zeniarta et al., 2020). Berikut merupakan langkah-langkah dari metode *forward selection* yang berbentuk *flowchart* menurut (Yang, 2011).



Gambar 2.2. Flowchart Forward selection

Sumber : (Yang, 2011)

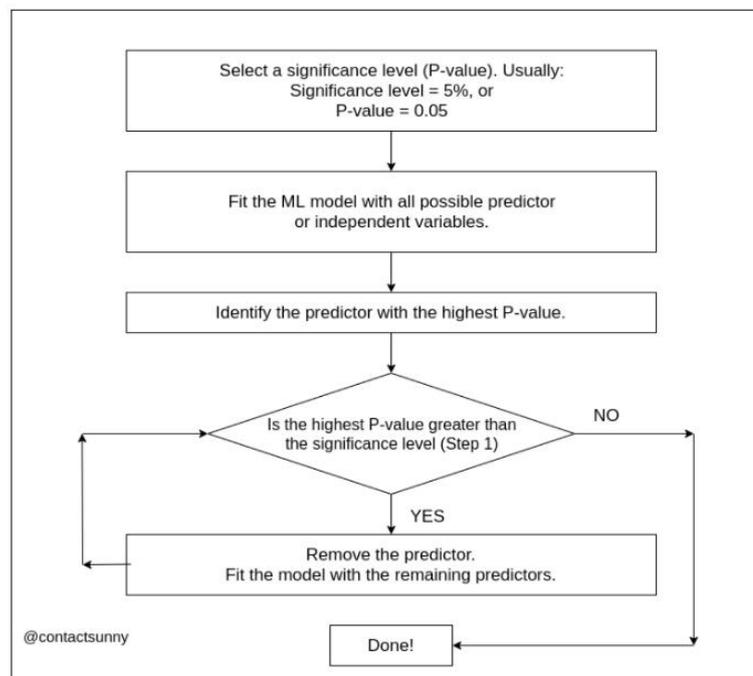
Dari gambar 2.2 diatas dapat dilihat *flowchart* dari metode *forward selection* terdiri dari tahapan-tahapan sebagai berikut dan akan diterangkan dalam penjelasan dibawah ini :

- a. Dimulai dengan fitur kosong dalam model

- b. Fitur yang relevan tersebut secara berurutan ditambahkan kedalam fitur model
- c. Langkah tersebut dilakukan secara berulang sampai fitur yang diinginkan telah memenuhi ukuran dari *subset* yang telah ditentukan

2.3.2. Backward elimination

Prosedur dari metode *backward elimination* dimulai dengan full variabel, lalu variabel yang tidak signifikan tersebut satu per satu di keluarkan sampai variabel yang tidak signifikan sudah tidak ada (Yang, 2011). Berikut merupakan langkah-langkah dari metode *backward elimination* berbentuk *flowchart*.



Gambar 2.3. Flowchart Backward elimination.

Sumber : (Yang, 2011)

Dari gambar 2.3 diatas dapat dilihat *flowchart* dari metode *backward elimination* terdiri dari tahapan-tahapan sebagai berikut dan akan diterangkan dalam penjelasan dibawah ini.

- a. Menentukan *P-Value* terlebih dahulu, umumnya *P-Value* bernilai sebesar 0.05.
- b. Setelah itu, memulai dengan semua prediktor atau variabel independent.
- c. Identifikasilah prediktor dengan *P-Value* tertinggi
- d. Jika nilai *P-Value* tertinggi lebih besar daripada target yang telah ditentukan, maka hapus prediktor tersebut.
- e. Jika tidak, maka model telah selesai dibentuk.

2.4. Algoritma *K-Nearest Neighbour*

K-nearest neighbour atau yang biasa disingkat dengan K-NN adalah teknik algoritma klasifikasi untuk mengklasifikasikan objek berdasarkan contoh pelatihan terdekat diruang masalah. K-NN adalah tipe pembelajaran berbasis contoh, atau *lazy learning* dimana fungsinya hanya untuk didekati secara lokal dan semua perhitungan ditangguhkan sampai klasifikasi (Lloyd-Williams, 1998). Menurut (Sinha, 2015) K-NN memiliki penerapan yang sederhana namun efektif dalam banyak kasus serta K-NN memiliki data latih (*training data*) yang sangat kuat dan cepat meskipun memiliki *noise* pada data. K-NN juga mempunyai perfoma yang baik dimana sebuah sampel memiliki banyak *label class* (Jadhav & Channe, 2013).

Sebelum melakukan perhitungan K-NN, harus melakukan pembagian data menjadi data latih (*training set*) dan data uji (*testing set*). Setelah dilakukan

pembagian data, akan dilakukan proses perhitungan untuk mencari jarak antara dua titik pada data latih dan data uji menggunakan *euclidean distance* (Nugraha, pratama dwi,, Said al faraby, 2018) . Berikut merupakan rumus *euclidean distance* beserta keterangan yang akan diterangkan dibawah ini.

$$d(a, b) = \sum_{i=0}^n (X_i - Y_i)^2$$

Rumus 2.2. Menghitung *euclidean distance*

Keterangan :

$d(a,b)$ = jarak *euclidean*

x = data 1

y = data 2

i = fitur ke –

n = jumlah fitur

Berikut merupakan langkah-langkah algoritma K-NN yang berbentuk *flowchart* menurut (Kasper, Dennis L, 2020) dan (Dzikrulloh & Setiawan, 2017) meliputi :



Gambar 2.4. flowchart K-NN

Dari gambar 2.4 diatas dapat dilihat bahwa *flowchart* dari K-NN terdiri dari tahapan-tahapan sebagai berikut dan akan diterangkan dalam penjelasan dibawah ini :

- a. Pertama-tama, sebelum mulai mencari jarak data ke tetangga adalah dengan menentukan nilai K tetangga (*Neighbour*).
- b. Setelah itu, bagi *dataset* menjadi 2 bagian yaitu data latih (*training set*) dan data uji (*testing set*).
- c. Kemudian baru mulai menghitung jarak *Euclidean Distance* antara data latih (*training set*) dan data uji (*testing set*).
- d. Lalu urutkan *dataset* tersebut yang mempunyai jarak terkecil.

- e. Tentukan kelompok data hasil uji berdasarkan *label* mayoritas dari K tetangga terdekat.

2.5. Algoritma *Decision-Tree*

Algoritma *Decision tree* adalah algoritma yang biasa digunakan untuk pengambilan keputusan, cara kerja *decision tree* adalah dengan mencari solusi permasalahan dengan menjadikan kriteria sebagai *node* yang saling berhubungan dan membentuk seperti struktur pohon (Hermawanti & Safriandono, 2016). Algoritma *decision tree* berdasarkan pada pendekatan yang bersifat *divide-and-conquer* dan bekerja dari atas kebawah yang mencari setiap atribut dan membaginya kedalam bagian terbaik dalam suatu *class* lalu memproses secara rekursif submasalah yang telah dihasilkan dari pembagian yang telah dibagi tersebut (Witten et al., 2011).

Proses dalam algoritma *decision tree* adalah sebagai berikut (Sartika & Indra, 2017):

- a. Mengubah data yang berbentuk tabel menjadi sebuah model *tree*(pohon) .

Pada langkah ini adalah untuk menentukan atribut yang akan terpilih mulai dari akar, cabang hingga menuju sebuah keputusan. Untuk menentukan atribut tersebut, dapat menggunakan metode perhitungan *gainration* dari setiap kriteria dengan data sampel. Berikut merupakan rumus untuk menghitung *gainratio*.

$$Gainration(S, A) = \frac{Gain(S,A)}{Splitinformation(S,A)}$$

Rumus 2.3. Menghitung Gain Ratio

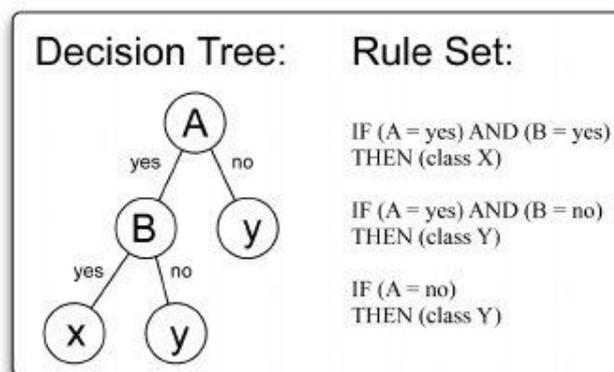
Dimana, nilai *information gain* memiliki makna seberapa banyak informasi yang diperoleh dengan cara mengetahui nilai dari suatu atribut dan nilai untuk *split information* dipergunakan untuk atribut yang mempunyai *instance* lebih banyak.

- b. Setelah itu, maka model *tree* akan diubah menjadi sebuah *rule* dengan formula sebagai berikut

IF premis THEN konklusi

Rumus 2.4. Rule Model Tree

Bagian *root* dan *branch* dari model *tree* akan menjadi premis dari sebuah *rule* sedangkan *leaf* akan menjadi bagian dari konklusinya. Gambar 2.5 dibawah ini merupakan contoh dari pembuatan *rule* dari sebuah model *tree*.



Gambar 2.5. Rule Decision tree

Sumber : (Freitas et al., 2010)

Proses terakhir merupakan menyederhanakan rule yang telah dibuat atau biasa disebut dengan *prunning*. *Prunning* dilakukan

untuk mengurangi kompleksitas pengklasifikasi akhir sehingga dapat meningkatkan akurasi prediksi dengan pengurangan *overfitting*. Berikut merupakan langkah-langkah untuk *prunning*:

- Membuat tabel distribusi yang menyatakan semua nilai kejadian yang terjadi pada setiap *rule*.
- Melakukan perhitungan terhadap tingkat independensi antara kriteria pada suatu *rule* yang merupakan antara atribut dengan target atribut.
- Melakukan eliminasi terhadap kriteria yang dianggap tidak perlu seperti kriteria yang mempunyai tingkat independensi yang tinggi.

2.6. Algoritma *Logistic regression*

Logistic regression merupakan salah satu algoritma analisis *multivariate* yang berfungsi dalam memprediksi variabel dependen berdasarkan variabel independen (Nawawi, 2018). *Logistic regression* juga merupakan salah satu metode statistika yang sering kali dipakai untuk menganalisa data yang mendeskripsikan satu atau lebih variabel prediksi dengan variabel respon yang bersifat kotomi yang hanya bernilai 1 (ya) atau 0 (tidak) sehingga hasil dari variabel respon tersebut akan mengikuti distribusi *Bernoulli* (Faiza & Ratnasari, 2019). Fungsi *classifier* untuk *logistic regression* ditunjukkan oleh formula sebagai berikut (Aziz, 2020).

$$\log\left(\frac{p}{1-p}\right) = \sum_{i=1}^{i=n} \beta^{(i)} * x^{(i)} + e = \beta^T x + e$$

Rumus 2.5. Fungsi *Classifier* pada *Logistic Regression*

dimana, $\beta = (\beta^{(1)}, \beta^{(2)}, \dots, \beta^{(n)})$ adalah vektor koefisien *hyperplane* sehingga didapatkanlah model persamaan *logistic regression* sebagai berikut.

$$p \frac{e^{\beta^T x + e}}{1 + e^{\beta^T x + e}}$$

Rumus 2. 6. Model Persamaan *Logistic Regression*

Selain itu, untuk memperoleh estimasi dari parameter *logistic regression* dapat dilakukan dengan cara *maximun likelihood estimation* (MLE). Teknik MLE juga merupakan teknik yang digunakan untuk mencari titik tertentu sehingga dapat memaksimalkan sebuah fungsi dan teknik ini juga memprediksi β yang berumus sebagai berikut (Bimantara & Dina, 2019).

$$L(\beta) = \ln(l(\beta)) = \sum_{j=0}^p [\sum_{i=1}^n y_i x_{ij}] \beta_j - \sum_{i=1}^n \ln[1 + \exp(\sum_{j=0}^p \beta_j x_{ij})] \quad (1)$$

2.7. Tools / Alat Bantu Software

Tools yang dipakai dalam penelitian ini adalah menggunakan *data science tools* Rapid Miner karena menurut (Dwivedi et al., 2016) Rapid Miner memiliki kemampuan grafis yang baik dan mudah digunakan dan menyediakan dukungan visualisasi yang bersifat *limited*. Serta pada tahun 2013 dan 2014 *annual software poll KDNuggets* memberi peringkat Rapid Miner sebagai *software data analytics* yang paling populer digunakan yang berasal dari *polling* respondennya. Rapid Miner juga menerima salah satu peringkat kepuasan pelanggan terkuat pada tahun 2011 di *Rexer Analytics Data Miner Survey* (Dwivedi et al., 2016).

2.8. Penelitian Terdahulu

Dibawah ini merupakan tabel 2.2 tentang perbandingan serta perkembangan penelitian sebelumnya yang berhubungan dengan deteksi pada PGK. Jurnal tersebut antara lain sebagai berikut.

Tabel 2.2. Penelitian Sebelumnya

Nama Jurnal	Penulis/ Tahun	Vol/ No	Masalah	Hasil	Simpulan
<i>Prediction of kidney disease stages using data mining algorithms</i>	(Rady & Anwar, 2019)	15/12	Deteksi dan karakterisasi dini dianggap sebagai faktor penting untuk pengendalian PGK.	Akurasi PNN 96.7% , akurasi SVM 60.7% , akurasi RBF 87% dan akurasi MLP 51.5%	Bahwa algoritma PNN menghasilkan akurasi klasifikasi dan prediksi kinerja yang lebih baik untuk memprediksi tahapan pasien PGK
Implementasi <i>data mining</i> untuk deteksi penyakit ginjal kronis (PGK) menggunakan <i>k-nearest neighbor</i> (K-NN) dengan <i>backward elimination</i>	(Gamadenda & Waspada, 2020)	07/02	Penyakit ginjal kronis (PGK) merupakan masalah kesehatan publik di seluruh dunia dengan insiden yang terus meningkat	Metode K-NN dengan eliminasi <i>backward</i> hasil akurasi 99,25% , sensitivitas sebesar 99,5%	Menggunakan fitur <i>backward</i> akurasi dan sensitivitas kinerja yang dihasilkan lebih baik daripada tanpa atribut
Analisis Penyakit Ginjal Kronis menggunakan teknik klasifikasi <i>data mining</i>	(Kunwar et al., 2016)	16/06	Sejumlah besar data dikumpulkan untuk menemukan informasi tersembunyi untuk diagnosis dan pengambilan keputusan yang efektif	<i>Naive Bayes</i> adalah <i>classifier</i> paling akurat daripada ANN	Dalam penelitian ini, beberapa faktor yang dipertimbangkan adalah usia, diabetes, tekanan darah, jumlah RBC dll

Tabel 2. 3. Penelitian Sebelumnya

<p>Rancang Bangun Aplikasi Pendeteksi Penyakit Ginjal Kronis dengan Menggunakan Algoritma C4.5</p>	<p>(Rianto & Iswari, 2017)</p>	<p>09/01</p>	<p>PGK merupakan <i>silent disease</i> dimana penderita tampak normal dan tidak menunjukkan gejala tetapi hasil tes menyatakan fungsi ginjal penderita mengalami penurunan</p>	<p>Rancang bangun aplikasi pendeteksi penyakit ginjal kronis dengan menggunakan algoritma C4.5 telah berhasil dibuat, dengan bantuan dari <i>weka library</i> dalam pembuatan algoritma <i>decision tree</i></p>	<p>Proses pengujian dilakukan dengan menggunakan <i>cross-validation</i> dan berdasarkan hasil yang sudah dihitung aplikasi ini memiliki akurasi 91.50% pada saat <i>decision tree</i> dibuat tanpa menggunakan <i>preprocess menu</i>.</p>
<p><i>classification of liver disease by applying random forest algorithm and backward elimination</i></p>	<p>(Herliawan et al., 2020)</p>	<p>06/01</p>	<p>Deteksi dini terhadap kanker khususnya kanker hati atau liver sangat penting dilakukan untuk menanggulangi resiko kematian sangat tinggi disebabkan oleh kanker hati atau liver</p>	<p>permodelan <i>backward elimination</i> untuk optimasi akurasi serta algoritma <i>Random Forest</i> dan validasi <i>split validation</i> untuk memvalidasi permodelan. Hasil dari penelitian ini menunjukkan hasil akurasi sebesar 76.00% dan nilai AUC yaitu 0.758</p>	<p>Hasil tersebut menunjukkan bahwa hasil penelitian ini cukup baik untuk membantu mengklasifikasi kanker hati atau liver</p>

Tabel 2. 4. Penelitian Sebelumnya

Implementasi greedy <i>forward selection</i> untuk prediksi metode penyakit kutil menggunakan <i>decision tree</i>	(Arifin, 2020)	05/01	Penyakit kutil dapat ditangani dengan berbagai metode seperti cryotherapy dan immunotherapy akan tetapi dokter belum mengetahui metode pengobatan yang paling tepat untuk pasien.	Hasil penelitian pada <i>dataset</i> cryotherapy menggunakan <i>Decision tree</i> mendapatkan nilai akurasi 90%, sedangkan hasil seleksi fitur menggunakan <i>Greedy Forward selection</i> sebesar 92,22%.	Hasil ini menunjukkan bahwa seleksi fitur <i>Greedy Forward Selection</i> dapat meningkatkan kinerja dari model
--	----------------	-------	---	--	---

Berdasarkan hasil dari beberapa penelitian terdahulu pada tabel 2.2 diatas, mendapatkan hasil bahwa penggunaan teknik *feature selection* sangatlah berpengaruh dalam meningkatkan performa akurasi dari suatu model klasifikasi dan algoritma yang dipakai seperti K-NN, *decision tree* dan *logistic regression* yang mendapatkan performa akurasi yang cukup tinggi. Sehingga dapat ditarik kesimpulan bahwa tinjauan pustaka sebelumnya mendukung penelitian yang akan dilakukan pada skripsi ini. Selain itu deteksi dini PGK penting untuk dilakukan agar tenaga medis dapat mengetahui metoda apa yang cocok dan tepat untuk pasien serta dapat mengurangi biaya rumah sakit jika tahapan pada PGK terdeteksi lebih awal.